Extending XML in the Enterprise

Jon **Parsons**

November 2005

Abstract

This presentation explores how recent advances in user interfaces have blurred the once clear distinction between structured and unstructured data. It examines how these tools can be used to empower a new class of user to participate in an XML workflow and a managed content environment.

Table of Contents

1. Introduction

Today, XML, the Extensible Markup Language from the W3C, has been widely adopted as a standard data format for many purposes within the enterprise. Originally created within a publishing-oriented environment and emerging from the Standard Generalized Markup Language (SGML — ISO 8859:1986), XML was designed to declare and enforce structure on content. It met the needs of publishing communities of two kinds: 1) where content is the product (legal publishers, STM societies, standards bodies, directory and reference publishers) and 2) where content supports a product (aerospace and defense technical documentation, automotive repair manuals, owner's guides, warranty information, etc.).

In both cases, the content in question was essential to the business of the enterprise. Either it constituted the product itself or it supported the product in a mission-critical way. Without the information required on how to maintain an aircraft, that aircraft can't fly. Without complete user information for an automobile, that vehicle cannot be sold.

Given the high value of content to the two communities in question, considerable investment in data analysis, automated tools, and human attention were warranted in order to eliminate redundancy and rework, achieve automation and reuse, and position the publisher to be able to meet the growing requirements of multichannel delivery

Within these communities Document Type Definitions (DTDs) developed that were highly complex because the nature of the created content was complex. Very granular markup became the norm. Having the content explicitly tagged in a standard manner preserved options for formatting, delivering to new devices, and sharing the content across company boundaries with partners and customers.

Because the DTD's (and the schemas that evolved from them) were complicated, tagging content was difficult. Authors and editors had to be conscious of the structure of the content they worked on. They became responsible for tagging correctly, validating the markup using software tools and correcting errors when they were identified.

Specialized tools were developed to support these activities. Structured editors became essential because they provided guidance on what tags were valid in a particular context, enabled users to check their work with a validating parse, helped in locating errors in markup, and provided a general support environment oriented to a particular DTD or schema. These tools became the default tools of the trade for anyone working with generic markup and content destined for the repository of corporate information assets.

The early adopters of generic markup were highly skilled in information analysis and design. They took to the challenge of thinking about content from a disciplined structural perspective. And they saw clearly the value of creating content that could be reused, delivered in a variety of ways, and processed by tools from many vendors and/or an open source community.

Once the benefits of generic markup were understood, it was only natural that people wanted to extend these benefits beyond the confines of the communities of the early adopters to a broader population of content creators. Publishers who received content authored by technical experts, for instance, wanted to encourage the authors who supplied their content to give them marked up content tagged in a standard format to achieve a more efficient workflow.

2. The Problem

It was here that the wider use of XML met a major obstacle. Only a few specialists care about the structure of information. Hardly anyone wants to tag content when typing in a word processor. Once people have become used to the (relative) freedom of content creation in a word processing environment and have become enamored with change tracking, spell checking, grammar suggestions, search and replace capabilities, and the ability to control how a document looks by changing fonts, specifying bold or italic formatting, they resist giving up the familiar interface for something new, especially something oriented to such daunting objects as tags from an XML schema.

Every major effort to extend the use of XML encounters this cultural issue. Groups whose roles had not focused on structure see the surfacing of structure in markup as burdensome, threatening, and boring. The downstream benefits of better control over content do not seem as important as the added complexity, increased number of keystrokes, and intimidating debugging procedures that the new approach appears to bring.

For a decade and a half now, we have lived with the distinction between structured and unstructured information, between those types of content that warrants the expense and investment at many levels required to author, maintain, review, and deliver structured content on the one hand and those other types of content authored and delivered in word processing tools or in some proprietary format on the other. The great bulk of content within an enterprise falls within the unstructured domain, because word processors were universally available, people did not want to think about structural markup, and, for the most part, there was not sufficient reason to force the issue of using XML for content that was not mission critical to the business.

3. Two Advances

Two major advances have occurred which suggest that the distinction between structured and unstructured information may now be blurring. This blurring creates an opportunity to extend XML authoring to a wider class of end user than was possible before. The two changes that open this opportunity are:

- Adoption of XML by the IT industry at large
- Availability of simpler user interfaces for authoring content in XML

To a large extent the second advance flows from the first.

Because XML is (nearly) universally adopted as a standard data format for IT, vendors of all types of applications are supporting it. It is at the heart of the Web, fundamental in manipulating data coming and going from databases, and used as the communications protocol for Web Services, the foundation for a Service-Oriented Architecture. XML has become the key for managing and processing content throughout the enterprise. Because XML has achieved this status within IT, fundamental XML capabilities are now found in word processing programs from both large proprietary vendors (Microsoft's Office 2003) and in open source offerings (http://www.openoffice.org/). In addition, there has been increasing interest in a forms-based approach for capturing content in XML markup, where the form masks the details of markup from the end user (Cf. Microsoft's InfoPath, Adobe's Forms Designer 6.0).

4. The Opportunity

Given the possibility of new interfaces that support the use of XML in ways that are transparent to the end user, what new kinds of content can now be usefully written and managed in XML? The following discussion is based on conversations XyEnterprise consultants have had with representatives in several different businesses who are interested in extending the use of XML within their companies. Content typically found in an enterprise can be broadly classified into three types according to the way it is created:

- Structured editing
- Word processing
- Forms-based input

Content can also be compared using a number of common characteristics across the three environments. These classifications provide a useful way of thinking about where the use of XML could be extended.

Each of these may be examined in turn.

5. The Classic Case — Structured Editing

Structured editing describes the classic situation for a high-end XML system. The content has an involved logical structure, which is important to its presentation and use. The life span of the content is long, measured in years, if not decades, and typically it goes through a number of revisions during that time. The content is itself the product or is a prerequisite for shipping the product, and the specifics of the markup used to author the content is an explicit and important aspect of the authoring. The author knows the DTD or schema well and decides the relevant tags to use for different pieces of content.

In such an environment, content management is essential. There is a formal collaborative workflow that can be automated, the content is assembled into a common repository where it can be found and located easily, reports on the status of the content pieces can be easily obtained, and the distinction between work in progress to be reviewed in draft, and approved content ready for delivery to the outside world, is scrupulously observed. Examples are those mentioned earlier and include legal, reference, and standards publishing, product and repair information for the automotive industry, and maintenance procedures and technical information for defense systems, to name a few.

6. The Second Layer of the Onion — Word Processing

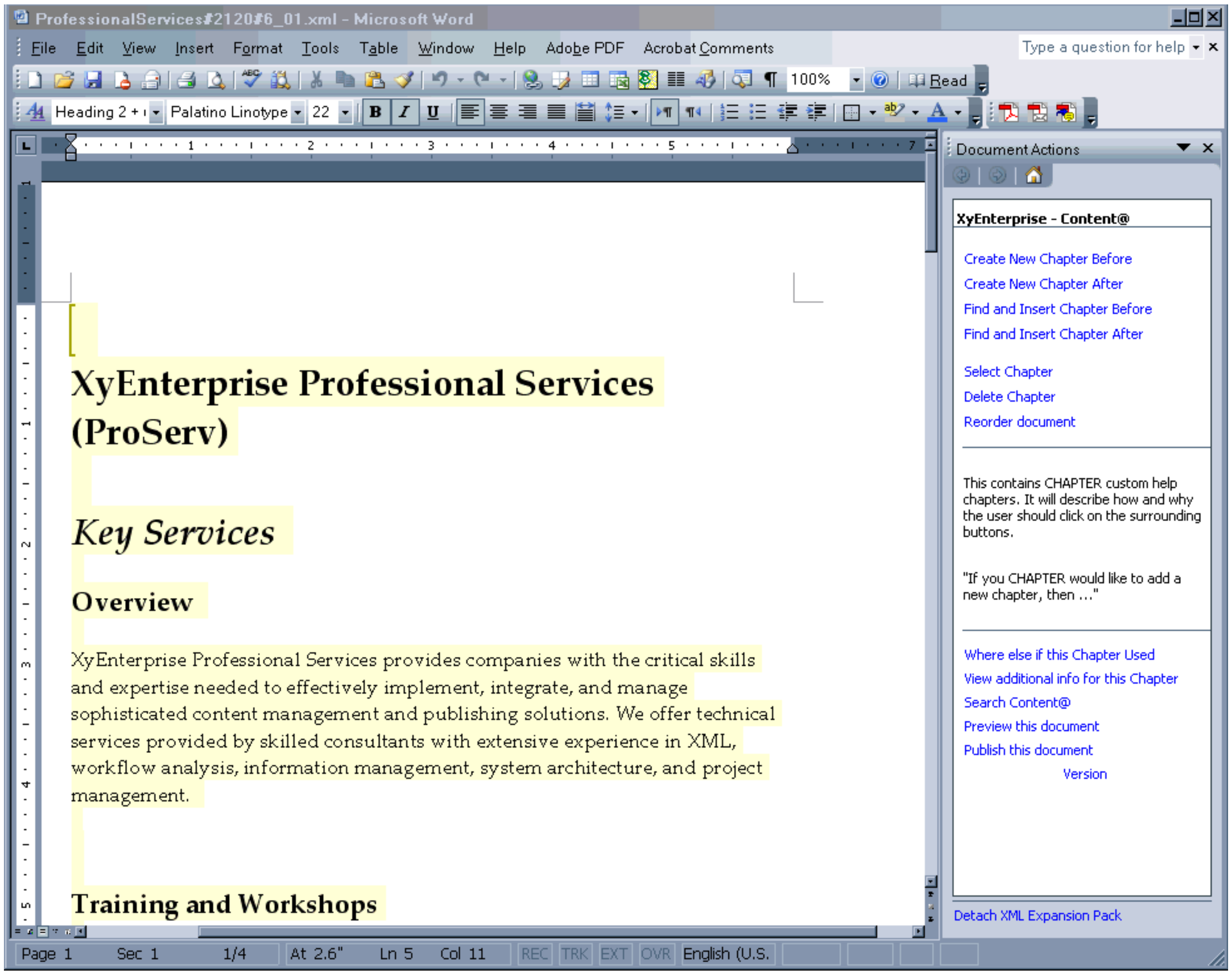
The second way of authoring extends the use of XML to a wider class of users where the logical structure of the document is fairly simple. The markup is focused upon standard office elements like paragraphs, lists, tables, and graphics grouped into domain-specific “chunks” that serve as “sections” or “chapters”. These container chunks might be identified by other names that fit the author’s context, such as “policy component” or “safety bulletin” or “deployment plan”. The point here is that the structure is limited to three or four main container elements within which the simple list of document elements is used. The author need only know a handful of meaningful content types.

In this second case the life span of the content is reasonably short, measured in months to a year, versus the many years of the previous content type. The content is still essential to conducting the business of the enterprise, but it supports that business rather than being part of the primary deliverable. The author needs to have some idea of the document structure, but only in a very limited way: asking to create or move a “container” of content. All other markup is handled invisibly behind the scenes and derives from the basic word processing activities of the author (creating lists and tables, applying bold and italics, etc.).

The need for content management in this scenario comes from the desire to reuse the container chunks. In order to “write once and use many times,” a content management system that supports that scenario is required.

Examples of situations in which this integration of word processing technology with XML authoring can be profitably applied include owner’s manuals for mass produced electronic devices, safety sheets governing the policy, use, and maintenance of rides in amusement parks, and planning documents dealing with the logistics of deploying large quantities of equipment personnel under various scenarios. Other applications might include insurance policies, financial reporting documents, or policy documents produced in a human resources department.

An example of a word-processing interface that captures XML is shown in the following figure. The authoring tool is Microsoft Word augmented by a “Smart Doc” solution oriented to simple marketing documents. Note that in this screen snap the user sees a simple Word document and can edit it with all the familiar interface conventions available. The right-hand panel supplies several choices available to the user and also offers context-relevant advice on what can be done at this point in the document. The user can also participate in a managed workflow from here.


ProfessionalServices#2120#6_01.xml - Microsoft Word
File Edit View Insert Format Tools Table Window Help Adobe PDF Acrobat Comments
Type a question for help
Heading 2 + Palatino Linotype 22
100%
Read
Document Actions

XyEnterprise - Content@

Create New Chapter Before
Create New Chapter After
Find and Insert Chapter Before
Find and Insert Chapter After

Select Chapter
Delete Chapter
Reorder document

This contains CHAPTER custom help chapters. It will describe how and why the user should click on the surrounding buttons.

"If you CHAPTER would like to add a new chapter, then ..."

Where else if this Chapter Used
View additional info for this Chapter
Search Content@
Preview this document
Publish this document
Version

Detach XML Expansion Pack

XyEnterprise Professional Services (ProServ)

Key Services

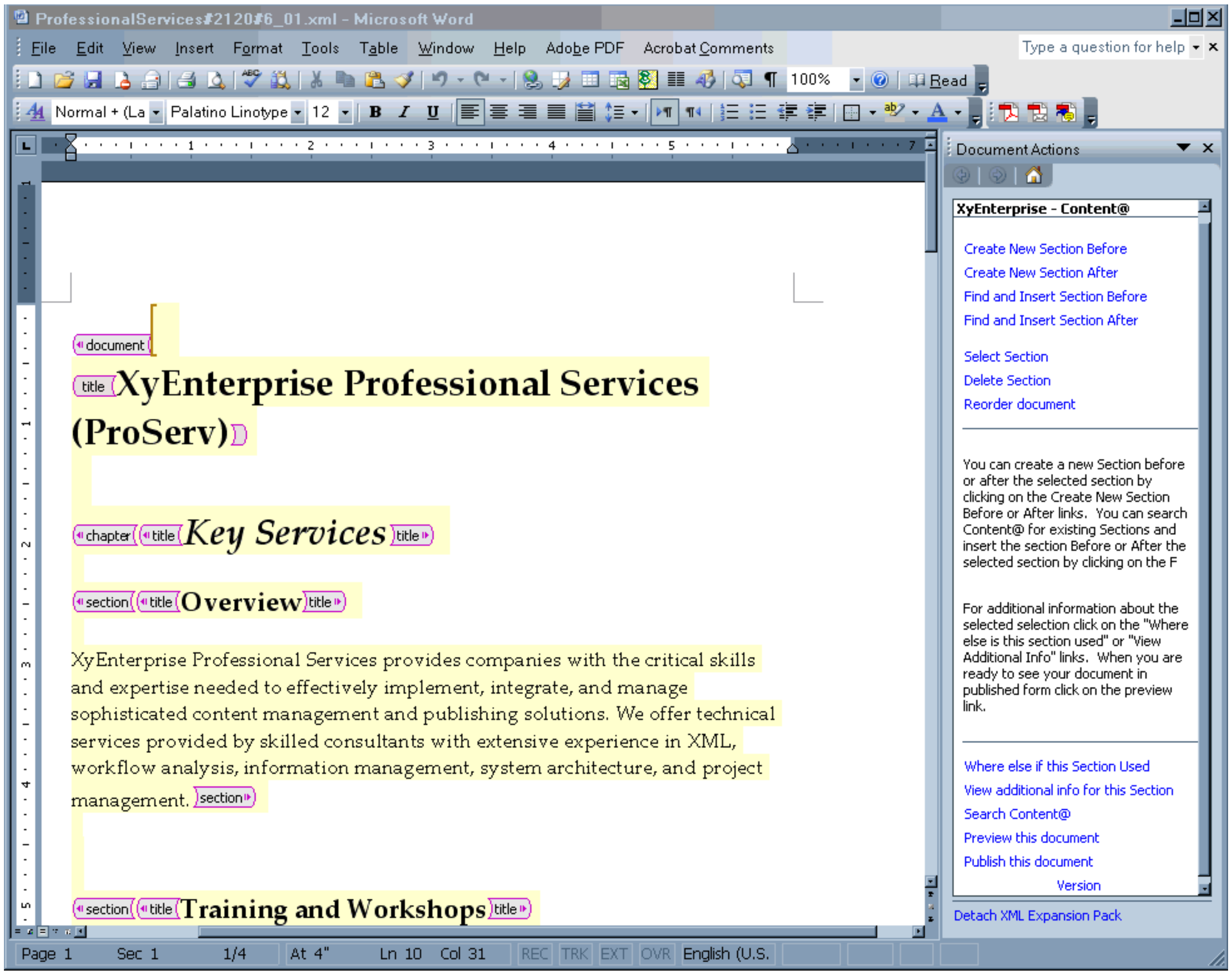
Overview

XyEnterprise Professional Services provides companies with the critical skills and expertise needed to effectively implement, integrate, and manage sophisticated content management and publishing solutions. We offer technical services provided by skilled consultants with extensive experience in XML, workflow analysis, information management, system architecture, and project management.

Training and Workshops

Page 1 Sec 1 1/4 At 2.6" Ln 5 Col 11 REC TRK EXT OVR English (U.S.

The following figure shows the same document with two changes. First, the XML tags have been turned on. There is no requirement for the user to see them — they have been turned on to show that this is in fact an XML instance being edited. The second change is that the cursor has been placed in another location and so the available choices and the help text have changed to reflect the options at this location.


ProfessionalServices#2120#6_01.xml - Microsoft Word
File Edit View Insert Format Tools Table Window Help Adobe PDF Acrobat Comments
Type a question for help
100%
Read
Normal + (La
Palatino Linotype
12
Document Actions
Page 1 Sec 1 1/4 At 4" Ln 10 Col 31 REC TRK EXT OVR English (U.S.

document

title XyEnterprise Professional Services (ProServ)

chapter title Key Services title

section title Overview title

XyEnterprise Professional Services provides companies with the critical skills and expertise needed to effectively implement, integrate, and manage sophisticated content management and publishing solutions. We offer technical services provided by skilled consultants with extensive experience in XML, workflow analysis, information management, system architecture, and project management. section

section title Training and Workshops title

XyEnterprise - Content@

Create New Section Before
Create New Section After
Find and Insert Section Before
Find and Insert Section After

Select Section
Delete Section
Reorder document

You can create a new Section before or after the selected section by clicking on the Create New Section Before or After links. You can search Content@ for existing Sections and insert the section Before or After the selected section by clicking on the F

For additional information about the selected selection click on the "Where else is this section used" or "View Additional Info" links. When you are ready to see your document in published form click on the preview link.

Where else if this Section Used
View additional info for this Section
Search Content@
Preview this document
Publish this document
Version

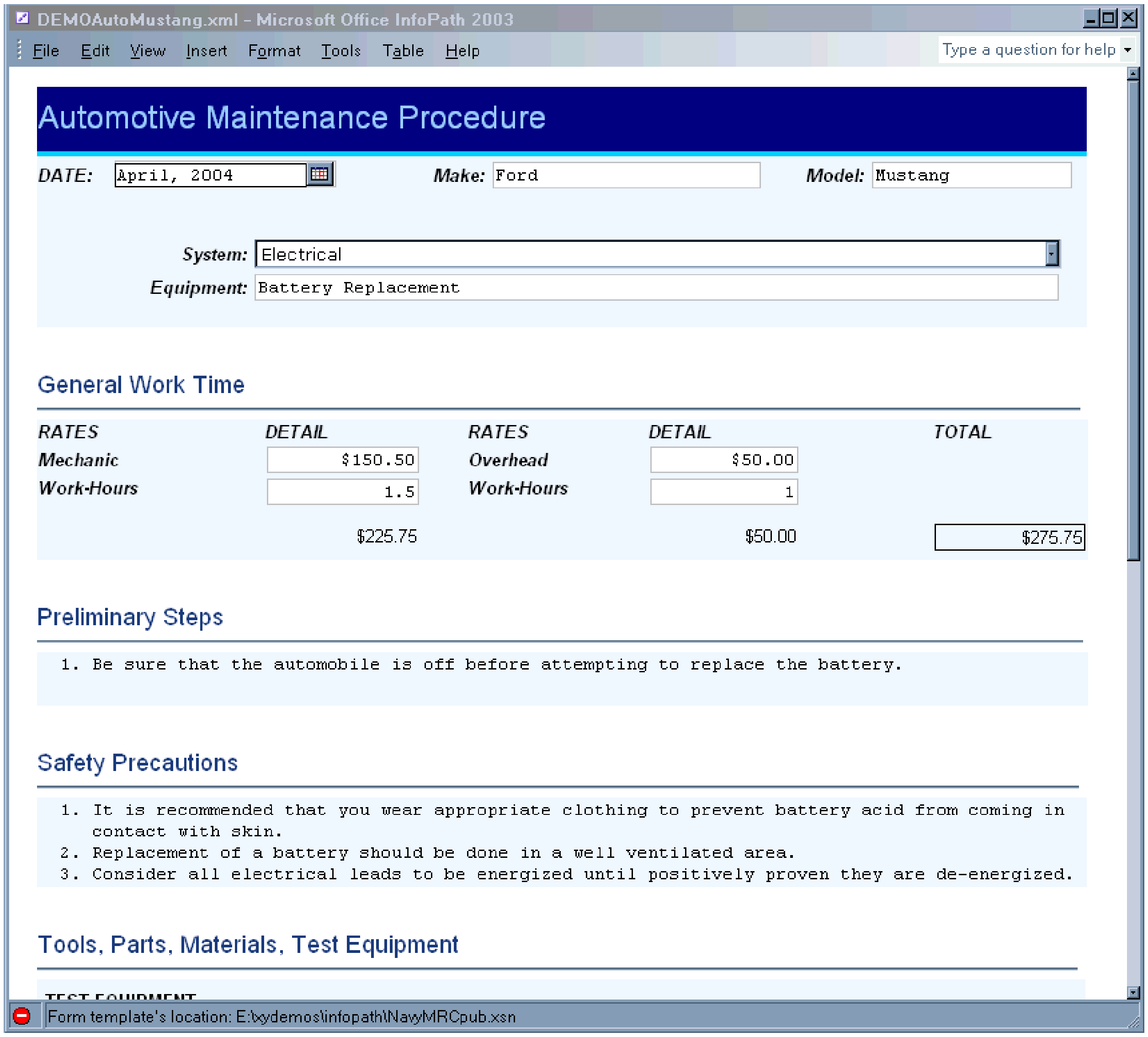
Detach XML Expansion Pack

7. The Third Layer — Forms-based Input

In the third instance, the content is largely, but not exclusively, fielded data stored in a database. Some amount of free text is also included, and the content can contain many graphics as well. The form does all the work in capturing the structure and enforcing the markup of the content. The user doesn't need to think about the kind of document he or she is working on or how it might be structured. The life span of the content is short, often measured in months, so the content is frequently updated. Content management provides the coordination needed for versioning and deploying this content over a large number of sites dispersed around the globe. A CMS system can automate the creation and delivery of these documents and provide an efficient way for a team of authors to keep content current when it is describing a large number of products and product variants marketed to particular market segments across many geographical regions.

Examples of situations in which the forms-based approach is useful include government administrative agencies with a large number of forms that need to be kept current, and global corporations who must produce consistent, coordinated marketing materials for variants of multiple products targeted to a variety of market segments. Interestingly the notion of "form" is stretching to include more than just facts and numbers placed in a field. Narrative prose sections can be included as part of the form. In some cases, it is useful to have a short, medium, and long description that can be chosen as appropriate for a particular delivery situation on the Web or in print.

An example of a forms-based interface developed for repair procedures is shown in the following figure. The authoring tool is Microsoft's Infopath. The numerical data are filled in automatically from an external database and the author is responsible for describing the steps to take. A file containing the procedure marked up in XML is created when the form is saved. This file can then be processed and managed by other XML-based applications.


DEMOAutoMustang.xml - Microsoft Office InfoPath 2003
File Edit View Insert Format Tools Table Help
Type a question for help

Automotive Maintenance Procedure

DATE: April, 2004 Make: Ford Model: Mustang

System: Electrical

Equipment: Battery Replacement

General Work Time

RATES	DETAIL	RATES	DETAIL	TOTAL
Mechanic	$150.50	Overhead	$50.00	
Work-Hours	1.5	Work-Hours	1	
	$225.75		$50.00	$275.75

Preliminary Steps

1. Be sure that the automobile is off before attempting to replace the battery.

Safety Precautions

1. It is recommended that you wear appropriate clothing to prevent battery acid from coming in contact with skin.
2. Replacement of a battery should be done in a well ventilated area.
3. Consider all electrical leads to be energized until positively proven they are de-energized.

Tools, Parts, Materials, Test Equipment

Form template's location: E:\xydemos\infopath\NavyMRCpub.xsn

8. Conclusion — Five Key Advantages to XML

Clearly the evolution of technology has opened up new situations in which XML can be used to author, maintain, and deliver content. This does not mean that there is some magic bullet that suddenly enables everyone to use XML for everything. Human nature has not changed. It does mean that there are significant, measurable paybacks for extending XML within the enterprise in areas that might not have been possible just a few years ago. The advantages of using XML in these new applications include:

- Enabling efficient and automated multichannel delivery — making it possible to deliver the content quickly and in a form usable for the consumer
- Enabling reuse of the content — reducing redundancy and rework, enabling a “write once, use many times” scenario
- Managing complexity — enabling support of larger numbers of products produced with more specialized variants

- Supporting content creators — eliminating mistakes in markup, freeing up more time to focus on clarity, accuracy, and completeness
- Integrating with other corporate systems — often many of the fields in a form can be pre-populated, saving time and improving accuracy

Biography

Jon **Parsons**
Product Marketing
XyEnterprise [http://www.xyenterprise.com/]
Reading
Massachusetts
United States of America

Jon Parsons has over 20 years experience automating the creation, management, and delivery of content in multiple forms. Currently he directs product marketing for the Content@ content management system and XML Professional Publisher at XyEnterprise. Prior to XyEnterprise, he was a writer, editor, tools developer, and publishing consultant for Digital Equipment Corporation. Long an advocate of generic mark-up and an enthusiast for XML, he has served on the Board of Directors of OASIS, the Organization for the Advancement of Structured Information Standards and is a frequent speaker at trade shows and industry gatherings.