Auto-Authoring Instruction from Ontological Representations of Procedures

Alan O Meeks, Gabriel Aviles, Lee Lacy
Dynamics Research Corporation
Orlando, Florida
ameeks@drc.com, gaviles@drc.com, llacy@drc.com

ABSTRACT

The Semantic Web is the next evolution of the World Wide Web (WWW), offering new technology solutions for developing and presenting instruction. The key enabling technology supporting the Semantic Web is ontologies. A new WWW Consortium (W3C) standard, the Web Ontology Language - OWL, is used to encode ontologies for the Semantic Web. OWL can be used to represent and structure procedural knowledge for use in job aid delivery systems. By auto-authoring procedural instruction in real-time based on current conditions, development costs are reduced and just-in-time training can occur.

This paper describes the results from a Defense Advanced Research Projects Agency (DARPA) research effort to develop ontologies and software to auto-author procedural Interactive Multimedia Instruction “on-the-fly” from disparate Semantic Web knowledgebases. The job aid delivery system dynamically selects content using specific user conditions when instruction is needed. This capability is enabled through the use of OWL-encoded procedural knowledgebases. The content is formatted based on pedagogical rules that take into consideration the end user’s form factor and delivery mechanisms including user interface issues (e.g., screen real estate). Sample demonstration content has been developed for Explosive Ordnance Disposal (EOD) technicians responsible for handling Improvised Explosive Devices (IEDs).

ABOUT THE AUTHORS

Alan Meeks is a software engineer for Dynamics Research Corporation (DRC). He has worked on several large simulation programs including WARSIM2000, JSIMS, and the Close Combat Tactical Trainer (CCTT). He has a B.A. in Political Science with a Minor in Computer Science from the University of Central Florida.

Gabriel Aviles is a software engineer for Dynamics Research Corporation (DRC). He has participated in a DARPA/DAML Computer Aided Knowledge Engineering research and development effort to define a semantic web language, DAML+OIL which has evolved to OWL. He has a B.S. in Electrical Engineering from Boston University.

Lee Lacy is the Director of Orlando Operations for Dynamics Research Corporation (DRC). He has worked on major simulation programs including the OneSAF Objective System, WARSIM 2000, and the Close Combat Tactical Trainer (CCTT). He serves as the Vice-Chairman of the Board of Directors of the National Center for Simulation. He is currently the Principal Investigator on DRC’s DARPA Agent Markup Language (DAML) semantic web research effort. His research areas include the semantic web, XML, and Computer Generated Forces. He received an M.S. and B.S in Computer Science from the University of Central Florida and is a Certified Modeling and Simulation Professional.

Auto-Authoring Instruction from Ontological Representations of Procedures

Alan O Meeks, Gabriel Aviles, Lee Lacy
Dynamics Research Corporation
Orlando, Florida
ameeks@drc.com, gaviles@drc.com, llacy@drc.com

INTRODUCTION

The Semantic Web is the next evolution of the World Wide Web (WWW), offering new technology solutions for developing and presenting instruction. The key enabling technology supporting the Semantic Web is ontologies. A new WWW Consortium (W3C) standard, the Web Ontology Language - OWL, is used to encode ontologies for the Semantic Web. OWL can be used to represent and structure procedural knowledge for use in job aid delivery systems. By auto-authoring procedural instruction in real-time based on current conditions, development costs are reduced and just-in-time training can occur.

This paper describes the results from a Defense Advanced Research Projects Agency (DARPA) research effort to develop ontologies and software to auto-author procedural Interactive Multimedia Instruction "on-the-fly" from disparate Semantic Web knowledgebases. The Job Aid Delivery System dynamically selects content based on specific user conditions when instruction is needed. This capability is enabled through the use of OWL-encoded procedural knowledgebases. The content is formatted based on pedagogical rules that take into consideration the end user's form factor (e.g. Tablet PC, PDA) and delivery mechanisms including user interface issues (e.g., screen real estate). Sample demonstration content has been developed for supporting Explosive Ordnance Disposal (EOD) technicians responsible for handling Improvised Explosive Devices (IEDs).

BACKGROUND

A variety of technologies have been leveraged to enable auto-authoring of instruction.

Semantic Web

The semantic web represents the next evolution of the World Wide Web (WWW) in which software will be able to glean information directly from web pages (Berners-Lee, 2001). Currently, most web pages are free-form text, encoded in Hypertext Markup Language (HTML), that are targeted for human readers. This seriously limits the ability of automated processes to perform complex operations such as integrating information from multiple web sites and performing inferencing of new facts based on provided information. The W3C manages many critical web standards (e.g., HTML, XML) and is championing new semantic web technologies.

Ontologies

Semantic web technologies promoted by the W3C are empowered by formal vocabularies, called ontologies, which describe particular domains of interest. By providing explicit computer-readable descriptions about a domain in an ontology, software can more easily interpret the descriptions of associated information. The ontologies explicitly define common information representation constructs and relationships between information such as class/instance relationships and subclasses.

OWL

Ontologies must be expressed in a standard computer language so that compliant tools can interpret the ontologies and their associated instance data. The W3C has recently released the Web Ontology Language – OWL, that is based on DARPA and European Union research that resulted in the DARPA Agent Markup Language combined with the Ontology Inference Layer (DAML+OIL) (W3C, 2004). OWL provides language constructs for defining classes, properties, and individuals. OWL is based on the W3C's Resource Description Framework Schema (RDFS) (W3C, 2004) technology that is encoded using the Extensible Markup Language (XML).

An ontologist, using OWL, can encode axioms about a particular domain in an ontology file. Instance files that comply with the ontology can be understood by OWL-compliant tools that have access to the domain's ontology.

Job Aids

A common tool for providing "just-in-time training" is an electronic job aid. The authoritative, approved DoD glossary of training terms (DoD, 2001) defines a job aid as "A checklist, procedural guide, decision table, worksheet, algorithm, or other device used by job incumbents to aid in task performance."

We use the term electronic job aid to refer to a subset of Electronic Performance Support System (EPSS) features. The DoD Glossary considers an EPSS to be "An integrated electronic environment that is available to and easily accessible by each user and is structured to provide immediate, individualized access to the full range of information, software, guidance, advice and assistance, data, images, tools, assessment and monitoring systems to permit performance with minimal support and intervention by others. EPSSs can also be considered a type of job performance aid."

Historically, many electronic job aids have been custom programmed using authoring software or more recently, with web design tools (e.g., Dreamweaver). Content developers typically harvest procedural and system description information as part of a knowledge acquisition process. That information is then normally encoded for a particular display format to support the delivery device. This approach suffers from a lengthy and expensive programming process.

An important distinction has been made between developing the content (authoring) and presenting formatted information (delivery). The concept of separating job aid content from specific formats is not new. Standards have evolved for representing and exchanging job aid information. These standards include Interactive Electronic Technical Manuals (IETMs) (DoD, 1995) and Sharable Content Object Reference Model (SCORM) (ADL, 2004). MIL-PRF-87269A (DoD, 1995) was developed nearly 10 years ago for interchanging IETM data using SGML (the precursor to XML). SCORM standards continue to evolve that support interchanging content across systems to support reuse and repurposing of the content.

JEOD DSS Background

It is the job of an EOD technician is to "render safe" IEDs. EOD technicians must have appropriate procedural information available to "render safe" encountered IEDs. The amount of procedural information currently available to the EOD technician is voluminous. It is currently organized in part by mission phase and is not tailored to meet an individual EOD technician's current situational need. EOD technicians must search through TTPs to find applicable procedures that relate to their situation. EOD technicians need a way to filter those procedures that are applicable to their current situation and they need a way to view procedural information based on their own display requirements.

Autoauthored Job Aids

Many job aid development tools have been developed that separate the content from the format. These systems often use database or proprietary information representation techniques to store the procedural and system information regarding a particular domain. That content is then delivered using independent formatting software that takes the content and formats it for user access. Early object-oriented research tools that pioneered this paradigm include the Procedure Job Aid Production System (PJAPS) (Mears, 1989) and the Prototype Animated Training Hypermedia System (PATHS) (Lacy, 1992). Pedagogical rules can be applied to add sound instructional approaches for the presentation of the job aid information (Mears, 1989).

The benefit of autoauthoring job aid content on the fly or even in a batch mode is that the programming burden of formatting the content for delivery is greatly reduced. This saves time and development costs and results in a consistent delivery of the content. However, care must be taken to prevent the presentation from looking computer-generated and overly "canned".

PROBLEM DOMAIN

The need to filter procedural information exists in many domains. There are an endless number of domains that an auto-authoring Job Aid Tool could be used for. One could be an auto mechanics domain that contains procedures detailing how to change a tire. One user needing information on how to change a tire could have no idea how to do this. Another user may know how to change a tire, but not for a specific type of car.

Using this example, there is a need to filter information that is relevant to certain conditions (such as the whether someone has or has never changed a tire) related to its user. Another condition could be the type of car that needs a tire changed. Using these two conditions as filter values against all the possible steps to changing a tire could reduce the amount of information that would be presented to the tire changers. This would help the tire changers by reducing the amount of irrelevant information presented to them.

DRC used these auto-authoring technologies to implement a Job Aid Tool with a focus on the EOD Domain. The intent was to use the Job Aid Tool to filter procedural documentation to help an EOD technician dispose of different types of IEDs based on a set of conditions relevant to that EOD technician's situation.

APPROACH

DRC created a prototype that utilizes a procedural description within the EOD Domain that contains several subprocedures. Sets of procedureal-related information are referred to as Tactic, Techniques, and Procedures (TTP). TTPs contain hierarchies of several procedures, tasks, subtasks, and steps. These procedures and everything contained below them in the hierarchy constitute the knowledgebase. The information contained in the knowledgebase is also referred to as content or data. The content is compliant with the OWL-encoded TTP Ontology. Portions of the Job Aid Delivery Tool, such as a filtering and formatting web service are created to aid (in this case, an EOD technician) in providing relevant filtered and formatted information to its users.

Job Aid Delivery software was developed to demonstrate the use of OWL-encoded TTP information. The context diagram for the prototype is presented in Figure 1.

Requirements Analysis

A Software Requirements Specification (SRS) was developed to specify the Job Aid Tool requirements. The SRS provided a description of the product and a list of requirements related to the following software areas:

- Graphical User Interface,
- TTP Ontology,
- TTP Knowledge Base,
- Filter Web Service, and
- Format Web Service.

Overarching requirements included ensuring that the Job Aid Tool used an OWL-encoded ontology and that the content (data) is compliant with the ontology.

Ontology Design

Once the requirements for the ontology were clearly defined, the ontology design phase began. A driving requirement for the dynamic job aid tool was to ensure that the ontology was "domain-agile". Procedural information was encoded as data for presentation as formatted information. The ontology was represented graphically in using Microsoft Visio (see Figure 2).

To support the concept of multiple domains, a "Domain" class was needed to represent information about the domain. The procedural information about a domain was organized into groups of information called TTP. The "TTP" class was associated with multiple formatted items, represented by the "FormattedItem" class. Formatted items are organized into tree structures by a relationship to other formatted items. Formatted items include procedures, tasks, and subtasks.

A critical requirement involved being able to tailor the presentation of the job aid information based on current conditions. For example, in the automotive repair domain, a tire changing procedure may have an associated condition of "time of day". Based on the current "time of day" condition value, the presented procedure may include a step for using a flashlight.

The conditions requirement was satisfied in the ontology by associating a "Conditions" class with TTP. Each "Conditon" has an associated "ConditionValue" class used to represent the possible values of the "Condition" including the default value. The "ConditionValue" instances are tied to the "FormattedItems" instances so that only certain formatted items are included in the display based on the current conditions.

The "Index" and "IndexTerm" classes are used to represent terms in the knowledgebase that are used to dynamically construct an index view. The index view allows the user to quickly navigate to information on a particular topic, similarly to a book index.

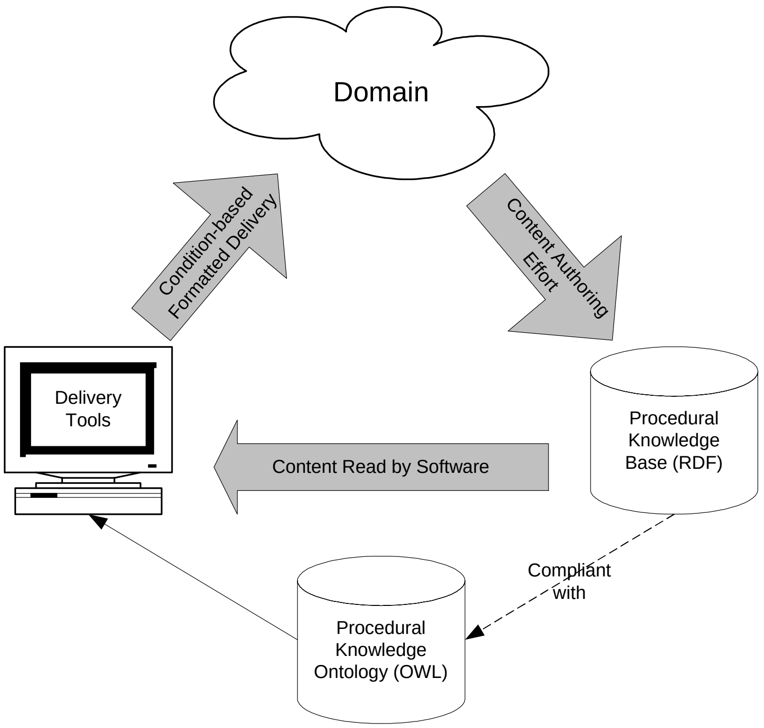

Domain
Condition-based Formatted Delivery
Content Authoring Effort
Delivery Tools
Content Read by Software
Procedural Knowledge Base (RDF)
Compliant with
Procedural Knowledge Ontology (OWL)

Figure 1. Job Aid Tool Context Diagram

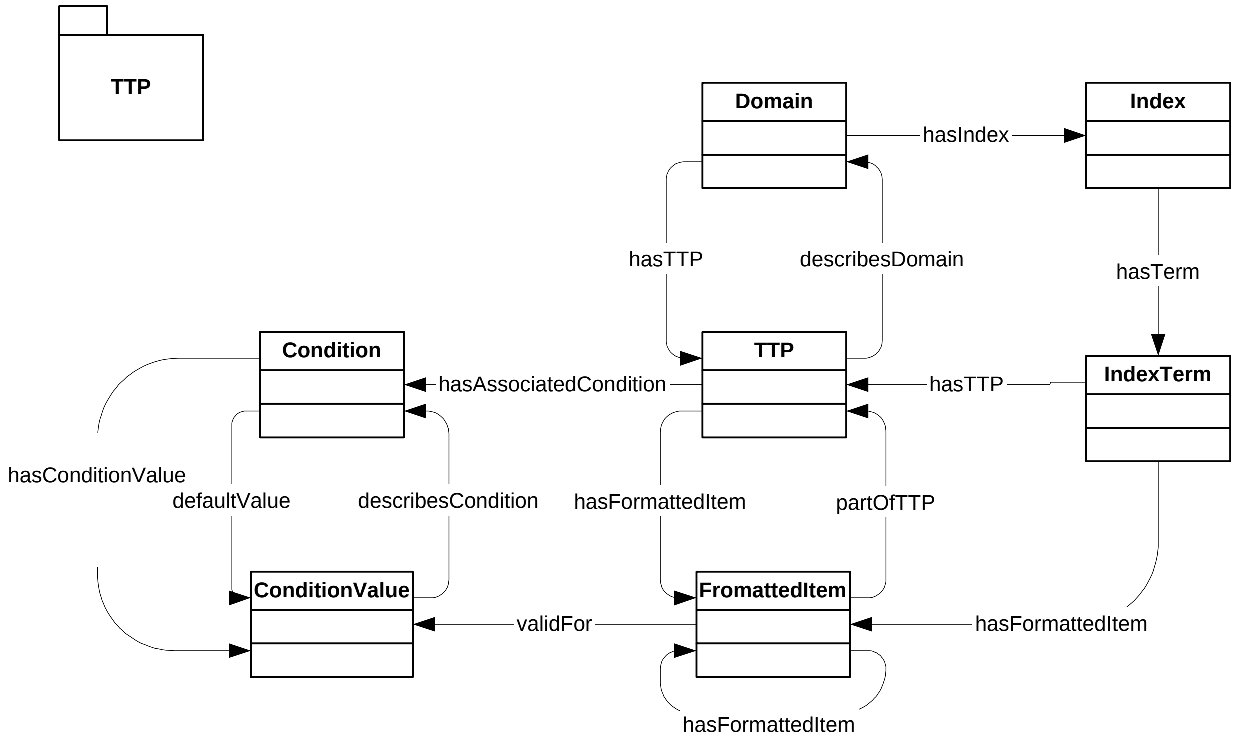


Figure 2. Ontology Design

Prototype Design

The design of the Job Aid Tool was initially captured and scoped with a script that outlined a use case scenario of the tool as defined by the software requirements document. The major components of the design include:

- Domain navigation,
- Condition-based content filtering,
- TTP Knowledge Base (KB), and
- Content formatting.

The content was assumed to exist in a Knowledge Base. The domain navigation component allows for the traversal of any domain for the purpose of selecting TTP KB desired by the end users. The KB can then be filtered based on conditions that have been predefined for the KB in scope. The KB, whether or not it has been filtered, is then formatted in real time based on the end users selection of predefined formats.

Prototype Implementation and Demonstration

The approach to prototyping the Job Aid Tool was to:

- Encode the TTP ontology into OWL code,
- Mark up existing EOD TTP content to be compliant with the TTP Ontology,
- Create a filter web service that utilizes user selected conditions to extract appropriate procedures from the content,
- Create a formatting web service to allow the user to format the content in the view that is desired, and

The TTP auto authoring capability is designed and developed to be domain independent. A demonstration was developed in order to contextualize this capability within the Joint Explosive Ordnance Disposal (JEOD) domain and provide a capability for just-in-time training for JEOD war fighters needing to follow procedural steps to “render safe” encountered IEDs.

RESULTING SOLUTION

The resulting job aid tool design is presented in Figure 3 and described in the following subsections.

Solution Overview

The solution is deployed in three parts: The Job Aid Tool web site, the Filter TTP XML Web Service, and the Format TTP XML Web Service. Once deployed, the end user is able to access the website via a platform-independent browser client. The interaction between the web browser client and the deployed solution is depicted in Figure 3. The web client provides the end user with the ability to navigate the domain to select the desired KB. The Uniform Resource Locator (URL) identifying the KB is used to filter and/or format the KB. The web client enables the end user to select conditions to filter the content in the KB. Behind the scenes, the web client makes and receives Simple Object Access Protocol (SOAP) requests as identified in steps 1 through 6 in Figure 3.

Step 1 defines the URL to the TTP KB and the URL to the file containing the conditions as selected by the end user. In this step the Filter TTP XML Web Service is invoked to filter the TTP KB based on the user-defined conditions.

Step 2 is the response to Step 1 and returns the URL to the filtered TTP KB. Both the unfiltered and filtered KBs conform to the TTP Ontology.

Steps 3 and 5 represent requests for two different types of format. The can be invoked at the end users request and in any order. Step 3 defines the format type to be applied to the TTP KB and the URL to the TTP KB which may have been filtered if so desired by the end user. Step 4 is the response to Step 3 and returns the URL to the XHTML formatted TTP KB. Step 5 defines the index format option and the URL to the TTP KB.

Step 6 is the response to Step 5 and returns the URL to the XHTML Hyperlinked view of the indexes for the TTP KB.

Application developers can reuse the XML Web Services depicted in Figure 3 to provide new functionality or enhance the existing functionality. These TTP XML Web Services can be used independently of each other. The XML Web Services were developed and deployed using Microsoft Visual Studio .NET 2003.

Authoring Content

At run-time, the domain’s procedural information must be encoded in RDF instance files that conform to the TTP ontology. Various competing tools can be developed to populate these standard RDF files.

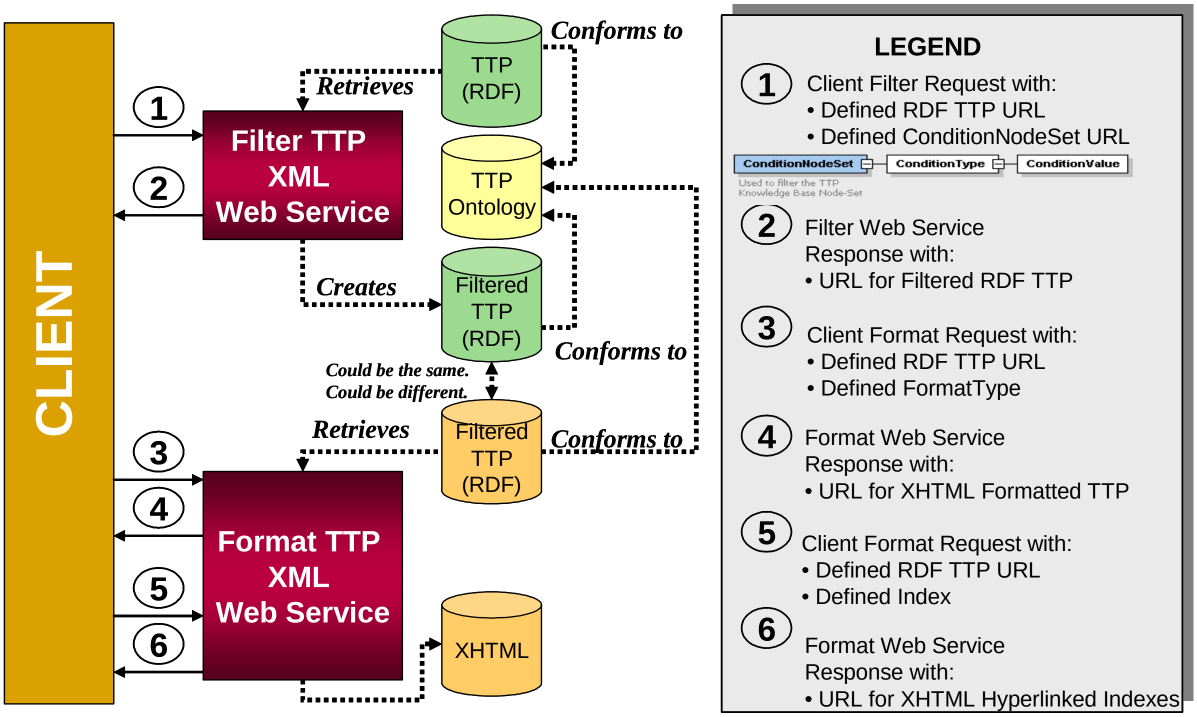

CLIENT
1
2
3
4
5
6
Retrieves
Filter TTP XML Web Service
Creates
TTP (RDF)
Conforms to
TTP Ontology
Filtered TTP (RDF)
Conforms to
Could be the same.
Could be different.
Retrieves
Filtered TTP (RDF)
Conforms to
Format TTP XML Web Service
XHTML
LEGEND
1 Client Filter Request with:
• Defined RDF TTP URL
• Defined ConditionNodeSet URL
ConditionNodeSet
ConditionType
ConditionValue
Used to filter the TTP Knowledge Base Node-Set
2 Filter Web Service Response with:
• URL for Filtered RDF TTP
3 Client Format Request with:
• Defined RDF TTP URL
• Defined FormatType
4 Format Web Service Response with:
• URL for XHTML Formatted TTP
5 Client Format Request with:
• Defined RDF TTP URL
• Defined Index
6 Format Web Service Response with:
• URL for XHTML Hyperlinked Indexes

Figure 3. Job Aid Tool Component Interaction Diagram

As part of the DARPA effort, a Microsoft Access database application was developed to assist in the creation of the RDF files. Longer-term, a more sophisticated authoring tool would be desirable

Because of the Semantic Web's support for information being distributed over the web, multiple RDF files on multiple servers could be used to integrate the job aid content for the user.

Filtering Content

The web-based filtering application invokes the XML Web Service which performs the filtering via an Extensible Stylesheet Language Transformation (XSLT) in real time. The XSLT processes the TTP KB looking for associated conditions specified by the end user at run-time. The XSLT dynamically produces a new RDF representation of the TTP KB based on the specific user conditions at the time the instruction is needed. This capability is enabled through the use of OWL-encoded procedural KBs.

The Filter TTP XML Web Service is a stand alone service that can be consumed by any application that can communicate via TCP/IP using SOAP. It has been incorporated as a major component of the Job Aid Tool web-based application.

Behind the scenes, the Filter TTP XML Web Service performs the filtering via an XSLT in real time. The XSLT processes the TTP KB looking for associated conditions specified by the end user at runtime. The XSLT dynamically produces a new RDF representation of the TTP KB based on the specific user conditions at the time the instruction is needed. This capability is enabled through the use of OWL-encoded procedural KBs.

As a major component of the Job Aid Tool web-based application, the application must manage the interaction with the web service. Specifically, the application will define the URLs to the TTP KB and conditions defined by the end user and invoke the web service. The application will then store the returned URL to the filtered TTP KB to be used by the Format TTP XML Web Service in the next section.

Formatting Content

The Job Aid Tool provides the ability to format either filtered or non-filtered procedural information to meet the user's particular needs.
We intentionally separated the format web service from the filter web service because the content is format independent. Any additional content enhancement to this Job Aid Tool can now be content focused.

With this design, formatting is not dependent on the end user's platform. The user can select between the following views depending on their needs:

- Checkbox – where each formatted item has a checkbox next to it so that EOD technicians may check off those procedures completed,
- Hyperlink – hyper linking each procedure with all subprocedures,
- Expanded – a full text view of entire TTP, and
- Indexed – an alphabetical index of all applicable words related to a certain level.

Connecting to Domain-Specific Ontologies

Content authors, with subject matter expertise in their domain, are able to define associations to external ontologies in order to contextualize the information they are capturing semantically. The "rdfs:seeAlso" property within RDFS provides the means within OWL instance files to reference other resources that may provide additional information. By enabling content authors to capture these semantic relations, the information encoded can be contextualized by semantic web agents; enabling a web that is machine readable.

RELEVANCE

The Job Aid Tool provides benefits not only for the EOD community, but to any group that wishes to filter or format any TTP that is compliant with the TTP Ontology.

The military impact of this tool is in providing EOD technicians with a "just-in-time" TTP presentation capability for handling IEDs. Autoauthored on-the-fly information retrieval is faster and cheaper because the content is already tailored and can be reused.

The training community benefits in the reduction of development costs for Interactive Multimedia Instruction (IMI) in terms of hours because developers who create IMI can focus on content rather than format.

SUMMARY

Through the use of ontologies, the Semantic Web will become the next evolution of the World Wide Web. The ontology web language is becoming the standard for web service markup, providing a more consistent way to link various pieces of information with one another.

DRC has demonstrated this with the Job Aid Tool. In using this tool, and EOD technician can have access to enormous amounts of procedural information that relate to a set of specific conditions, those that are relevant to the EOD technicians own situation.

The ability to filter procedural information could save the EOD technicians time and provide them with "just-in-time" information retrieval in rendering an IED safe.

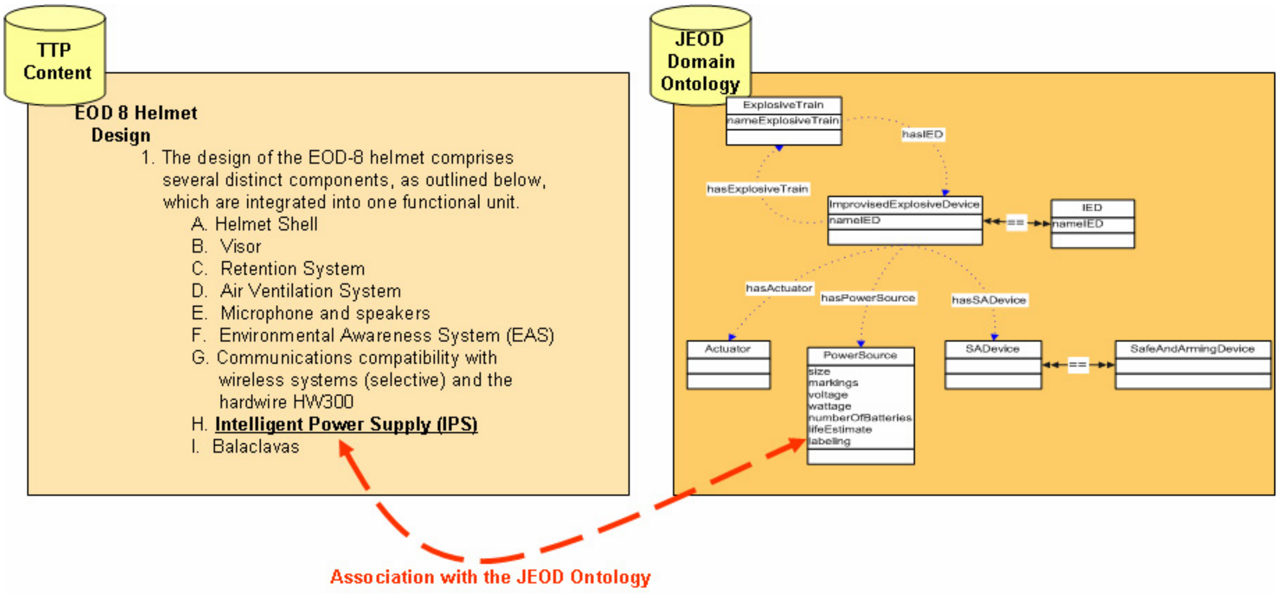
REFERENCES

Advanced Distributed Learning, (2004). *Sharable Content Object Reference Model (SCORM) 2004 Overview*. Retrieved June 25, 2004, from http://www.adlnet.org/

Berners-Lee, et al, (2001). "The Semantic Web". *Scientific American*, Retrieved June 25, 2004 from http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21

DARPA Agent Markup Language, (2001). *DAML+OIL (March 2001)*, Retrieved June 25, 2004 from http://www.daml.org/2001/03/daml+oil-index.html

Department of Defense, (1995). *Performance Specification: Data Base, Revisable -Interactive Electronic Technical Manuals, For The Support Of MIL-PRF-87269A 1 October 1995*. Retrieved June 25, 2004, from http://navycals.dt.navy.mil/cals/documents/sp87269A.pdf



EOD 8 Helmet Design

1. The design of the EOD-8 helmet comprises several distinct components, as outlined below, which are integrated into one functional unit.
   - A. Helmet Shell
   - B. Visor
   - C. Retention System
   - D. Air Ventilation System
   - E. Microphone and speakers
   - F. Environmental Awareness System (EAS)
   - G. Communications compatibility with wireless systems (selective) and the hardwire HW300
   - H. **Intelligent Power Supply (IPS)**
   - I. Balaclavas

Figure 4. Association with the JEOD Ontology

Department of Defense, (2001). *Department Of Defense Handbook Glossary For Training (Part 4 Of 5 Parts) MIL-HDBK-29612-4A*. Retrieved June 25, 2004, from http://www.dtswg.org/PDF%20Files/-4%20hdbk.pdf

Lacy, Lee W., et al, (1992). Autoauthoring Procedural Training Hypermedia. *Proceedings of the Interservice/Industry Training Systems and Education Conference*, (I/ITSEC '92), San Antonio, Texas, November 2-4, 1992.

Mears, John E., et al, (1989). *Organization of Technical Information for Electronic Delivery*, U.S. Army Research Institute for the Behavioral and Social Sciences, Research Report 1543, October 1989.

World Wide Web Consortium, (2004). *OWL Web Ontology Language Overview*. Retrieved June 25, 2004 from http://www.w3.org/TR/2004/REC-owl-features-20040210/

World Wide Web Consortium, (2004). *Resource Description Framework (RDF) Schema Specification 1.0*. Retrieved June 25, 2004 from http://www.w3.org/TR/2000/CR-rdf-schema-20000327/