This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

Embedding X.509 Digital Certificates in Three-Dimensional Models for Authentication, Authorization, and Traceability of Product Data

Thomas D. Hedberg, Jr.*
National Institute of Standards and Technology
Gaithersburg, Maryland 20899

Sylvere Krima
Engisis LLC
Bethesda, Maryland 20817

Jaime A. Camelio
Grado Department of Industrial and Systems Engineering
Virginia Tech
Blacksburg, Virginia 24061

ABSTRACT

Exchange and reuse of three-dimensional (3D)-product models are hampered by the absence of trust in product-lifecycle-data quality. The root cause of the missing trust is years of "silo" functions (e.g., engineering, manufacturing, quality assurance) using independent and disconnected processes. Those disconnected processes result in data exchanges that do not contain all of the required information for each downstream lifecycle process, which inhibits the reuse of product data and results in duplicate data. The X.509 standard, maintained by the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T), was first issued in 1988. Although originally intended as the authentication framework for the X.500 series for electronic directory services, the X.509 framework is used in a wide range of implementations outside the originally intended paradigm. These implementations range from encrypting websites to software-code signing, yet X.509 certificate use has not widely penetrated engineering and product realms. Our approach is not trying to provide security mechanisms, but equally as important, our method aims to provide insight into what is happening with product data to support trusting the data. This paper provides a review of the use of X.509 certificates and proposes a solution for embedding X.509 digital certificates in 3D models for authentication, authorization, and traceability of product data. This paper also describes an application within the Aerospace domain. Finally, the paper draws conclusions and provides recommendations for further research into using X.509 certificates in product lifecycle management (PLM) workflows to enable a product lifecycle of trust.

Keywords: Trustworthiness, Authentication, Authorization, Product Data Quality (PQD), Model-Based Enterprise (MBE)

Acronyms

3D three-dimensional. 1–4, 6–9, 14

ANSI American National Standards Institute. 7
AP Application Protocol. 6
API application programming interface. 2

CAD computer-aided design. 3, 4, 7–9, 14
CAM computer-aided manufacturing. 6, 7
CMS coordinate-measurement system. 6

*Corresponding Author, tdh1@nist.gov

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

DMC Digital Manufacturing Certificate. 2, 11, 12, 21
DMSC Dimensional Metrology Standards Consortium. 7

FAA Federal Aviation Administration. 3, 4

GD&T geometric dimensions and tolerances. 6

HTTPS Hypertext Transfer Protocol over Secure Sockets Layer. 9

IP intellectual property. 9
ISO International Standards Organization. 8
ITU-T Telecommunication Standardization Sector of the International Telecommunication Union. 1, 8

MBD model-based definition. 2, 6, 7, 14
MBE model-based enterprise. 2, 4, 6
MBM model-based manufacturing. 6

NC numerical control. 7
NIST National Institute of Standards and Technology. 2

OASIS Organization for the Advancement of Structured Information Standards. 8
OEM original equipment manufacturer. 4

PDF Portable Document Format. 7
PDM product-data management. 21
PDQ product-data quality. 4–6, 10, 13, 14, 19
PLM product lifecycle management. 1, 3
PMI product and manufacturing information. 6, 16
PRC Product Representation Compact. 7

QIF Quality Information Framework. 7, 11, 12, 14, 17–19, 21

S/MIME Secure/Multipurpose Internet Mail Extensions. 8
SaaS software-as-a-service. 9, 10
SAML Security Assertion Markup Language. 8
SFTP Secure File Transfer Protocol. 9
SOA service-oriented architecture. 8
SSL Secure Sockets Layer. 8
STEP Standard for the Exchange of Product Model Data. 6
STEP AP242 Standard for the Exchange of Product Model Data Application Protocol 242. 6, 7

TLS Transport Layer Security. 8

V&V verification and validation. 5, 6

WSN Wirth Syntax Notation. 15

X.509-PKI Public Key Infrastructure. 3, 8, 10, 21
X.509-PMI Privilege Management Infrastructure. 8
XML Extensible Markup Language. 7, 12, 17, 18
XSD XML Schema Definitions. 7

1 Introduction

Information technology advances such as big data, service-oriented architectures, and networking have triggered a digital revolution [1] that holds promise of reduced costs, improved productivity, and higher quality. Modern manufacturing enterprises are both more globally distributed and more digital than ever before, resulting in increasingly complex manufacturing system networks [2, 3]. Manufacturers are under mounting pressure to perform digital manufacturing more efficiently

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

and effectively within these distributed manufacturing systems. Moreover, engineers are being pushed by industry and business demands to use more manufacturing information and knowledge in their design decisions [4]. To do so, industry is changing how product definitions are communicated – from paper to models.

Those leading the efforts to transition communication methods for manufacturing complex products coined the term "digital thread" to convey the data flows between engineering, manufacturing, business processes, and across supply chains [5]. With the advent of new manufacturing-data standards [6] and more powerful engineering software, it is now possible to perform all engineering functions using a model-based definition (MBD) [7]. A MBD is a three-dimensional (3D) digital product model that defines the requirements and specifications of the product. A model-based enterprise (MBE) approach uses these models, rather than documents, as the data source for all engineering activities throughout the product lifecycle. The core MBE tenets are that models are used to drive all aspects of the product lifecycle and that data is created once and reused by all downstream data consumers.

This transition to a MBE has introduced new requirements on data usage across the product lifecycle. The need for automated methods to collect, transmit, analyze, and act on the most appropriate data is gaining attention in the literature [8, 9, 10, 11]. Research in model-based-data interoperability between design activities (e.g., product and assembly design) and manufacturing activities (e.g., fabrication, assembly, and quality assurance) is also gaining momentum [12]. However, more effort is needed in the area related to trustworthiness to support authentication, authorization, and traceability of product data. Product data must be guaranteed by an authority to a predefined level of data quality and trustworthiness if that information is to be used throughout the product lifecycle. That is, the user must be able to know who did what to whom and when it was done.

We developed a method and technology to support authentication, authorization, and traceability of product data. This technology enables trust throughout the product lifecycle. We do not define requirements of trustworthiness, because that work is happening in other places[1]. We aim to supplement the requirements work by providing the infrastructure for transmitting the information (e.g., provenance, metadata) required to enable trustworthiness in the product lifecycle.

Our methodology and technology follows recommended practices from Semantic Web [13] concepts using the X.509 standard [14]. This standard enables us to embed digital certificates with authentication, authorization, and traceability meta-data into 3D models. We developed an open-source Digital Manufacturing Certificate (DMC) toolkit[2] that provides an application programming interface (API) and user interface to embed digital certificates into four standards-based 3D-model formats. While X.509 has been adopted heavily by the cyber-security domain, we are not trying ot provide security methods. Our goal is to provide a mechanism for a data user to know what the data is (i.e., the authentication), how the data can be used (i.e., the authorization), and what has happen to the data throughout the product lifecycle (i.e., the traceability).

This paper describes the use of X.509 certificates and proposes a solution for embedding X.509 digital certificates in 3D models for authentication, authorization, and traceability of product data. This paper also describes the application of this technology to an Aerospace part. Finally, the paper draws conclusions, provides recommendations, and details our

---

[1]The National Institute of Standards and Technology (NIST) Cyber-Physical Systems Public Working Group is working on trustworthiness requirements and frameworks. For more information about the Cyber-Physical Systems Public Working Group goto: `https://pages.nist.gov/cpspwg/`

[2]The toolkit is available at: `https://github.com/usnistgov/DT4SM`

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

next steps for further research into using X.509 certificates in product lifecycle management (PLM) workflows to enable trustworthiness throughout the product lifecycle.

2 Background and Motivation

2.1 Data Authentication, Authorization, Traceability

In the regulated U.S. aerospace industry, the Federal Aviation Administration (FAA) requires that aerospace manufacturers to define a plan and receive FAA approval for managing and maintaining electronic design data (e.g., 3D computer-aided design (CAD) models, digital parts lists) used in the certification process [15]. Then, a parts manufacturer must be able to, "[determine] the quality, eligibility, and traceability of aeronautical parts and materials intended for installation on U.S. type-certificated products and articles," to ensure compliance with applicable regulations [16]. This requires the manufacturer to know the correct type-certificated design data and if that data was used during production. Accomplishing this task is easier said than done. Today, the traceability process is often done with significant human capital and minimal-to-no automation.

The literature we reviewed supports the FAA requirements. For instance, from the perspective of product design and manufacturing, traceability is defined as the ability to discover the history of decisions in the lifecycle, control the quality of data, products, and processes, and understand the relationship between assets [17, 18, 19, 20, 21]. Tracing dependency links between assets supports establishing relationships between those assets [18]. Understanding those links and relationships helps determine how decisions made during the creation and modification of assets affect related assets. Therefore, traceability may be considered a critical quality attribute intended to ensure system outputs conform to stakeholder requirements [19, 20].

Ouertani et al. [21] suggests the following questions must be answered to support data traceability:

1. What product knowledge is created or represented?
2. Who are the actors playing different roles in creating, using, or modifying product knowledge?
3. Where is the product knowledge created and located?
4. How is the product knowledge being created or modified?
5. Why was certain product knowledge created or modified?
6. When was the product knowledge created or modified?

Therefore, data traceability cannot be separated from authentication and authorization. Authentication is the act of determining that an entity (e.g., person, data) is as the entity is declared. For example, Public Key Infrastructure (X.509-PKI) is often used to guarantee a user is authentic. In contrast, authorization is the process of determining what permissions an entity is granted by a trusted source. For example, authorization methods could define how data can be used in a defined process. In manufacturing, contracts between organizations typically define what data is declared to be and how to confirm the data declarations (i.e., authentication). However, authorization requirements are not negotiated typically such that a data user could know how data should be used during a prototype versus a production run.

Ensuring complete data integration of authentication, authorization, and traceability is important to manufacturing in-

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

dustries. Those organizations must be able to determine data declarations, who did what to the data, when they did it, and potentially why it was done. Both regulated and non-regulated industries need effective and efficient processes for data authentication, authorization, and traceability. Regulated industries (e.g., aerospace, automotive, medical) focus significant resources on data authentication, authorization, and traceability to ensure they comply with the appropriate public-safety oversight. Manufacturers in both regulated and non-regulated industries care about data authentication, authorization, and traceability to reduce product-liability exposure within their supply chains and in the public realm.

The cost of achieving data authorization and traceability is thought to outweigh the benefits in paper-based systems [17]. As far back as 2006, reports showed major original equipment manufacturers (OEMs) were outsourcing 60 percent to 80 percent of their manufacturing [22]. Today, the majority of OEMs are manufacturing even less product in-house – relying more on their external supply chains. For example, the Boeing 787 (Dreamliner) has 30 tier-one suppliers, which in turn contract to hundreds of tier-two and tier-three suppliers [23]. Aside from the communication challenges that come with drawing-based systems, tracing what data is being used by whom and for what purpose is costly and inefficient for the Boeing 787 program. Moreover, knowing and ensuring that the data being used is the actual FAA-approved data is a real problem. This is why Boeing made the decision to switch to a MBE to define and certify the aircraft using only 3D CAD models. However, 3D CAD models still lack commercial-off-the-shelf support for authentication, authorization, and traceability. This is the motivation for our research into using embedded X.509 digital certificates in 3D CAD models for authentication, authorization, and traceability of product data.

2.2 Product Data Quality

Product-data quality (PDQ) must be a crucial focus to ensure successful data authentication, authorization, and traceability. Product data represents product-related specifications and is typically defined using a CAD system [24]. There are two uses of product data: (1) lateral direction and (2) vertical direction [24]. Lateral direction means using product data within a phase of the product lifecycle. Vertical direction means reusing product data in subsequent product lifecycle phases. PDQ is important to both uses.

Estimates show a significant number of engineering change orders and CAD re-modeling hours are the result of error, ambiguity, and data that is unusable by downstream applications [25, 22]. A manual healing process is used typically to reach the intended quality level. The two types of healing are repair (e.g., partial restoration for improving invalid data) and rework (e.g., disposing of and remaking the whole data set) [24]. Historically, the use of computer-aided systems to represent products promised effective and efficient communication of product data across the product lifecycle. Figure 1 shows the various data formats used during product-data exchange in the product lifecycle. However, promised benefits have not been achieved fully due to quality, technology, and cost limitations.

Data-interoperability formats (e.g., JT [26], PDF/PRC [27], STEP AP242 [6]) address the technology and cost factors. There is also commercial support for quality, but quality is less understood by industry than data interoperability. Again, if the translation or transfer of product data induces an error, then the product data becomes unusable for interfacing between various applications in the product lifecycle. Verifying PDQ at the point of creation ensures a known level of quality. Then,

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

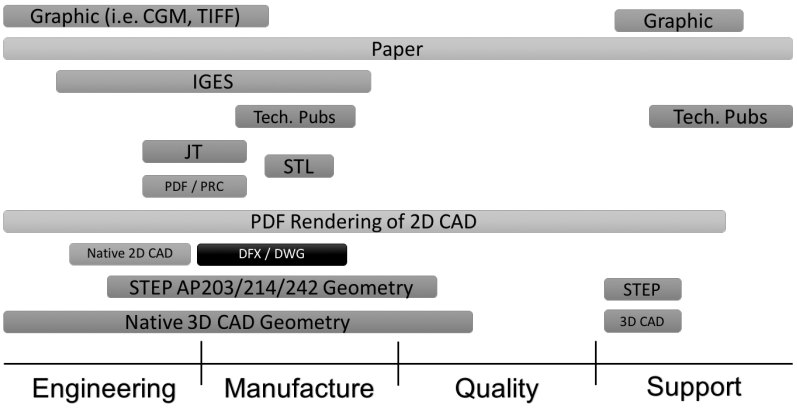

Fig. 1. Landscape of data formats used for product-data exchange

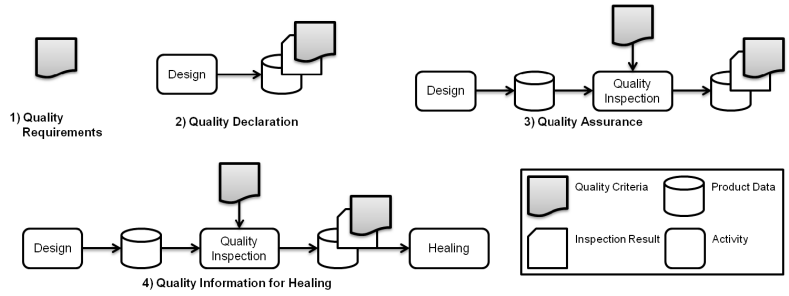

Fig. 2. PDQ information usage scenarios (from [24])

PDQ can be validated and traced throughout the product lifecycle to safeguard the data as it is exchanged, translated, inferred, and augmented. This approach enables a strong industrial verification and validation (V&V) strategy.

Kikuchi et al. [24] suggests a set of scenarios for PDQ information use (see Figure 2). We suggest re-purposing the Kikuchi et al. [24] scenarios as a PDQ workflow. The workflow would be four steps: (1) define PDQ requirements, (2) declare the PDQ level, (3) conduct PDQ assurance, and (4) report PDQ information.

PDQ requirements are defined PDQ criteria related to the tolerance and accuracy of product data. The PDQ requirements should also detail the V&V-diagnostic-algorithm needs. The requirements must be defined and communicated independently of the product data to ensure they are unbiased requirements. This mitigates risk of data-quality defects and economic loss from any required repair and rework. The requirements must be easily extensible to support a wide range of product types. Various standards [28, 29, 30, 31] exist that define quality requirements. Such are generally geared to an industry sector (e.g, automotive, aerospace), but are a good source for unbiased PDQ requirements.

PDQ declaration is information attached to product data that declares the PDQ level the product data satisfies. The product-data creator would declare the PDQ level. The declaration identifies what PDQ requirements are used during the V&V process. The PDQ information would also be transferred together with the product data to receiving systems. An example of PDQ level is the three technical-data-package levels (i.e., conceptual, developmental, and production) defined in MIL-STD-31000 Revision A [31]. These levels would align with the PDQ requirements defined for each level to ensure the

product data is as it is declared and satisfies usage expectations.

PDQ assurance is a diagnostic test of the product data against specified PDQ requirements. The PDQ information related to quality assurance would ensure the PDQ requirements from the PDQ declaration are satisfied. The PDQ workflow step would conduct quality activities in the cyber-space similarly to the way industry conducts quality activities on physical products. The quality-assurance information would also be transferred with the product data. This supports authentication and authorization of the product data throughout the product lifecycle.

PDQ information reporting is the reporting of the PDQ results from the PDQ assurance step. If defects are discovered during diagnostic testing, the level of defect severity and any healing methods used to correct defects would also be reported. This PDQ information is used to present what quality defects were detected, the exact location of defects in the target product data, and the seriousness of the defects. The PDQ information reporting should contain information about the product data, a link to the PDQ requirements, a description of the diagnostics algorithms, and the defect information (e.g., error location, type, severity).

This workflow would run in support of the V&V strategy previously discussed. The workflow could be run during product-data creation to verify the data meets PDQ requirements. Then, the workflow could be run after exchanging or translating the product-data, to ensure the output conforms to both the input and PDQ requirements. Thus, every stage of the product lifecycle may confidently take full advantage of interfacing with the product data with traceable PDQ information. Consequently, reuse of product data significantly reduces cost, risk, and cycle time while increasing product quality [7].

2.3 Open-data Formats

2.3.1 ISO 10303-242 (STEP AP242)

The standard, ISO 10303-242:2014 [6] titled "Managed Model Based 3D Engineering" or known commonly as Standard for the Exchange of Product Model Data Application Protocol 242 (STEP AP242). Barnard Feeney et al. [5] says, "The intent of STEP AP242 is to support a manufacturing enterprise with a range of standardized information models that flow through a long and wide 'digital thread' that makes the manufacturing systems in the enterprise smart." Digital data plays a central role in achieving the intent of STEP AP242.

Published in December 2014, STEP AP242 contains extensions and significant updates to other Standard for the Exchange of Product Model Data (STEP) Application Protocols (APs) for product and manufacturing information (PMI), kinematics, and tessellation [12]. PMI is the presentation and representation of geometric dimensions and tolerances (GD&T), material specifications, component lists, process specifications, and inspection requirements within a 3D product definition [4]. Hedberg Jr et al. [4] says, "PMI has the potential to make many lifecycle processes run faster, with fewer errors, and at lower cost." That's because STEP AP242 offers standards-based models that include the representation of PMI that is computer interpretable [5]. This is a major breakthrough that supports manufacturings need for model-based computer-aided manufacturing (CAM) and coordinate-measurement system (CMS) processes because STEP AP242 increases the effectiveness of MBE by enabling a common path for MBD and model-based manufacturing (MBM) integration [32, 12]. For the

concept proposed in this paper, STEP AP242 comes from design and represents the edict for the product definition.

2.3.2 ISO 6983 (G-CODE)

ISO 6983-1:2009 [33] is an international standard that defines the data format to program position, line motion, and contouring control systems in the numerical control (NC) of machines. This data format is commonly known as G-code. G-code was created at MIT in the late 1950s and, like CAD, rose in popularity through the 1970s [34]. Today, G-code is the near-universal format for programming computer-based NC machines.

G-code is generated typically from a manufacturing plan using a CAM system. G-code files are defined using a standardized ASCII-based set of commands. Each line of the G-code is a new command to the machine. Header information is standardized to support some traceability. For the concept proposed in this paper, additional header information and metadata is added to the G-code to support linking the G-code back to both STEP and CAM data.

2.3.3 ANSI/DMSC Quality Information Framework

The Quality Information Framework (QIF) [35] is an American National Standards Institute (ANSI) standard sponsored by the Dimensional Metrology Standards Consortium (DMSC) that defines an integrated set of Extensible Markup Language (XML) information models to enable the effective exchange of metrology data throughout the entire metrology process. QIF handles feature-based dimensional metrology, quality measurement planning, first article inspection, and discrete quality measurement. QIF supports importing the product definition and reusing data for inspection planning, execution, analysis, and reporting.

QIF uses terminology and semantics from the inspection world to represent the various elements in the QIF specification. The QIF information models are normalized in XML Schema Definitions (XSD). The QIF XSDs are organized into six application areas for metrology: (1) MBD, (2) Rules, (3) Resources, (4) Plans, (5) Results, (6) Statistics. The MBD (containing the product definition) is combined with measurement rules and resources definitions to generate a plan. The plan is then executed and the results are captured. Multiple results are combined to generate statistics. QIF is an information model and format that can be exported from commercial metrology applications available in the marketplace. While, QIF does not perform the task of statistics and the other metrology methods, QIF does enable the ability to put raw inspection data into a quality context that is computer-processable. For the concept proposed in this paper, QIF documents are generated from the STEP AP242 file and linked back to both STEP AP242 and G-code data.

2.3.4 ISO 32000 and ISO 14739 (PDF/PRC)

The Portable Document Format (PDF) [36] and Product Representation Compact (PRC) [27] international standards are often combined into technologies for visualizing product-definition data. PDF/PRC is often referred to as a 3D PDF. While there are several "flavors" of 3D PDF, the combination of PRC embedded in a PDF document is emerging as the industry recommended practice. PDF/PRC enables the display of 3D product definition in any PDF reading software that conforms to the standard. Using PDF/PRC enables effective and efficient visualization of product data throughout the lifecycle for

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

human-consumption. For the concept proposed in this paper, PDF documents are generated from native CAD data and linked to the native data.

2.4 X.509 Certificates

The X.509 standard [14], titled *Information Technology – Open Systems Interconnection – The Directory – Part 8: Public-key and Attribute Certificate Frameworks*, was first published in 1988. The Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) developed the standard, first as a recommendation, intended as the authentication framework for the X.500 series of electronic directory services. The latest version of X.509 was published in 2014 by the International Standards Organization (ISO) under standard number ISO/IEC 9594-8:2014 [14].

The X.509 standard normalized two concepts for authentication and authorization. The first is X.509-PKI [37] and Privilege Management Infrastructure (X.509-PMI) [38]. X.509-PKI addresses authentication and X.509-PMI addresses authorization.

Figure 3 displays the basic components of X.509-PKI (3a) and X.509-PMI (3b). The purpose of X.509-PKI is to create and manage digital certificates – primarily for authentication with a certificate authority at the top of a certificate hierarchy. The hierarchy consists of hardware, software, people, policies, and procedures [39]. Common implementations of X.509-PKI today use asymmetric (public) key encryption, where a user is issued both a private key that is only known to the user and a public key that is known to everyone [39]. X.509-PKI is the most familiar certificate infrastructure used by end-users.

X.509-PMI is less known to end-users. X.509-PMI is similar to X.509-PKI, except X.509-PMI is used for authorization. The purpose of X.509-PMI is to manage user authorizations with an attribute authority at the top of a certificate hierarchy [39]. The attribute authority references an X.509-PKI identity and delegates privileges to the identity based on the assigned privileges from a "source of authority." The attribute authority issues an "attribute certificate" that is linked to the identity provided by the X.509-PKI-based certificate. Adoption of the X.509-PMI in practice has been minimal with only a few commercially available applications.

In practice, X.509-PKI is implemented significantly more than X.509-PMI. X.509-PKI enjoys a broad range of applications – most notably Secure Sockets Layer (SSL)/Transport Layer Security (TLS) encryption of websites and Secure/Multipurpose Internet Mail Extensions (S/MIME) signing/encrypting of emails. However, X.509-PMI has seen minimal-to-no commercial adoption since its introduction to the X.509 standard in 2001. This is, in part, due to the rise of service-oriented architectures (SOAs) and attribute assertions via the Security Assertion Markup Language (SAML) specification [40] developed by Organization for the Advancement of Structured Information Standards (OASIS) [41].

X.509-PKI can be extended to include authorization information by embedding additional metadata in signatures to describe privileges. Our research uses X.509-PKI primarily and includes additional privilege metadata to manage authorization requirements. Taking this approach enables us to simplify the implementation of X.509 constructs while introducing traceability, authentication, and authorization to 3D CAD models.

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

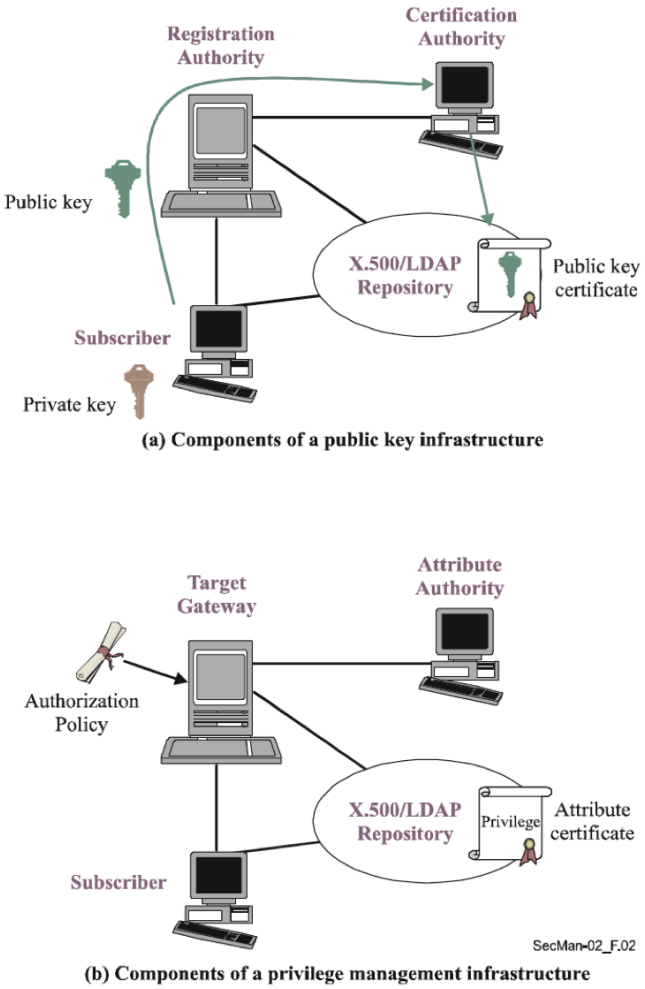

Fig. 3. X.509 components of public key infrastructure and privilege management infrastructure (from [39])

2.5 Alternative Technologies

Steve Jobs said [42], "Simplicity is the ultimate sophistication." Our goal was to make implementation and use for the end user as simple as possible to alleviate the need for understanding complex interactions between various actors, systems, and technologies. We investigated two alternative solutions in addition to the solution we present in this paper. Our solution utilizes X.509-based digital certificates embedded in 3D CAD models for the purposes of authentication, authorization, and traceability. In addition to the solution we describe in this paper, we investigated two alternative solutions, which were:

1. Brokered data-exchange mechanisms
2. Cloud-based and software-as-a-service (SaaS) product-data repositories

Brokered data-exchange mechanisms, such as Secure File Transfer Protocol (SFTP) and Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS) portals, have been a long-term solution for industry. Brokered data-exchange mechanisms are based on stable technology that has been in existence for decades. These types of mechanisms are usually continuously available and support on-demand access. They also support a simple distribution of data across the supply chain and can be centrally managed. However, brokered data-exchange mechanisms lack support for authentication, authorization,

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

and traceability of product data unless metadata is added explicitly to the data files stored within the system. Brokered data-exchange mechanisms also require a large user, data, security, and intellectual property (IP) management overhead to ensure users can access only the data each is authorized to use. Lastly, with brokered data-exchange mechanisms, there is no control of the product data once the data is downloaded. Overall, data-exchange processes are manual and require a large management overhead, which makes brokered data-exchange mechanisms a poor choice for authentication, authorization, and traceability.

Cloud-based and SaaS product-data repositories are the newest alternative that we investigated. This alternative is implemented with the use of a proprietary application that is installed on client systems and interfaces with a centralized data repository to control the usage of product data. The cloud-based and SaaS solutions provide direct support of data authorization; they also provide continuously available data repositories and support on-demand access. These systems typically wrap product data in a proprietary-format container to keep end-users from accessing the product data without the required proprietary application being installed on client systems. Cloud-based and SaaS solutions require constant connections between client and server systems to ensure the product data is accessible. This means systems must remain "on-line" at all times, which adds a layer of unneeded complexity to interfacing with the product data. In addition, cloud-based and SaaS solutions lack standards support because the technologies are still emerging and do not have wide-spread industrial support. The lack of standards limits the industrial scalability of the cloud-based and SaaS solutions.

The shift to distributed manufacturing in retail and commercial industries is reducing traceability of product requirements. Geographically decentralizing manufacturing assets results potentially in product data being dispersed across an entire network of internal and external suppliers. In this case authentication, authorization, and traceability are imperative to ensure the right data is used at the right time. Further, the globalization and commoditization of manufacturing in the aerospace, automotive, medical, and similar industries is increasing both regulatory and data-management burdens. The best solution supports authentication, authorization, and traceability without adding to the existing burdens.

We chose to use X.509-based digital certificates to implement our strategy for authentication, authorization, and traceability because X.509 puts forth a simpler solution than the alternatives. This simpler approach supports wider opportunities for industrial adoption and scalability. In addition, our solution is enabled by the widely adopted X.509 standard.

3 Implementation Description

Digital certification could help to control and improve the PDQ – the digital backbone of smart manufacturing – throughout the product lifecycle. Smart manufacturing is a recent concept whose requirements and possibilities have not all been explored and standard information frameworks for product data do not currently cover all of them. We choose to use the X.509-PKI infrastructure because it is widely adopted across several industries and it is efficient to implement. We are digitally signing the data – not encrypting the data. We took this approach because we are not trying to provide security mechanisms. There are various activities in industry that are focused on the topic of cyber-security [43] – we don't want to duplicate those efforts. In addition, encryption only provides security at the edge of the communication. Once the data is decrypted for usage, there is limited-to-no way to control how that data usage continues on in the lifecycle. Recall, our goal

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

was to provide a vehicle for transmitting the required information to support the process of trustworthiness in the product lifecycle. Digital signatures conforming to X.509-PKI provide a method for including traceability information in a sustainable way. If a user modifies the data, either intentionally or unintentionally, the digital certificate would become invalid. This provides a level of control and data guarantees that ensure the data user knows who did what to the data and when it was done.

Traceability may be classified across three categories: i) internal/external, ii) forward/backwards, iii) active/passive. Cheng and Simmons [44] defines the first category as both internal and external traceability. Internal traceability is the traceability inside the factory and the product system. External traceability follows the product into its relationships with customers, maintainers, and service providers. The next category comes from Jansen-Vullers et al. [45], where traceability is classified into backward and forward traceability. Backward traceability records information and data on the past history of the product. Forward traceability explains what will happen to a certain product, in terms of operations and processes – this information is written before performing any operation. The last category is active and passive traceability. Active traceability is considered to be on-line and synchronous, which implies the data may be "phoning home" to a central server. Passive traceability may be on-line or off-line and is typically asynchronous. Our method supports all combinations of the traceability categories. The only requirement to using our method for traceability is the availability to validate the attached digital certificates.

In the remainder of Section 3, we describe and discuss our solution. First, we define the signature block for capturing the digital signature using a digital certificate. Next, we present a short use case to provide context to implementing our solution. Then, we describe our proposal for extending both the ISO 10303-21 (STEP Part 21) and QIF standards to support our solution. Lastly, we provide an aerospace example to demonstrate the usage of the authentication, authorization, and traceability information.

3.1 Signature Block

The DMC toolkit is designed specifically to provide certification – digital signature using software and hardware certificates, and verification – of manufacturing-related data. It enables generating, embedding, and verifying signatures in a manufacturing-related file.

ISO 10303 defines different serialization mechanisms to encode product data in ASCII files, such as ISO 10303-21 [46]. The DMC toolkit follows the work from the ISO Technical Committee 184 / Subcommittee 4 on the Draft International Standard (DIS) 10303-21 3rd edition. This draft recommends to embed the signature, following the PKCS#7 format, at the end the data file, in a signature data block. A signature data block opens with a `SIGNATURE;` tag and closes with an `ENDSEC;` tag. A valid signature block example is shown in Listing 1.

Listing 1. Signature block for STEP (DIS 10303-21 ed. 3)

```
1 SIGNATURE;
2 -----BEGIN PKCS7-----
3 signature in pkcs7 format
```

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

```
-----END PKCS7-----
ENDSEC;
```

ISO 6983 does not officially provide guidance on how to embed signatures in its data file. The DMC Toolkit implementation follows similar guidelines to ISO/DIS 10303-21 3rd edition. In the current version, digital signatures can be found at the end of the file as comments between the `(-----BEGIN PKCS7-----)` and `(-----END PKCS7-----)` tags. A valid signature block would look like the example shown in Listing 2.

Listing 2. Signature block for ISO 6983

```
(-----BEGIN PKCS7-----)
signature in pkcs7 format
(-----END PKCS7-----)
```

The DMC toolkit signs PDF [36] with embedded PRC [27] files using the iText PDF library[3]. This library follows the official ISO standard, providing interoperable signatures that can be read by any software component compatible with ISO 32000 [36].

The Quality Information Framework uses XML to represent its data. The W3C has a standard that describes digital-signature representation for XML data (XMLDsig)[4]. The DMC toolkit follows the XMLDsig specification to encode QIF signatures. QIF signatures are the last XML nodes of the root node of the QIF document. A valid signature example is shown in Listing 3.

Listing 3. Signature example in a QIF document

```
<?xml version="1.0" encoding="utf-8"?>
<QIFDocument ...>
...
<Signature xmlns="http://www.w3.org/2000/09/xmldsig\#">
<SignedInfo>
<CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
<SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig\#rsa-sha1" />
<Reference URI="">
<Transforms>
<Transform Algorithm="http://www.w3.org/2000/09/xmldsig\#enveloped-signature" />
</Transforms><DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig\#sha1" />
<DigestValue>...</DigestValue>
</Reference>
</SignedInfo>
```

[3] http://sourceforge.net/projects/itext/
[4] https://www.w3.org/TR/xmldsig-core/

```
<SignatureValue>
...
</SignatureValue>
<KeyInfo>
<X509Data>
<X509Certificate>
...
</X509Certificate>
</X509Data>
</KeyInfo>
</Signature>
</QIFDocument>
```

3.2 Use Case

Our goal is to demonstrate the benefits of digitally signing product data to support and improve data flows and quality control in a smart-manufacturing environment. Our use case focuses on manufacturing data elements transformations. Smart manufacturing requires digital product information to be available to each of its processes. The model-based paradigm on which smart manufacturing relies often needs product information to be expressed in different formats and processed by different software through the product lifecycle. This results in manipulating and generating variations of master-product models. These variations come from different transformations, either from the master models themselves or from other variations. These variations constitute what we identify as a transformation network, as seen in Figure 4.

A transformation network can be represented as a directed graph where nodes represent data and directed edges represent transformations. The head of a directed edge is the result of the transformation; the tail is the source to which the transformation is applied. Because of the significant number of transformations and variations during the product lifecycle, it is crucial to know who did what to whom and when. Our objective is to leverage X.509 certificates and digital signatures to embed and secure traceability information into the product models. This traceability information is crucial in a space where data is created and used on different platforms and in different locations by different users. Embedding and signing such information increases product data trustworthiness and enables reliable PDQ control. PDQ controls can take different forms. Quality control ensures data consistency, helps in troubleshooting data loss, identifies defective data processing, and improves the overall quality of data flows through the product lifecycle.

We identified four types of relevant traceability information: i) who: identifies the system or person responsible for the generation of the attached dataset; ii) what: describes the type of transformation that has been applied to a source to generate the attached dataset; iii) whom: describes the source data to which the transformation was applied to generate the attached dataset; and iv) when: time at which the transformation was recorded. Put back in the context of the transformation network, for any node at the head of a directed edge, we embed the edge and its tail, the creation time, and the creator of the edge.

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

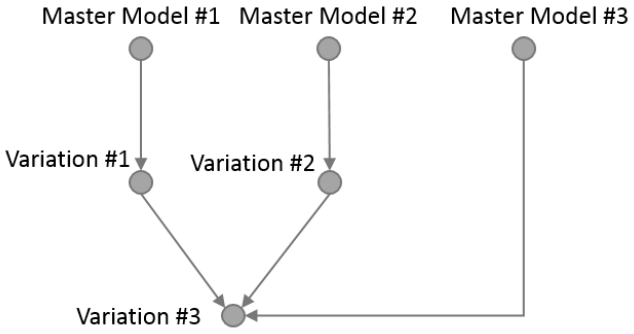

Fig. 4. Transformation network

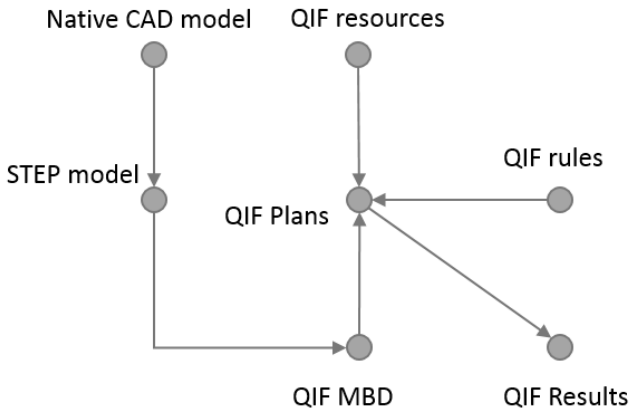

Fig. 5. Example of a transformation network

Traceability information quality is vital to enable reliable PDQ control and requires consistency and accuracy. While the first, third, and fourth (who, whom, when) data fields are unambiguous, the second one (what) can be a source of ambiguity and generate inaccurate and ambiguous traceability information. To mitigate the possible ambiguity, we identified the three most common transformations that occur during the product lifecycle: i) translation: happens when the same information is reproduced in a different representation/file format; ii) inference: happens when computation or reasoning is applied on existing data to validate it or infer a new one; iii) augmentation: happens when an inference is run over a set of data and the result is added to the original set of data.

To illustrate the notion of transformation network we built a trivial example (Figure 5) commonly found in a smart manufacturing environment. We translated a native 3D CAD model into a signed STEP AP242 model. That model was translated and augmented into a signed QIF MBD model. This new model, together with a list of metrology resources (QIF resources) and metrology knowledge (QIF rules), is reasoned on to generate a list of QIF measurement plans.

3.3 Extending ISO 10303-21 to support transformation network and multi-path hierarchical signings

Despite its current extension to support digital signatures, STEP Part 21 (10303-21 edition 3) cannot embed and sign traceability information such as the ones we mentioned in Section 3.2 . We described in the previous section how ISO 10303-21 edition 3 intends to embed digital signatures into STEP Part 21 files. Our main goal is to enrich digital signatures

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

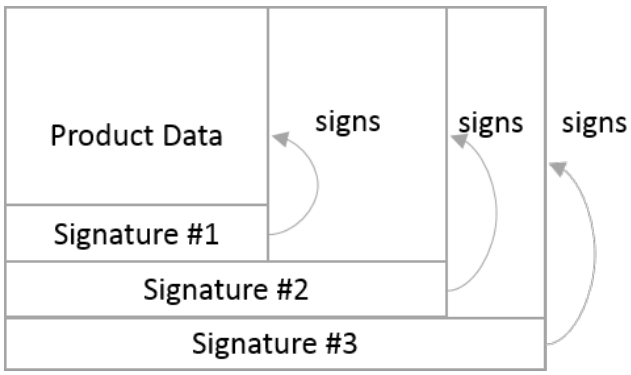

Fig. 6. Multiple signatures support in STEP 10303-21 edition 3

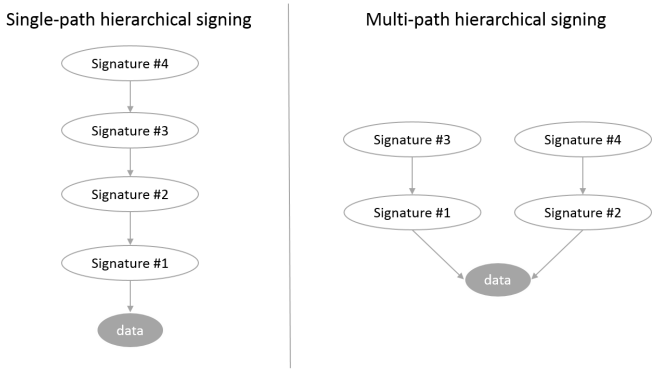

Fig. 7. Single-path vs multi-path hierarchical signing

with meta-data about the provenance of the file content. We also use this opportunity to improve the multiple-signatures support in STEP. In Part21 edition 3, each signature section contains a digital signature that vouches for all the bytes before the section, whether those bytes are product data by itself or product data and other signatures. As shown in Figure 6, the signature mechanism requires Signature #2 to vouch for Signature #1, and Signature #3 to vouch for #2 and #1 combined. This mandatory vouching can raise legal issues and constraints in a field where product models and data must be legally signed and endorsed by a multitude of different actors from different organizations, along the product lifecycle.

ISO 10303-21 edition 3, in its current form, does not adequately support the business need for digital signature in an environment where it is a legal matter for companies to make sure their signatures are properly used and endorse the right content. Our goal is to enhance ISO 10303-21 to support multi-path hierarchical signing as shown in Figure 7. In a multi-path hierarchical signing, a new signature does not necessarily have to vouch for the latest signature in the file. This allows multiple organizations to sign on a same file while only vouching for signatures issued from their own organizations.

These enhancements to ISO 10303-21 require us to extend its syntax to support the following two requirements:

- A signature block must enable a signer to vouch for other existing signatures to support multi-path hierarchical signings
- A signature block must enable a signer to attach a set of metadata to document the signature and represent the file transformation information

ISO 10303-21 uses a Wirth Syntax Notation (WSN) [47] meta-syntax to define its syntax. A meta-syntax, or grammar, is

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

a set of rules that describe and constrain a domain specific language and its valid syntax and vocabulary. Our WSN extension to the current WSN grammar is shown in Listing 4.

Listing 4. WSN extension to current WSN grammar

```
(1.1) SIGNATURE_SECTION = ``TRACE:''ENTITY_INSTANCE_NAME
TRACE
SIGNATURE
``ENDSEC;''.
(1.2) TRACE = ENTITY_INSTANCE_NAME ``=PKCS_TRACE(`` METADATA'')''.
(1.3) METADATA = ``{`` LIST_OF_FIELDS ''}''.
(1.4) LIST_OF_FIELDS = FIELD { ``,'' FIELD}.
(1.5) FIELD = FIELD_NAME``:''FIELD_VALUE.
(1.6) FIELD_VALUE = ``'''' STRING ``''''.
(1.7) FIELD_NAME = STRING.
(1.8) SIGNATURE = ENTITY_INSTANCE_NAME ''=``PKCS_TOKEN .
(1.9) PKCS_TOKEN = ``PKCS(`` PKCS_SIGNATURE '',`` CROSS_BOOL '',``
    CROSS_INDEX'')''.
(1.10) PKCS_SIGNATURE= `` ' ''PKCS7`` ' ''.
(1.11) PKCS7 = STRING.
(1.12) CROSS_INDEX = ``[``LIST_OF_TRACE_IDS'']''
(1.13) LIST_OF_TRACE_IDS = ENTITY_INSTANCE_NAME {``,'' ENTITY_INSTANCE_NAME}.
(1.14) CROSS_BOOL = ``Y''|``N''.
```

Every signature block (Listing 4, 1.1) starts with the keyword `TRACE` followed by a numerical identifier in the form of an `ENTITY_INSTANCE_NAME`, ends with the delimiter `ENDSEC;` and contains two elements, a `TRACE` (Listing 4, 1.2) and a `SIGNATURE` (Listing 4, 1.8). A `TRACE` records `METADATA` (Listing 4, 1.2 and 1.3), as a set of `FIELD`s (Listing 4, 1.5) representing transformation information in a simplified JavaScript Object Notation (JSON) like format. The `SIGNATURE` (Listing 4, 1.8) itself contains a `PKCS7` (Listing 4, 1.10 and 1.11) string that represents its `PKCS7` encoding. This string is followed by a field – `CROSS_BOOL` (Listing 4, 1.9 and 1.14) – that determines whether the signature vouches for others or not. The last field – `CROSS_INDEX` (Listing 4, 1.9 and Listing 4, 1.12) – records the list of signature block identifiers that the current signature block vouches for.

The code snippet in Listing 5 is an example of our extension in use. In this example, a STEP file is signed three times (Listing 5, 2.3, 2.8, and 2.12) and contains three signature blocks (Listing 5, 2.1, 2.6, and 2.10). The first signature asserts that the current file results from a translation from December 2015 (Listing 5, 2.2). The second signature (Listing 5, 2.8) on the file certifies the authenticity of the STEP file itself and validates (Listing 5, 2.7) the transformation it comes from (Listing 5, 2.2). The third signature (Listing 5, 2.12) signs the STEP file. We use the associated metadata (Listing 5, 2.11) to record that the signer only acknowledges that the PMI in the STEP file conforms to the PMI in the native file.

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

Listing 5. Signature and Traceability example in a STEP document

```
(2.1) TRACE:#122
(2.2) #123 = PKCS_TRACE({source:'c:\\file.native', date:'12-DEC-2015',
    operation:'translation'})
(2.3) #124 = PKCS('pkcs7_signature',N,[])
(2.4) ENDSEC;
(2.6) TRACE:#125
(2.7) #126 = PKCS_TRACE({source:'c:\\file.native', date:'12-DEC-2015',
    operation:'validation'})
(2.8) #127 = PKCS('pkcs7_signature',Y ,[#122])
(2.9) ENDSEC;
(2.10) TRACE:#128
(2.11) #129 = PKCS_TRACE({source:'c:\\file.native', date:'12-DEC-2015',
    operation:'validation', what:'PMI'})
(2.12) #130 = PKCS('pkcs7_signature',N,[])
(2.13) ENDSEC;
```

3.4 Extending QIF to support transformation network and multi-path hierarchical signings

QIF 2.1 XML implementation presents at the moment some of the same limitations as ISO 10303-21 edition 3. QIF does support multiple signatures, but its implementation is different from what Part 21 uses. It does not allow to attach data to signatures, which makes it impossible to record the metadata that represents the transformation information.

The current version of the standard implements multiple signatures in a way that each signature vouches for the XML document alone, not the existing signatures at the time of signing, unlike Part 21. We categorize this pattern as a multi-path flat signing strategy, shown in Figure 8, because of the lack of hierarchy between the signatures. In this section we discuss an extension to the QIF information model to support a multi-path hierarchical signing strategy as shown in Figure 7.

QIF architecture relies on a `QIFDocument` container element that contains other elements, each representing concrete information (statistics, measurement plans, ...). In Section 3, we discussed the `Signature` element owned by a `QIFDocument`. A summary of this architecture is shown in the UML class diagram in Figure 9. Not only does this architecture lack a placeholder for transformation and traceability metadata, it also requires any signature to vouch for all of the information contained in `QIFDocument`. We extended this architecture to support the same two requirements we previously defined for Part 21: i) implementation of a multi-path hierarchical signing strategy and ii) a means to attach metadata to a digital signature. We created a new type – `Trace` – to represent a signature block. This signature block has a unique identifier `ID`, and contains two elements. The first element, of type `PKCS_TRACE`, is defined in a simplified JSON-like format that contains a unique identifier – `PKCS_TRACE_ID` – and a string – `Metadata` – used to record the traceability information, attached to the signature in the current signature block. The second element – `PKCS`, with a unique identifier – `PKCS_ID`, contains the digital signature itself. The `Cross_bool` attribute indicates whether the current signature and

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

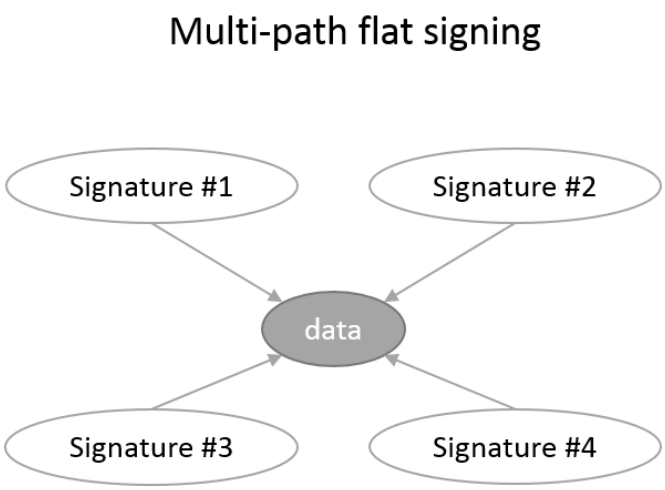

Fig. 8. Mutli-path flat signing

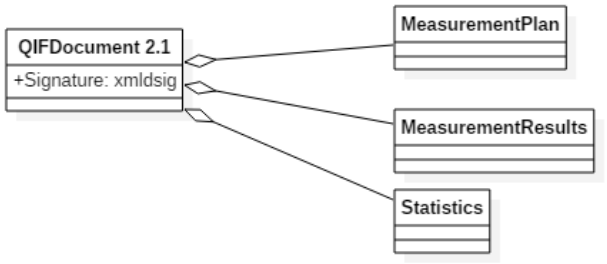

Fig. 9. Digital Signature implementing in QIF 2.1

signature vouches for other signature(s) or not. If so, the `Cross_index` attribute lists the `QPID`s of the signature block(s) vouched for by the current block. A summary of the extension is presented in the UML class diagram in Figure 10.

The listing below shows an example of a QIF XML document that contains statistical information. The statistical information element has been digitally signed, independently from the rest of the document.

Listing 6. Signature and Traceability example in a QIF document

```
<?xml version="1.0" encoding="UTF-8"?>

<QIFDocument>
<QIFStatistics>
...
<Trace ID="TID1NIST">
<PKCS_TRACE PKCS_TRACE_ID="PTI1NIST">
<Metadata></Metadata>
</PKCS_TRACE>
<PKCS PKCS_ID="PID1NIST" Cross_bool="false">
<Signature>
```

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

```
12  ...
13  </Signature>
14  </PKCS>
15  </Trace>
16  </QIFStatistics>
17  ...
18  </QIFDocument>
```

3.5 Example usage of authentication, authorization, and traceability information

Recall, we are adding authentication, authorization, and traceability information as meta-data attached to the digital signature. This allows us to make a declaration of the quality of the data in a workflow similar to the Kikuchi et al. [24] usage scenarios discussed in Section 2.2. We could also declare how the data may be used. For example, assume we have a pre-defined verification criteria for data that would be used in a development or prototyping workflow. We could embed information in digital certificates to show that product data meets the PDQ requirements for development workflows and that the product data can only be used in a development workflow.

Figure 11 shows the example authentication, authorization, and traceability certification process for a development aerospace part. The first step in the example is to declare that the product data is of a development type. Next, an in-

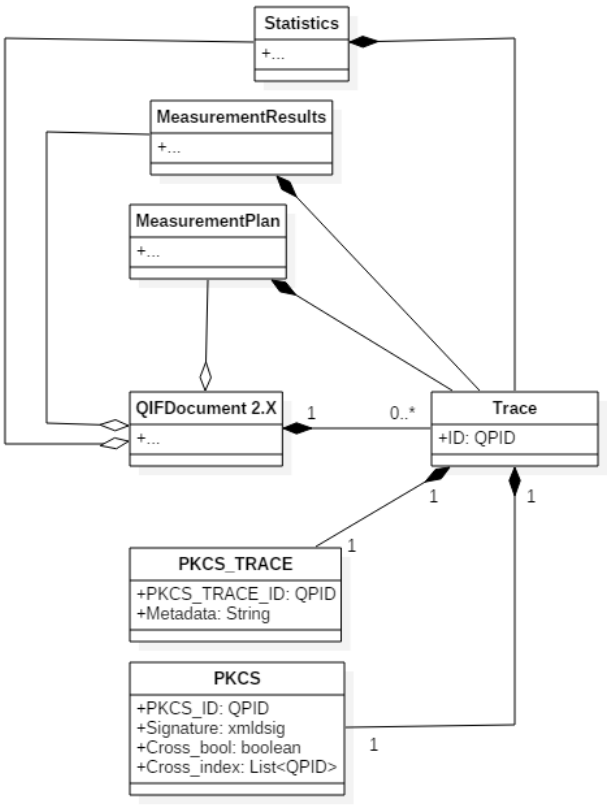

Fig. 10. QIF extension for multi-path signing strategy support

This is an Accepted Manuscript of an article published by ASME in Journal of Computing and Information Science in Engineering on 14 JUL 2016, available online: http://doi.org/10.1115/1.4034131.

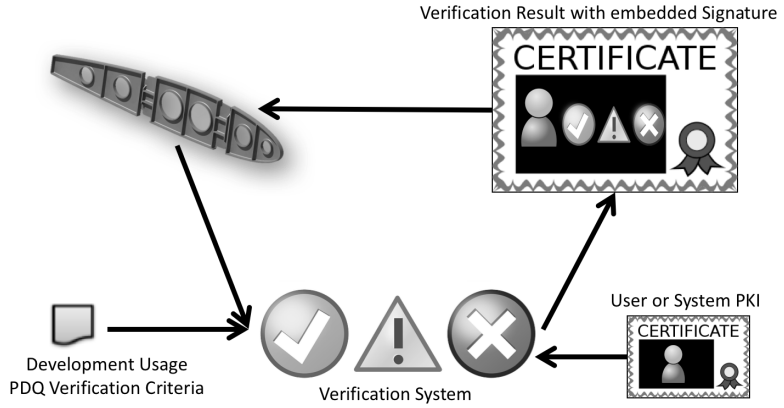

Fig. 11. Example process for verifying the quality of product data and embedding usage restrictions

dependently developed development-usage-PDQ-verification criteria is selected for checking that the product data is of a sufficient level to meet a development workflow's requirements. In this example, the aerospace product data is checked against PDQ criteria using a commercially available verification and validation system. The results of the verification (e.g., pass, warning, error) are captured and combined with the digital signature of the user running the verification check. The signature could also be from a system that is running the check autonomously. The resulting digital certificate containing the verification results, digital signature, and meta-data for authentication, authorization, and traceability is embedded in the aerospace part's product data. Listing 7 shows the authentication, authorization, and traceability information embedded in a STEP file translated and validated from the verified native product data.

Listing 7. Authentication, Authorization, and Traceability information in a STEP document for an aerospace part

```
1  TRACE:#3415
2  #3416 = PKCS_TRACE({source:'URI:15.1115\734.13.wingrib', date:'17-FEB-2016',
       operation:'verification', usage:'development', result:'pass with warnings',
       report:'URI:15.1115\734.13.wingrib.verification'})
3  #3417 = PKCS('pkcs7_signature',N,[])
4  ENDSEC;
5  TRACE:#3418
6  #3419 = PKCS_TRACE({source:'URI:15.1115\734.13.wingrib',
       destination:'C:\\wingrib.stp', date:'17-FEB-2016', operation:'translation'})
7  #3420 = PKCS('pkcs7_signature',Y,[#3415])
8  ENDSEC;
9  TRACE:#3421
10 #3422 = PKCS_TRACE({source:'c:\\wingrib.stp', date:'17-FEB-2016',
       operation:'validation', result:'pass', report:'c:\\wingrip-validation.pdf'})
11 #3423 = PKCS('pkcs7_signature',Y,[#3415,#3418])
12 ENDSEC;
```

4 Conclusion

In this paper we discussed how to leverage X.509-PKI-based digital certificates to restore trust in product data across the product lifecycle. We showed that digital signatures are a means of authenticating, authorizing, and tracing product data. Embedding authentication, authorization, and traceability information in the product data builds trust throughout the product lifecycle.

During our effort we analyzed, evaluated, and implemented different commonly used standardized product-data formats. Our use case highlighted gaps in the current version of these standards, gaps that we addressed for two of the formats: STEP and QIF. We are working with the appropriate standards-developing organizations to resolve the gaps and enhance each standard to fully support authentication, authorization, and traceability of product data.

Our future efforts will focus on identifying gaps in other common standard formats and integrating digital certification of product data with various enterprise workflows. At the same time we will develop recommendations for the appropriate metadata to embed with the certificates. We plan to integrate the DMC toolkit into a commercially available product-data management (PDM) tool to study automated processing of authentication, authorization, and traceability information for some of the most common enterprise workflows (e.g., engineering release, change management, manufacturing planning). We expect combining the DMC toolkit with automated enterprise workflow will significantly increase industry's confidence in product-data throughout the product lifecycle – such that industry can quickly understand who did what to whom and when it was done.

Acknowledgements

The authors wish to thank Allison Barnard Feeney, Mark Carlisle, and Vijay Srinivasan from NIST and the peer-reviewers for their comments and input to this paper.

Disclaimers

The work presented in this paper is an official contribution of the National Institute of Standards and Technology (NIST) and not subject to copyright in the United States. Certain commercial systems are identified in this paper. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the products identified are necessarily the best available for the purpose.

References

[1] Dazhong Wu, David W. Rosen, Lihui Wang, and Dirk Schaefer. Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation. Computer-Aided Design, 59(0):1–14, 2015. ISSN 0010-4485. doi: http://dx.doi.org/10.1016/j.cad.2014.07.006. URL `http://www.sciencedirect.com/science/article/pii/S0010448514001560`.

[2] Dazhong Wu, Matthew John Greer, David W. Rosen, and Dirk Schaefer. Cloud manufacturing: Strategic vision

and state-of-the-art. Journal of Manufacturing Systems, 32(4):564–579, 2013. ISSN 0278-6125. doi: http://dx.doi.org/10.1016/j.jmsy.2013.04.008. URL http://www.sciencedirect.com/science/article/pii/S0278612513000411.

[3] Xun Xu. From cloud computing to cloud manufacturing. Robotics and Computer-Integrated Manufacturing, 28 (1):75–86, 2012. ISSN 0736-5845. doi: http://dx.doi.org/10.1016/j.rcim.2011.07.002. URL http://www.sciencedirect.com/science/article/pii/S0736584511000949.

[4] Thomas Hedberg Jr, Nathan Hartman, Phil Rosche, and Kevin Fischer. A research strategy for using manufacturing knowledge earlier in the product lifecycle. International Journal of Production Research (Accepted), 2016.

[5] Allison Barnard Feeney, Simon P. Frechette, and Vijay Srinivasan. A portrait of an ISO STEP tolerancing standard as an enabler of smart manufacturing systems. Journal of Computing and Information Science in Engineering, 15(2):021001–021001, 2015. ISSN 1530-9827. doi: 10.1115/1.4029050. URL http://dx.doi.org/10.1115/1.4029050.

[6] International Standards Organization. Industrial automation systems and integration – product data representation and exchange – part 242: Application protocol: Managed model-based 3D engineering, 2014.

[7] Thomas D. Hedberg Jr, Joshua Lubell, Lyle Fischer, Larry Maggiano, and Allison Barnard Feeney. Testing the digital thread in support of model-based manufacturing and inspection. Journal of Computing and Information Science in Engineering (In Press), 2016. ISSN 1530-9827. doi: 10.1115/1.4032697. URL http://dx.doi.org/10.1115/1.4032697.

[8] Energetics Inc. Measurement science roadmap for prognostics and health management for smart manufacturing system. Report, National Institute of Standards and Technology, 2015.

[9] Moneer Helu and Thomas Hedberg Jr. Enabling smart manufacturing research and development using a product lifecycle test bed. Procedia Manufacturing, 1:86–97, 2015. ISSN 2351-9789. doi: http://dx.doi.org/10.1016/j.promfg.2015.09.066. URL http://www.sciencedirect.com/science/article/pii/S2351978915010665.

[10] R. Gao, L. Wang, R. Teti, D. Dornfeld, S. Kumara, M. Mori, and M. Helu. Cloud-enabled prognosis for manufacturing. CIRP Annals - Manufacturing Technology, 64(2):749–772, 2015. ISSN 0007-8506. doi: http://dx.doi.org/10.1016/j.cirp.2015.05.011. URL http://www.sciencedirect.com/science/article/pii/S000785061500150X.

[11] Min Li, Shuming Gao, and Charlie C. Wang. Real-time collaborative design with heterogeneous CAD systems based on neutral modeling commands. Journal of Computing and Information Science in Engineering, 7(2):113–125, 2006. ISSN 1530-9827. doi: 10.1115/1.2720880. URL http://dx.doi.org/10.1115/1.2720880.

[12] Asa Trainer, Thomas Hedberg Jr, Allison Barnard Feeney, Kevin Fischer, and Phil Rosche. Gaps analysis of integrating product design, manufacturing, and quality data in the supply chain using model-based definition. In 2016 Manufacturing Science and Engineering Conference. American Society of Mechanical Engineers, 2016.

[13] World Wide Web Consortium. Semantic web, 2006. URL https://www.w3.org/standards/semanticweb/.

[14] Telecommunication Standardization Sector of ITU. Information technology – open systems interconnection – the

directory – part 8: Public-key and attribute certificate frameworks, 2014. URL http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=64854.

[15] David W. Hempe. Advisory Circular 21-48. Report, Federal Aviation Administration, U.S. Department of Transportation, 2010. URL http://www.faa.gov/documentLibrary/media/Advisory_Circular/AC%2021-48.pdf.

[16] John M. Allen. Advisory Circular 20-62E. Report, Federal Aviation Administration, U.S. Department of Transportation, 2010. URL http://www.faa.gov/documentLibrary/media/Advisory_Circular/AC%2020-62E.pdf.

[17] V. L. Hamilton and M. L. Beeby. Issues of traceability in integrating tools. In IEEE Colloquium on Tools and Techniques for Maintaining Traceability During Design, pages 4/1–4/3, 1991.

[18] B. Ramesh. Process knowledge management with traceability. IEEE Transactions on Software Engineering, 19(3): 50–52, 2002. ISSN 0740-7459. doi: 10.1109/MS.2002.1003454.

[19] Kannan Mohan and Balasubramaniam Ramesh. Traceability-based knowledge integration in group decision and negotiation activities. Decision Support Systems, 43(3):968–989, 2007. ISSN 0167-9236. doi: http://dx.doi.org/10.1016/j.dss.2005.05.026. URL http://www.sciencedirect.com/science/article/pii/S0167923605000916.

[20] Kannan Mohan, Peng Xu, Lan Cao, and Balasubramaniam Ramesh. Improving change management in software development: Integrating traceability and software configuration management. Decision Support Systems, 45(4):922–936, 2008. ISSN 0167-9236. doi: http://dx.doi.org/10.1016/j.dss.2008.03.003. URL http://www.sciencedirect.com/science/article/pii/S0167923608000523.

[21] M. Z. Ouertani, S. Baïna, L. Gzara, and G. Morel. Traceability and management of dispersed product knowledge during design and manufacturing. Computer-Aided Design, 43(5):546–562, 2011. ISSN 0010-4485. doi: http://dx.doi.org/10.1016/j.cad.2010.03.006. URL http://www.sciencedirect.com/science/article/pii/S0010448510000618.

[22] J. Yang, S. Han, H. Kang, and J. Kim. Product data quality assurance for e-manufacturing in the automotive industry. International Journal of Computer Integrated Manufacturing, 19(2):136–147, 2006. ISSN 0951-192X. doi: 10.1080/09511920500171261. URL http://dx.doi.org/10.1080/09511920500171261.

[23] Mike Collins. The Boeing supply chain model. Manufacturing.net, 2010. URL http://www.manufacturing.net/news/2010/07/boeing-supply-chain-model.

[24] Yoshihito Kikuchi, Hiroyuki Hiraoka, Akihiko Otaka, Fumiki Tanaka, Kazuya G. Kobayashi, and Atsuto Soma. PDQ (product data quality): Representation of data quality for product data and specifically for shape data. Journal of Computing and Information Science in Engineering, 10(2):021003–021003, 2010. ISSN 1530-9827. doi: 10.1115/1.3402615. URL http://dx.doi.org/10.1115/1.3402615.

[25] Dan Walker. Introduction to TOPGUN XI. In 2001 COE Conference, 2001.

[26] International Standards Organization. Industrial automation systems and integration – JT file format specification for

3D visualization, 2012.

[27] International Standards Organization. Document management – 3D use of product representation compact (PRC) format – part 1: PRC 10001, 2014.

[28] Automotive Industry Action Group. Defining product data quality, 1999.

[29] Automotive Industry Action Group. Product data quality: Guidelines for the global automotive industry, 2001.

[30] International Standards Organization. SASIG product data quality guidelines for the global automotive industry, 2006.

[31] US Department of Defense. Standard practice: Technical data packages, 11/1/2009 2013.

[32] Kevin Fischer, Phil Rosche, and Asa Trainer. Investigating the impact of standards-based interoperability for design to manufacturing and quality in the supply chain. Report NISTGCR 15-1009, National Institute of Standards and Technology, 2015. URL http://www.nist.gov/manuscript-publication-search.cfm?pub_id=920033.

[33] International Standards Organization. Automation systems and integration – numerical control of machines – program format and definitions of address words – part 1: Data format for positioning, line motion and contouring control systems, 2009.

[34] Suk-Hwan Suh. Theory and design of CNC systems. Springer series in advanced manufacturing. Springer, London, 2008. ISBN 1848003358 (hbk.) 9781848003354 (hbk.) 1848003366 (ebook) 9781848003361 (ebook).

[35] Dimensional Metrology Standards Consortium. Part 1: Overview and fundamental principles in quality information framework (QIF) an integrated model for manufacturing quality information, 2014.

[36] International Standards Organization. Document management – portable document format – part 1: PDF 1.7, 2008.

[37] The Internet Engineering Task Force. Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile, 2013. URL https://datatracker.ietf.org/doc/rfc5280/.

[38] The Internet Engineering Task Force. An internet attribute certificate profile for authorization, 2013. URL https://datatracker.ietf.org/doc/rfc5755/?include_text=1.

[39] Telecommunication Standardization Sector of ITU. Security in telecommunications and information technology. Report, International Telecommunication Union, 2004.

[40] Organization for the Advancement of Structured Information Standards. Assertions and protocols for the OASIS security assertion markup language (SAML) v2.0, 2005.

[41] Organization for the Advancement of Structured Information Standards. About us, 2015. URL https://www.oasis-open.org/org.

[42] Walter Isaacson. How Steve Jobs' love of simplicity fueled a design revolution. Smithsonian Magazine, 2012. URL http://www.smithsonianmag.com/arts-culture/how-steve-jobs-love-of-simplicity-fueled-a-design-revolution-23868877/?no-ist.

[43] National Institute of Standards and Technology. Security and privacy controls for federal information systems and organizations. SP 800-53, 2015. doi: 10.6028/NIST.SP.800-53r4. URL http://www.nist.gov/manuscript-publication-search.cfm?pub_id=917904.

[44] M.J. Cheng and J.E.L. Simmons. Traceability in manufacturing systems. International Journal of Operations and Production Management, 14(10):4–16, 1994. doi: doi:10.1108/01443579410067199. URL http://www.emeraldinsight.com/doi/abs/10.1108/01443579410067199.

[45] M. H. Jansen-Vullers, C. A. van Dorp, and A. J. M. Beulens. Managing traceability information in manufacture. International Journal of Information Management, 23(5):395–413, 2003. ISSN 0268-4012. doi: http://dx.doi.org/10.1016/S0268-4012(03)00066-5. URL http://www.sciencedirect.com/science/article/pii/S0268401203000665.

[46] International Standards Organization. Industrial automation systems and integration – product data representation and exchange – part 21: Implementation methods: Clear text encoding of the exchange structure, 2002.

[47] Niklaus Wirth. What can we do about the unnecessary diversity of notation for syntactic definitions? Commun. ACM, 20(11):822–823, 1977. ISSN 0001-0782. doi: 10.1145/359863.359883.

List of Tables

List of Figures

List of Listings