Using S1000D to Bridge Collaboration Gaps Between Technical Data and Supporting Training Content

Wayne Gafford
Advanced Distributed Learning Co-Lab
Alexandria, Virginia
Wayne.Gafford@adlnet.gov

ABSTRACT

The Department of Defense (DoD) training community, lead by the Advanced Distributed Learning (ADL) Initiative, is accustomed to specification-driven content development. The Sharable Content Object Reference Model (SCORM) provides guidelines for referencing learning objects in courseware, and how that courseware communicates with a learning management system (LMS). In addition, Content Object Repository Discovery and Registration Architecture (CORDRA) is the framework that fosters registration, search and discovery of content through the ADL Registry (ADL-R). DoD SCORM and ADL-R policies are detailed in the Department of Defense Instruction 1322.26, "Development, Management and Delivery of Distributed Learning". However, there is a specification gap in the training arena that has not been addressed: XML structured learning content. Historically, learning content has been committed to formats not designed for configuration and content management, such as HTML and Flash. DoD training content can benefit from the use of XML that enables interoperability and management. This paper will demonstrate how learning content structured in the S1000D international technical data spec is a benefit to the DoD training community. The paper will also demonstrate how technical data structured in S1000D can be imported directly into courseware and made SCORM-conformant. The paper will conclude with a discussion of plans to support training requirements in S1000D.

ABOUT THE AUTHOR

Mr. Gafford combines a background in teaching and education with XML-based standards that has resulted in innovative ideas for e-learning content management and data interoperability. For the last three years, Mr. Gafford has lead subcommittees, studies and prototype projects that explore how learning content can benefit from structured markup. Results have lead to an increased awareness that standardized metadata and XML structure can unify diverse but related content that support common systems, procedures and products. Mr. Gafford has taken his research to the Advanced Distributed Learning Initiative where he is the Deputy Director of the Job Performance Technology Center. He is a member of the DITA and S1000D learning subcommittees, an active public speaker at S1000D and ADL events, and a supporter of developing XML schemas that model instructional development and learning content to improve knowledge management and distributed learning.

Background

An insightful irony separates two workforce cultures in the Department of Defense (DoD): one culture that develops highly technical systems data according to rigid specifications, and another culture that develops the learning materials to support the operation and maintenance of those systems. This irony is buried deep in each culture's obligation to use standards in its content development process.

The technical data community is populated by professionals that utilize strict content guidelines that must expressly conform to the size, fit and function of a particular system. The creativity in documenting the operation and maintenance of a technical system is limited to the operation and maintenance of the technical system. Liability issues, contract requirements, and military specifications constrict technical language to its most essential truths: if it breaks, how is it fixed? What are the safety procedures?

The technical training community is populated by professionals that utilize the art and science of instructional design, learning theories and creative thought to convincingly impart information to a learner to enhance human performance and operational readiness. The range of educational theories, instructional technologies and content development methods constitute a creative mix of strategies that can result in unpredictable styles of how learning content is developed and conveyed.

The irony resides in each culture's adoption of standards. The technical training community, instilled with a diversity of creative approaches and methods to building learning materials, uniformly adheres to building content that must operate according to SCORM. Although SCORM does not declare *what kind of content to develop*, it does specify *how a collection of content is digitally packaged* so it can render and play on a Learning Management System (LMS). The LMS presents the learning material and tracks the learner's progress through a mechanism known as an *Application Program Interface* (API). The API allows the learning content package to pass expected pieces of information to the LMS. This API forms the interoperable core of SCORM. It is the essential communication device that allows any SCORM-conformant content package to work on any SCORM conformant LMS.

The technical data community, instilled with strict guidelines in what content can be developed, adheres to a diversity of technical specifications. The Navy has used at least three specifications, while the Army, Air Force, and Marine Corps each have their own. The result has been a diverse set of Interactive Electronic Technical Manual (IETM) implementations that are not based on a common API. The closest the technical data community comes to any uniformity in data development is its use of the Standard Generalized Markup Language (SGML) and its use of the eXtensible Markup Language (XML). SGML and XML, as structural data standards, can still result in implementations where an Army IETM cannot render or play on a viewer developed for the Navy. One reason is because the specifications that declare the type of SGML or XML are different from service to service.

The cultural irony between the two communities is apparent: the training community approaches learning content development from diversified strategies whose results are committed to a uniform method of delivery and run time operation. The technical data community approaches content development according to strict guidelines which often results in delivery and run time products that are not interoperable between viewers.

The resulting paradox in the comparison of specification practices between each content development community is that *technical training content is directly based on its authoritative technical source data.* The supportability characteristic inherent in training content is not mirrored in how training and technical data communities collaborate: the communities do not collaborate. There is a collaboration gap between each community that reveals how *they do not observe a common specification for content development despite developing content to support a common system*

The reason for this irony is not the central investigation of this paper, but an observation that launches its premise, the repercussions that ensue, and a solution that will not eliminate the irony but foster an environment where the irony does not perpetuate a lack of collaboration between the two communities.

The Collaboration Gap

There is a growing recognition that technical data development and its supporting technical training development ought to be created together. This holistic perspective has never been possible under current organizational practices. For example, the training community often receives technical manuals via distribution lists … the same lists used to distribute

manuals to regular end users. This practice lumps training production organizations with end users that ought to receive updated training content in sync with updated technical manuals. The distribution list mechanism automatically builds in a scheduling lag.

Solving the data problems that naturally occur between organizations that depend on each other will require attention to infrastructure and communication. Knowing what the problems are will suggest solutions. For instance, the training community does not have direct access to technical data assets in a database. The training side does not have the opportunity to pair task analysis with learning objectives development during the technical design phase. Those are the moments when true content reuse can be planned across the program enterprise.

Coordinating the communities must start with a common denominator. Certainly, each has a common system to support; each has a common interest in using specifications. These elements have not been enough to motivate and shape a collaboration strategy. Collaboration cannot begin with a networked infrastructure or common source databases, because these elements do not exist in a mold that unifies the communities. The approach to a collaborative strategy for content development ought to start with basic requirements: *what are the technical data requirements for training, and how can those requirements be met?*

How the requirements are met is just as significant as the requirements themselves. The basic technical data requirement for training is access to content on which instruction is based. Obtaining the data long after the IETM is distributed creates scheduling problems of a different sort. An enlistee taking a course in **a** schoolhouse may be learning about requirements on a system version that is out of date after reporting to the position for which the training is intended. The Advanced Distributed Learning Initiative's Job Performance Technology Center (ADL-JPTC) recognizes this issue not as **a** distribution problem, nor as a technology problem, but as a problem in the lack of a common digital data standard between the communities.

THE ADL SPECIFICATION GAP

The ADL mission is to improve the way DoD delivers self-directed training to its service members. The answer to that challenge came in the form of SCORM, and as described earlier, centered on a common API between a content package and an LMS. The specification did not address content development, or content development quality, or reuse. What it did for learning content that the technical data community is just beginning to address, is that SCORM helped transform large, single file courses spanning many hours into smaller learning events. The course content included independent files known as assets. These assets, in computer jargon, became known as objects. SCORM does not prescribe the object's content, size or format…*SCORM is silent on content.*

The content development result in the first era of SCORM is that technical training data, dependent upon authoritative technical source material, is stored and managed in distribution formats. HTML, Flash and other formats that do not carry life cycle and configuration management *has become the de facto content schemas and distribution formats*. These formats are tightly integrated with rendering and interactivity capabilities, initially making them a natural data format option for training developers.

ADL's second specification, Content Object Registration and Discovery Resolution Architecture (CORDRA), is also silent on content and format. CORDRA is an architecture that enables search and discovery of information in a system of registered repositories. CORDRA as an architecture requires an instantiation of the architecture in the form of a registry for a content search to take place. The point is not to fully describe CORDRA but to highlight that between SCORM and CORDRA, ADL does not specify how technical learning content ought to be managed *during its development and production phase in a data standard suitable for collaboration with authoritative technical sources.*

The ADL specification gap sits at the learning data source level as it exists in a content management system prior to being aggregated into a Sharable Content Object (SCO). The common denominator between technical learning content and technical source content is found at the raw data level.

Pursuit of Commonality

The natural commonality between the technical data and the supporting learning content is the subject matter. Whether content is fashioned as technical data procedures about a system or fashioned as instructionally designed content meant to instruct the procedures, the subject is still the procedures themselves. Common topics can lead to common file naming and identification. It can also lead to common configuration and life cycle management metadata.

The collaboration is enabled through the appropriate choice of a data specification to support the collaborative and management needs of the content. ADL's position on which specification to use is to drive the choice based on data life cycle requirements: the right spec for the right data environment. The specification gap is where mixed content interoperability and collaboration can take place between technical data and related technical training communities.

The Path to a Collaboration Strategy

Since 2004, two subcommittees were established to discuss and study how the training community can benefit from direct access to its technical authoritative source material. The first group worked within the Navy's Integrated Learning Environment. The ILE, created and funded by the Naval Education and Training Command (NETC), is a program that works to provide learning and performance improvement opportunities to the sailor. All Navy training content is managed within the ILE environment.

Although the subcommittee no longer exists, its results helped establish two important issues: 1) training and technical communities have different semantics for common terms; 2) no network or computer-based infrastructure exists to enable any type of interoperability picture. In the final analysis, the group did not solve tech data requirements for training. It did, however, allow each community to peer into the other's profession and approximate the gap in the professional relationship. More significantly, the way in which the group discussed the problem underscored the lack of any insightful headway toward a solution. Discussions focused on semantic differences and infrastructure voids and left absent any unifying standard or discussion point required for progress.

In 2005, another training subcommittee was established to discuss and study how technical data and training content could work together. This current working group came under the S1000D International Technical Data Specification implementation group. S1000D is the harmonization of several technical specifications. In 1984, the Aircraft European Contractors Manufacturers Association (AECMA), along with partners like the British Ministry of Defense (MoD), collaborated on the specification project. Its first release in 1989 was based on the Air Transport Association of America (ATA) Spec 100. AECMA reformed into ASD. ASD is a merger of AECMA, EDIG (European Defense Industries Group) and EUROSPACE (Association of the European Space Industry). The group TPSMG (Technical Publications Specification Maintenance Group) of ASD and the Aerospace Industries Association (AIA) are responsible for the development and maintenance of the specification. S1000D is evolving to support air, land and sea systems requirements. As of this writing, the DoD has yet to adopt the specification. However, many programs are using S1000D. NAVAIR issued a directive in 2006 requiring all new acquisitions to procure technical data in S1000D.

In this context, the focus became the utilization of a data standard in support of solving data requirements issues for training. The group viewed the problem *through the lens of a standard and not through the lack of an infrastructure or through differences in semantics.* The standard released the discussion from roadblocks and into a context that offered an opportunity to solve a problem based on a unifying element: a technical data standard that offered content support in areas that SCORM does not: *naming, identifying and structuring source data in a common source database.*

Ironically, the group did not realize this virtue until after it completed its first white paper. That first white paper went much further than the Navy ILE training group, as it compared and contrasted the characteristics of S1000D and SCORM in the context of data sharing. This valuable discussion moved the debate beyond semantics and into a pure retrospection of two compatible standards. However, it took questions about what to do with the white paper in terms of forward motion that finally lead to a critical and practical breakthrough toward an interoperable solution.

A Question

In order to move the discussion forward after the white paper's completion, the group asked the question: *how does SCORM names files as compared to how S1000D names files?* The valid question came from an S1000D point of view, *because its file naming convention together with its life cycle management capabilities form the core of S1000D.* Upon examination, the group discovered that SCORM does not name files, rather it applies names to whole courses through a "title" element. These courses are made of many files which could be randomly named and not according to a specification. S1000D does name files in the form of a *data module code.* This code applies specific meaning to content while using the name as a configuration item. In this comparison, SCORM is revealed as a "model" that is concerned with how content is delivered, and S1000D is a "markup" specification that

is concerned with how content is created. Therefore, realizing that training is based on technical content, S1000D can fill ADL's specification gap by naming, identifying and structuring technical training content according to its naming and identification conventions.

Which Specification to Use?

ADL focuses on requirements and using standards to meet those requirements. Its original mission is linked back to the improvement of distributed learning in DoD. The learning content can range from highly technical system-specific content to rules that are applied by specific policies. In the former case, the learning content is subject to the configuration and life cycle of a piece of equipment. In the latter case, the learning content is based on policy documents. What is common between the two learning content instances is that each is based on authoritative source material. The learning objectives and content are subject to change when the authoritative source changes.

For the purposes of this paper, the focus is on learning content based on technical data that is written to support configurable equipment throughout a long life cycle. The specification that best meets the reuse and change management requirements *for all technically oriented data supporting a common system is S1000D.*

Why S1000D for System-based Learning Content Development?

What SCORM did for courseware development S1000D does for technical publication development. As SCORM helped to break large courses down into smaller units and chunks, S1000D breaks technical data down into smaller units and chunks that reflect the assembly, disassembly, procedure and maintenance schemes for equipment and systems. Authorship is not written with *an entire publication in mind, but with the idea to support a discreet task or concept.* The ability to devise content *on a discreet basis* helps enable the chances for reusable data. The reuse is a two way proposition: tech data can be reused in a course, and learning-based multimedia can be reused to enhance an IETM.

The primary reason for using S1000D is not because of reuse. Reuse is a bonus if and when it can be achieved. The primary reason for using S1000D is to name, identify and structure technical training and technical data content during *life cycle configuration management.* It is crucial for a data manager to know at a given point how an engineering change proposal effects the entire data support environment. There must be a common thread that is strung throughout the data. That data thread is manifested in metadata. The use of S1000D provides a common set of metadata *for all life cycle supportability content, thereby enabling queries and tools that help manage the content en masse.* This feature goes back to requirements. If the requirement is to track changes across all related content, then what spec will supply the means to meet that requirement? If the requirement is to distribute the proper content to an end user based on a configuration, what spec will supply the means to meet that requirement?

The ADL Specification Gap as an Opportunity

The opportunity for ADL to recommend the use of S1000D to meet its technical training specification gap will make possible a larger DoD vision than simply using XML to structure learning material. ADL can continue to act as a leader in the application of standards in the public and federal sector by emphasizing the holistic principles of enterprise content management using common digital data formats. ADL, an initiative rooted in DoD's Personnel and Readiness Office, can engage its counterparts in Material Readiness and Material Planning, an entity that contributes to technical data standards policies. If a common spec has the realistic potential to configure training to technical data, then the same can be applied to *all technical information supporting a common system.*

ADL will not be departing from its original mission, but acting on behalf of its mission in an endeavor benefiting all principles with investments into systems data. For the first time, DoD has an opportunity to address a little known problem that very few recognize: apply a standards-based solution to achieve technical data interoperability. The key to the interoperability solution is that it not be based on vendor products. Industry solutions and government policy for enterprise content *must be predicated on data standards.* S1000D enables program managers to take a more comprehensive inventory of its full compliment of system supportability data to a point that the data can be viewed as configuration items.

Finding Proof That S1000D is a Solution to the Collaboration Gap

A conceptual policy rendering of interoperable technical systems data can only be based on prototypes and implementations that showcase standards-based collaborative production. Over the last three years leading into the first quarter of 2007, case studies and projects have demonstrated the use of S1000D in

technical training development. The early concern about the strategy focused on XML as an inhibitor to instructional design. That concern evaporated in early prototypes that hypothesized that S1000D can name, identify and structure technical training content. Just as instructional design methodologies could be implemented in HTML, S1000D accommodated for the same practices…to a point. HTML needs the dynamics of javascript and other coding languages to allow for interactivity. S1000D is the same. HTML and S1000D will structure data, but require partnerships with programming languages to allow interactivity. Flash is often the format of choice for interactivity, which can easily be referenced by a data module. The challenge is to manage the learning data in the data module then allow Flash to import the data from the management environment.

There are numerous ways to implement a standards-based solution for interoperable data. Ironically, standards do not restrict implementation, but optimize options based on a common set of rules. HTML is the common markup for all websites, which has not impeded the diversity of styles and implementations. In fact, HTML's success and popularity lead to a demand for more web-based capabilities. That demand gave way to XML. XML can be used **as** a common digital data standard upon which interoperability can be based. Current S1000D prototyping is showing that a common markup language can support a variety of processing and output.

The variety of S1000D output is scoped down to technical content that support a complex array of machinery, weaponry, vehicles and other systems that have long and expensive life cycle requirements. In this regard, the power of S1000D is that it is *an information specification and not simply an XML architecture.* A specification that does not feature an information component will not serve the complex configuration and life cycle requirements demanded by program managers. S1000D is complex *because it supports a complex process.*

Projects and prototypes that seek to validate S1000D as a common digital data standard for system documentation and fill the ADL specification gap must highlight the semantic flexibility inherent in the specification. ADL's confidence in S1000D as a markup specification for technical training is validated in its application to related technical material. The evolution of the specification to support learning material will be discussed in the change proposal section of this paper.

The Early Prototypes…Proving the Hypothesis

Validating that S1000D does not impede instructional design was the important first step for training communities to accept the specification. **Figure 1** is a screen shot of the opening instructional page to one of the first prototypes to use S1000D in a SCORM-conformant sample course. The instructional design is apparent. The learner reads the introductory paragraph and is immediately put into a job-task context: "You are performing the maintenance turn-on procedure…". "You have applied primary power…" "Now you must check the inlet pressure gauge…" Notice that the training did not start with a physical description of the system. The training started with human performance as its driver. Once the learner's performance context is established, interactivity is offered. The interactivity helps establish how much the learner knows…and does not know about the system. This is accomplished by dragging and dropping component names into the empty fields. The check will allow instant feedback.

The learning strategy is apparent and not limited by S1000D…except in one regard: the interactivity piece is flash based. S1000D cannot drive multi-media based interactivity alone and neither can the vast majority of XML specifications. However, the interactivity supplied by Flash is still supported by S1000D in an important way: the jpeg graphic file used in the flash file is named according to the Illustration Control Number (ICN), which is a specialized graphic naming convention in S1000D. That file is used as a basis for the interactivity. The functionality that makes the interactivity interactive, such as the drag and drop, does not need to be named by S1000D. The visual components that represent the system do.

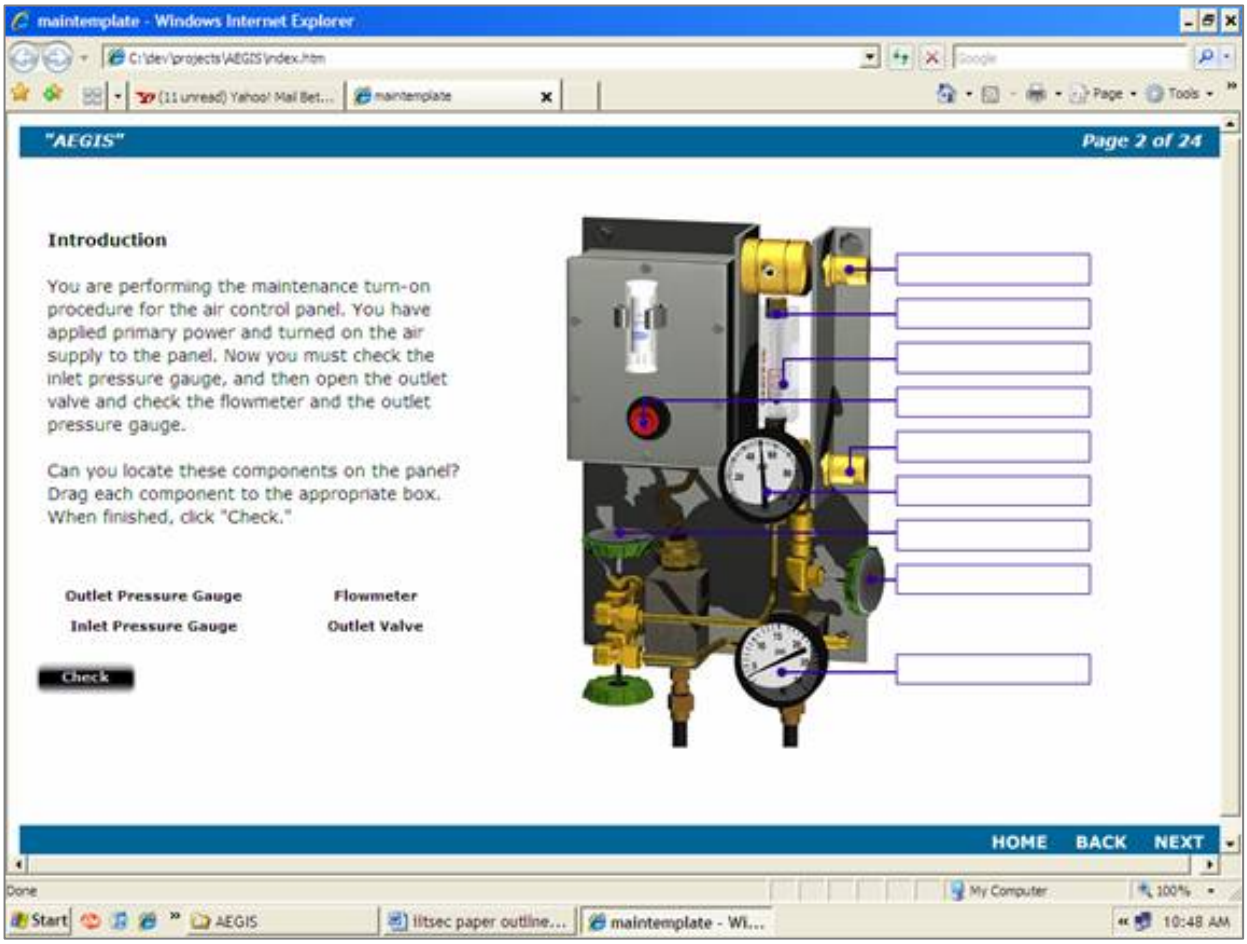

Figure 1

Figure 2 displays the S1000D data module that contains the content presented in figure 1. The first paragraph in the slide is highlighted. The paragraph is identified with an "id". Notice the <graphic> tag below the highlighted paragraph. It references the graphic with an ICN. Notice how the referenced graphic is declared in blue highlights at the top of the file. The graphic and content are imported into Flash at the time the SCO is launched. Finally, notice the data module file name highlighted in blue above the menu bar. The training file is a data module named by a data module code. The only training identification naming support in the file name is the last character. That character is called an "Item Location Code" (ILC). The ILC describes where the activity described in the file takes place. In this case, the activity takes place in a classroom or in a learning context.

Further Support for Training in S1000D

The current data modules in S1000D will certainly hold technical learning content within its structures. SCORM will easily accept data modules as assets in a SCO. The next step is to extend and support S1000D with learning-specific features. These steps require engaging in the S1000D Change Proposal Form (CPF) process. Five CPFs, outlined below, are currently under consideration to support training. The S1000D training CPFs originate from the following premises:

Premise one: SCORM **does not** set semantic requirements for how files are named. Naming is declared at the organization level, which translates to a course name. SCORM is content and format agnostic, which is an important characteristic of a model that references multiple standards. SCORM is not a markup specification. SCORM does not have a schema for

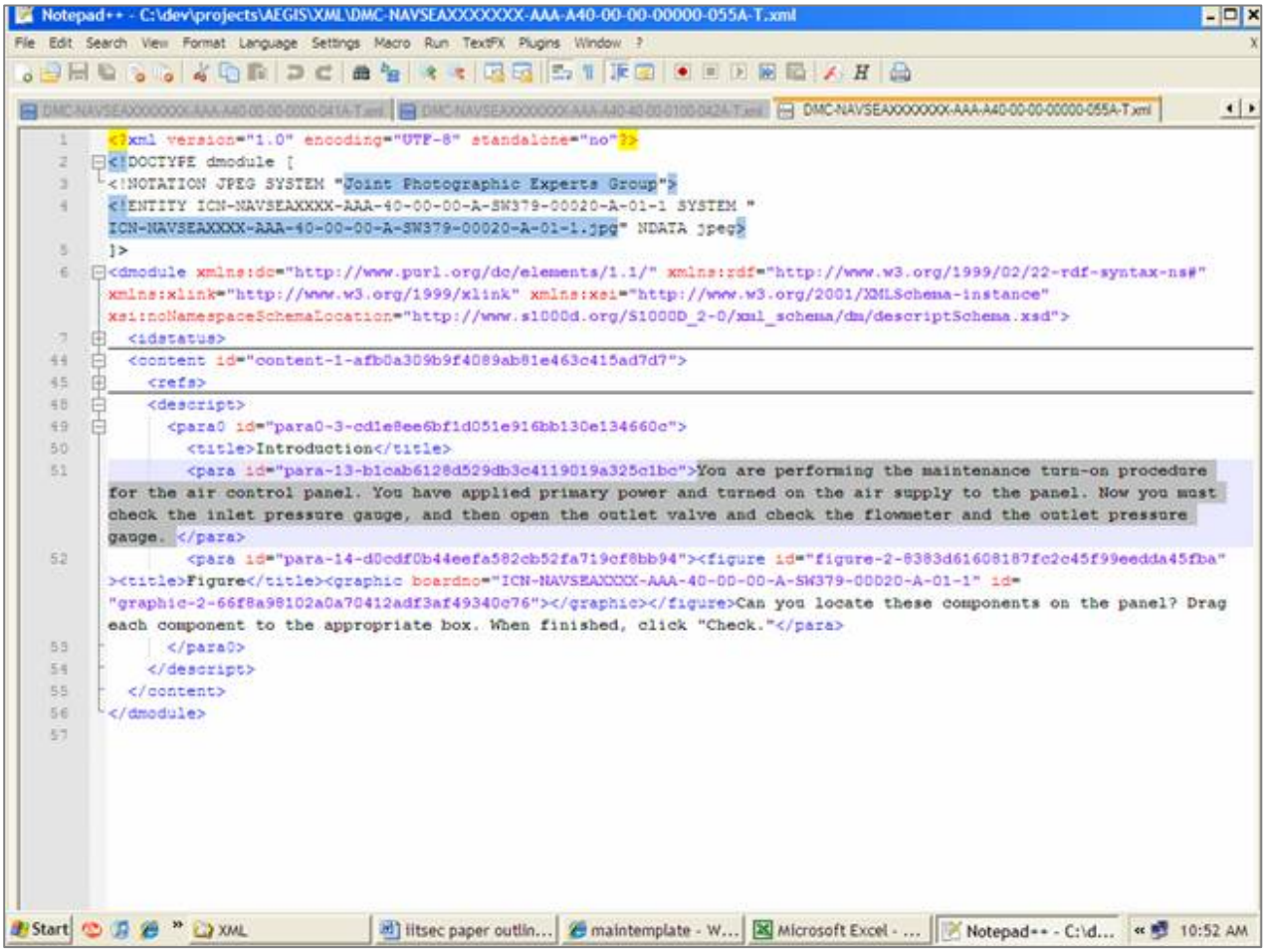

Figure 2

structuring actual learning content. SCORM includes metadata, a manifest and sequencing information.

Premise two: S1000D *does* set semantic requirements for how files are named. S1000D is not content and format agnostic, which is an important characteristic of a markup specification. S1000D is a markup specification. It wraps tags around the actual source content. S1000D offers ten singular placeholders for naming files that reside within the source content. Each placeholder is strung together to form the data module code, the essential content-identifying feature formed into a file name.

CPF #1: Importing LOM into the Publication Module

Sponsor: Carla Kieckhefer - **L-3/D.P. Associates Inc.**

This CPF proposes to add the IEEE Learning Object Metadata (LOM) requirements in the SCORM Content Aggregation Model (CAM) to the PMSTATUS section of the Publication Module. This addition will support the SCORM conformance requirement for training data modules intended for SCO packages.

SCORM-conformant SCOs consist of all of the content that makes up the SCO, including declared assets, metadata, and sequencing and navigation. The required metadata for a SCO package consists of the Content Aggregation Model (CAM) and the Sequencing and Navigation Model. The Sequencing and Navigation Model contains much of the same information required by the Publication Module in S1000D. Currently S1000D metadata or content tags do not map to the CAM. CAM describes:

- The components used in a learning experience,
- How to describe those components to enable search and discovery, and

- How to define sequencing information for the components.

Using the IEEE LOM in the PMSTATUS section during training content development will ease the process by which data modules and publication modules meet SCO processing requirements.

CPF #2: Training Content Reuse Preplanning Guidelines

Sponsor: Kevin Reuss – Intelligent Automation, Inc

This CPF proposes to develop a guide, including a table or matrix that will simplify the identification of project factors likely to influence reusability. Many of these factors will be the same as those that already go into determining reusability for tech data, such as data module size, system breakdown, etc., but this CPF addresses the specific reusability needs of training content. The guide will also present the case for sharing source content and the costs of not doing so, and set forth basic principles to inform the effort throughout the process.

The reusability of training content in relation to the equipment, training distribution, and learner environment will be a function of how well the goal of reuse is integrated into the development effort and its products. This design for reusability must be scoped at the beginning of a project and involve representation from both tech data and training contributors.

CPF #3: Learning Content Models into Data Modules

Sponsor: CDR Micheal Mazzone – Submarine Learning Center

This CPF proposes modifying the <content> wrapper to support learning types, which will provide DoD an XML environment to support the above management and configuration requirements. These learning types will include: Lesson Planning, Objectives, Learning Content, Enhancement Content, Summaries, and Assessments.

Current DoD technical training content is not developed in formats that facilitate life cycle management and configuration control. Additionally, there is no method for identifying how learning content is affected by changes in authoritative sources.

CPF #4: Learning and Human Performance Information Models Identified in DMC Information Code

Sponsor: Jeff Clem – Lockheed Martin

This CPF proposes the expansion of the data module code (DMC) to reserve a range of information codes that specifically define data modules that contain information intended for training or human performance support products. It will allow for the easier and more precise identification of data module information written for training purposes or use in human performance support systems employed outside the context of training programs. As an example of the latter, information that can be pulled into a job aid or quick reference guide format that can be used on the job by knowledgeable workers. Another example might be technical skill or test tool requirement data that organizations (e.g., US Navy) could extract to define/validate job skill requirements. The Publications Module can then be used for the content and associated maintenance data modules to more precisely describe a training course, module, lesson and/or topic and/or performance support requirements.

Research across DoD training communities has shown the value of integrating technical information from S1000D sources into technical training support products. Current S1000D specifications do provide for the identification of training information through the use of a "T" at the end of a data module code. However, this is too generic and limiting to fully support interoperability with SCORM and common requirements of Instructional Systems Development. Furthermore, learning is a continuous process that not only occurs during defined training events but continues in non-training contexts, such as during the performance of on-the-job tasks.

CPF #5: Data/Publication Module into SCO Processing Tool

Sponsor: Wayne Gafford – Advanced Distributed Learning

This CPF proposes to use a government-owned data module-to-SCO processing application originally written for a specific prototype deliverable and transform it into a publicly available tool. The SCORM processing tool will be made available on the S1000D.org website, and its documentation will be inserted into the specification.

Recent prototypes have demonstrated how data modules containing learning and maintenance content can be processed into a sharable content object and made ready for a learning event. Processing a data module into a SCORM-conformant content package is an important step in realizing how the complete cycle of content from the common source database (CSDB) to the rendered output is made. However, there is no publicly-available tool that transfers data modules into SCORM content packages. The ability to process data modules into content packages will promote the use of S1000D for technical training.

CONCLUSIONS AND SUMMARY

The discovery of the "ADL Specification Gap" allowed S1000D to offer file naming and markup support to technical training content. The DoD training community, although accustomed to spec-driven content development according to SCORM, has not utilized a markup specification for learning and instructional source material. HTML, Flash and other multimedia formats have held the learning content (graphics, samples, assessments…the language engaged by the learner). Word and other editor file formats have held the instructional content (objective statements, prerequisites, material lists, cognitive levels, learning strategies).

The suggestion of using a markup specification to contain learning content met resistance and doubt as to the utility and reason for its use. Primarily, it was perceived that a markup spec would impede instructional design. This was a fair concern and set the opportunity to formulate the first significant prototype for how S1000D can support learning content. The prototype challenged and overcame the fear that a markup spec would impede instructional design through the following hypothesis:

S1000D can name, identify and structure technical learning content without impeding instructional design and the science of learning.

The first prototype set out to test this hypothesis by using S1000D data modules to structure Navy technical training and Navy technical maintenance. The results concluded that the XML tag set did not impede the application of learning strategies to instructional design essential to thorough lesson planning. The data module codes accurately described the technical learning content in the XML files. Technical data in the IETM was deliberately planned for reuse in the training. S1000D data modules were used as assets and made SCORM conformant.

The next step for collaborative content development is to assume a larger role for S1000D beyond technical data and training. Those data products are only two elements within the entire system life cycle environment. Planned maintenance, testing, evaluation, parts lists, wiring, design, schematics and installation instructions join with technical manuals and training to form a more complete picture of all content written to support a system. Using S1000D to integrate this content would transform technical content into configuration items, for the content is written expressly to support configurable systems. Only then will standards-based data interoperability be truly viable.