To: Chris Lindgren

From: Linus Chan

Subject: DITA topic model design rationale

Date: 12.21.2015

DITA Topic model design - The comment element - best practices

In the long run, the comment element may prove to be the most used element in all programming. As such, my DITA topic model is designed around the best practices that are used internationally with the comment element.

The comment element annotates the source code of computer programs. It renders the text within its carets inert. These elements are often added to make code easier to understand, but there remains other best practices that are used prolifically by programmers across the board.

I designed this DITA topic model to give three task topics to the single concept topic of commenting. I intend this document to help the average beginner understand how commenting works and the best practices that are being presently utilized.

These are the three best practices that I go over in my DITA information structure. The comment element is used to edit within the source code, add metadata and create psuedocode. Each one of these tasks are taught and used in common code pedagogy.

When writing this document, I went over multiple design changes. My DITA map was initially much more complex. I wanted to create a document that had one overarching concept task topic that branched into three more concept topics. Then, I would create three independent task topics for each parent concept topic. However, the topic of ‘best practices’ helped make the entire category much more concise.

Instead, I structured my DITA map to become very straightforward. I followed two models. The first modle came from the DITA Best Practices by Bellamy. I utilized the framework that Bellamy uses to ‘install the espresso machine software’, on page 27. This framework enabled me to structure my task topics in ways that described prerequisites in writing source code. It also gave me a concrete concept to follow after in educating beginners in Commenting tasks.

The other model that I followed was the Monkey DITA example. The Monkey DITA example provided the initial code basis by which I could build my own DITA code. It was structured in a three part topic, something that I used in my own information structure.

Another distinct design decision that I intentionally made was the inclusion of step by step commands in all of my task topics. As opposed to broad concept topics, I wanted these best practices to be topics that could be followed. Best practices should be imitated, not just read. There are many ways to use the comment element, but DITA enables me to create a step by step document that can be easily utilized as well as created.

All of my task topics link back to my initial comment title as best practices. They help describe how the comment element in source code functions. They also give a glimpse towards the best practices that are currently used in coding. Most of all, it gives the reader a glimpse of how the comment element is currently being utilized by programmers.

I also added a link towards my historicizing website in the first concept topic about the Comment element. This draws the reader all the way up the hierarchy to a broad overview of what the comment element is. This DITA document helps describe what the comment element is, how it functions and what it can best be used for.

The comment element will remain a crucial component of code history and code practices. It gives much-needed authorial intent, describes important information and is a useful tool to edit with. It is the means by which authors can communicate with each other across the boundaries of space and time. Therefore, it is crucial that any beginning programmer at least acquaints himself with the best practices of the comment element.

In line with the iterative revision planning, I will now describe the history involved with the creation of this DITA document.

The Short Desc element is used throughout this entire DITA structure. The short desc element is written to succinctly describe the purpose of the topic. It briefly summarizes the main point of the topic. For particular project, the short desc element were used to give the reader a glimpse of what step by step instructions they were going to receive. Likewise, I also utilized the short desc element to provide a focus on the user goal. When I initially wrote the document, it was designed to be a concept ‘information’ dump. However, by revising this element, I was able to hone and focus the user’ goal to perform specific tasks such as editing with the comment element.

Another concept that I utilized throughout my document was the use of a conref tag. This document is used predominantly to instruct, So I utilized the conref tag to reference how the concept of commenting. The most important thing to be taught was the simple practice of commenting. All the tasks that followed hinged on that specific ability set. Therefore, I used the conref in each one of my tasks to reference again and again just how these comment tags were created.

In the creation of the document, I also came to the realization that images seeded throughout the document would aid the reader. While I had initially planned to create a document that was focused on textual information, I realized that a depiction of the tasks in action would aid the user and the goals that they wanted to achieve. The act of editing, especially through the use of comment tags is almost second-nature to proficient programmers but would be incredibly confusing to the beginner.

This driving purpose of teaching a specific task honed my ability to refine and revise this DITA document.

On a holistic basis, this DITA document is not overly complex. However, it accomplishes what it seeks out to do properly. With its system of organization and conref call backs to specific skills, this DITA topic at least enables the user to effectively use three best practices with the comment element.