Applying JAVA-Triggers for X-Link Management in the Industrial Framework

Abraham Alvarez and Youssef Amghar
Laboratoire d’Ingénierie des Systèmes d’Information, INSA de Lyon, France, 7 avenue Jean Capelle, F-69621, Villeurbanne
Tel: +33 4 72 43 85 88, Fax: +33 4 72 43 87 13, aalvarez@lsi.insa-lyon.fr, amghar@if.insa-lyon.fr

ABSTRACT

In the industrial context, the life cycle of a document plays a central role describing the “steps out” of a product. Structured or semi-structured, these documents are related by different kinds of links. User performs some manipulations like: creation, edition, suppression and querying under a multi-user environment, risking possible destruction or the alteration of the document’s integrity. A classical impact is the infamous “Error 404 file not found”. However, the user needs a notification Alert Mechanism to prevent and warrant the coherence of manipulations over all the life cycle process of a product. The main objective of this paper is to provide a generic relationship validation mechanism to remedy this shortcoming. The paper focuses in referential link integrity aspects. We believe that by the combination of some standard features of XML specifically XLL specification as a support for integrity management and JAVA-Triggers approach as an alert method.

INTRODUCTION

In this study, the coherence integrity problem means that a link depends on the “state” of the referenced document. When documents are edited or revised, locations can probably be altered, with the potential risk of destroying the record of a linked text. The main purpose of this paper is to provide a generic relationship validation mechanism to keep the links references in “coherent state” over all process involved. Therefore, some standard features of XML and highlighted in the XLL specifications (x-link & x-pointer) presented by XML-Linking working group, will be considered in this work as a support for integrity management and JAVA-Triggers approach as an alert method.

Following this introduction, the related work section specially analyzes the different approaches for managing link integrity, in addition, presents briefly a survey for the similar application domains as: hypermedia, hypertext systems and digital libraries referring to link integrity problem. The motivation and background of this study are explained in section 3. Section 4 illustrates and examines with an example the validation process. Section 5 concerns the algorithm and the application presentation. Finally, a conclusion and future perspectives are discussed.

RELATED WORK

In the last few years, a lot of time has been dedicated to the problem of maintaining links. Different approaches for managing link integrity are categorized and exposed in this section. Two main solution categories are discussed concerning the referential integrity problem: the preventive and corrective solutions. In the recent surveys, Ashman [Ashman 1999] and [Asham 2000] present some preventive solutions: Forbidding changes, this strategy, the modification or deletion of documents is not authorized. Another preventive solution is proposed by Davis [Davis 1999] introducing a concept called “publishing” solution, which consists in preventing the content reference problem prohibiting users from updating documents once they have pointers into them.

In the context of link attributes [Oinas-Kukkonen 1998], proposes to integrate link attributes like semantic link types and link keywords. Link attributes properties associated with links provide knowledge about interrelationships between pieces of information. This preserves the context of information units and increases the local coherence of an information collection link; attributes also provide the user a way to know or preview the target before activating the link, [Thüring and al. 1995]. Finally, DeRose presents the most up-to-date overview of the Xlink, Xpointer, and Xpath recommendations, explaining important features of hypertext systems not seen previously, e.g. one-to-many links, out-of-line or externalized links, and calculations of links positions [DeRose 2001]. A literature survey related to links history is presented by Lewis and al, in [Lewis 1999] the authors make a distinction between navigation and retrieval information handling. Nowadays, in the area of hypermedia research systems, we have found powerful systems MicroCosm [Hall 1993], Hyperware [Maurer 1996], linking Systems Dexter [Halasz 1990], [DeBra 1999], Intermedia [Haan and al. 1992]. However, still constrained by the limitations imposed by HTML. The OpenDoc architecture [Killpack 1994] is an answer to satisfy the interoperability problems in hypermedia system.

MOTIVATION

In the industrial context, the on-line technical documentation is a support describing and defining department activities overall manufacturing process, executed by several departments such as designing, manufacturing, maintaining and assembling areas. Considering this context and using Intranet technology to deliver, share and connect documents to other departments, any change in the on-line documentation can impact over other department activities, notably in the whole life cycle of a product’s development. Under the previous context, the main motivation of this study is to provide a generic relationship mechanism to maintain the integrity of links. The technical documentation projects in organizations require coherent manipulations on the overall process involved. The Figure 1 depicts this scenario.

Figure 1: Major phases of product manufacturing

This paper appears in Issues and Trends of Information Technology Management in Contemporary Organizations, the proceedings of the Information Resources Management Association International Conference. Copyright © 2002, Idea Group Inc.
Technical Documentation

Definition: technical documentation is the creation, control, delivery, and maintenance of distributed information across the extended enterprise and a network that includes sources and users. The technical documentation is not a recent concept in the Intranet environment [Albin 1996]. The on-line technical documentation serves as an interface between the user and the product.

World Wide Web, Intranet, hypertext links and hypermedia systems, are key concepts in the way we read, write and access information and have created a radical way of reading and writing office documents.

Using Intranet technology, document workflow relates to the product’s life cycle. The set of relevant information for a particular task or a set of related tasks during the life cycle of the product, e.g. manual specifications for a design department, modification instructions for maintenance department and so on.

- Users who create, use and maintain the information set are working under Intranet environment. User manipulations are concurrent.
- All documents are interrelated by links over all process
- Changes can impact notably in other department activities.

Background of Active Database and Triggers Approach

This section shows the basic knowledge of Active Databases and Triggers approach. An important functionality supported by many of the most recent relational and object-relation database systems is represented by triggers, that enhance the database with reactive capabilities and which can be used to support integrity constraints i.e. to maintain the integrity of the data, and maintaining long-running transactions. Event-Condition-Action (ECA) formalism, where the Event specifies when a rule should be triggered, the Condition is a query that is evaluated when the Event occurs and the Action is executed when the Event occurs and the condition is satisfied.

The notion of trigger appeared in the seventies, and has been generalized with the notion of active rule that is based on the Event-Condition-Action (ECA) formalism. The semantic of an ECA rule is noted as follows: when an event E is produced, if the condition C is satisfied, then the action A is executed. Actions are initiated by the DBMS when appropriate events occur, independently of external requests. These rules allow database designers to specify the active behavior of a database application that provides the enforcement of database integrity. In the literature, several approaches were proposed to integrate active concepts into databases.

Several commercial database management systems (DBMS) include event/trigger mechanism that has been proposed by [Kotz 1988], such as the Postgres rule system [Stonebraker90], Starburst’s production and alert rules [Lohman 1991], Ariel’s production rule system [Hanson 1989], the ECA model of HiPAC [Dayal 1988], and the event-action EA model of Ode [Gehani 1992]. Few researches have noted as follows: when an event E is produced, if the condition C is satisfied, then the action A is executed. Actions are initiated by the DBMS when appropriate events occur, independently of external requests. These rules allow database designers to specify the active behavior of a database application that provides the enforcement of database integrity. In the literature, several approaches were proposed to integrate active concepts into databases.

XML Specification

XLL allows XML tags to refer to other XML or HTML documents, as well as specific elements or data within those documents. In addition, X-link specifies a set of attributes in X-link namespace, one can add element tag references to other documents, while x-pointer specifies a syntax to select specific elements within a document.

The mechanism to maintain the link integrity is described in this section. After the hypertext document creation, the next important steps are: validating the document by a classical parser for XML-documents, making a recognition of links and finally the referential links validation.

Step 1. Parsing Validation. This step consists in verifying the syntax and structure of documents before applying the process of validation (step 3), we are using an evaluation copy of XML Spy 3.5 developed by IBM. Two kinds of parsers are available for the validation of documents: a “non-validating processor” that makes sure that xml documents are “well-formed-syntactically” correct. The second, is a “validating processor” that checks for much more, particularly the document structure against a set of declarations to be sure that the structure contains all the parts required by these declarations [Laurent99]. Unfortunately, both parsers are far less performing when it comes to checking the relationship coherence of structured documents. The importance of this step is to reduce potential syntax errors and check the document structure, for a more powerful analysis to the next step.

Step 2. Links Recognition. In order to keep the reliability of recognition, let us consider only “well-formed-syntactically” documents (step 1). The next step is to detect the tags between the documents, this means, to identify the type of link. The simple and extended links are the classical links embedded in most of the structured documents. The simple link is used to “surf” or to navigate between the documents (transversal notion), this link is unidirectional. Extended links are the classical links embedded in most of the structured documents. The simple link is used to “surf” or to navigate between the documents (transversal notion), this link is unidirectional. Extended links are multi-directional, allowing a multi-directional navigation and reference. The other types of links are not considered for the moment.

Step 3 – Referential links validation. The existence of a link is asserted by a linking element: Linking elements are reliably recognized by software applications. When the user inputs the URL or file location, the application searches within documents all occurrences of types HREF, XREF, etc. The application searches within documents all occurrences of types HREF, XREF, etc. The simple and extended links are the classical links embedded in most of the structured documents. The simple link is used to “surf” or to navigate between the documents (transversal notion), this link is unidirectional. Extended links are multi-directional, allowing a multi-directional navigation and reference. The other types of links are not considered for the moment.

To have a general panorama of x-link properties, a summary is considered. In this way, XLL language has two types of links: internal and external links, internal links are “on-line” links (within the document), an advantage lies in the use of X-pointer, this means that we can point any portion of the document and identify the end-points of the link, that work on the tree nodes, (child, descendant, antecessor, etc) e.g. HREF=http://www.isia.insa-lyon.fr/~a-alvarez/ id/publications).child (3, item). External links are “out-of-line” (stored in a database).

An Example

From now on, we will focus on describing the example. A small technical manual was chosen to illustrate the repercussion of any change during the life cycle of a product. The main purpose of producing the on-line technical documentation is to provide an easy way to find and up-to-date the information that they needed.

In this example, three types of links considered in this paper: composition, sharing and referential links respectively.

Composition links: they represent the composition of a document structure by elements called nodes. The source corresponds to root node, the target being a component of the source. Each anchor of link is both source and target. This link is bi-directional. Moreover,
they also show their hierarchical organization in order to retrieve the
document structure. In the figure below, Manual represents the docu-
ment type and the root element of the document: an article is com-
posed by child elements called nodes, in this case, a manual is com-
posed by a {ManualTitle} containing "Operating Information Manual",
a {Body element} and afterwards a {Reference element}. {Section
I}, {Section II} are children nodes of {Body element} and so on. Each
anchor of the link is both source and target. This link is bi-directional
and implicit. Moreover, shows the hierarchical organization generally
in order to retrieve the document structure.

Reference links: they allow to establish a reference between docu-
ments and documents fragments. These kind of links are unidirec-
tional. They are often represented by known expressions such as “see
section 1”, “for more information”, etc. Information sharing link:
also to point the same information between 2 or more fragments.
This link is unidirectional and explicit.

Considering an European manufacturing enterprise of VCRs,
Video Cassette Recorders, its products are manufactured with European
electric characteristics i.e. 220volts & 60Hz frequency. All the VCRs
operating manuals have a similar structure: a manual title, a body and
its sections. Section I contains the Introduction and General Informa-
tion, Section II Basic Operations, Section III corresponds to Technical
Specifications, and so on.

A new project of expansion focuses on the North American area.
Considering this expectation and the difference of American electric
characteristics, i.e. 110volts & 50Hz. The company needs to think
about the repercussions implied by this change in the related technical
documentation across the product life cycle. Figure 2 describes the
processes of a product life cycle and the creation of a new link, when
a new requirement is claimed. If we consider that case, i.e. the quality
service of documentation requires the addition of a new annex for
previous manufacturing instructions manual.

Algorithm
The algorithm starts from an initial pass through an input url-
location or a file location (establish a connection to the specified
URL, i.e. var stringURL) in order to search any text chains containing
the referential tags as “&lt;A HREF=","XLINK:HREF"," &lt;LINK
TARGET=","<IMG SRC=","<ID REF=" into document in a recursive
way. The verified links are stored in a table (Hashtable class provided by
Java.util;).

Application
The next figure depicts the parts of running application. The
input field takes the url or file location to be verified i.e. URL:
http://lisi.insa-lyon.fr/~aalvarez/document_a.xml (1); The Out-
put window shows the status of each link and the links contained in
the same document (2), when the link address is valid, an Ok mes-
gage is displayed.

Figure 2: Product life cycle related with document generation

Figure 3: Running application
CONCLUSION AND FURTHER WORK

We have looked at the cost-benefits using Intranet Technology to distribute electronic documents during life cycle product process provided by hypertext systems. The links role is multiple (structural, navigational, spatial, etc). A reliable documentation is a significant factor in a product’s success, since users are often learning how to use and to maintain the product, its manufacturer and its supplier. We believe that companies need to document manufacturing process, the notification systems need to keep production employees up-to-date on procedural changes and approval workflows necessary for ensuring a successful technical documentation. This success depends on being able to maintain referential integrity, and hence link integrity when the documents to which links refer are altered in some way.

Link integrity and Trigger concepts are the issue considered in this paper. We have shown the different approaches to avoid the link reference problem. Two categories of solutions (the preventive and corrective) were discussed. The actual solutions are concentrated in representation aspects and are not full solutions. We have proposed a preventive mechanism to maintain the link reference integrity supported by a strong algorithm, moreover considering some XLL specifications from Xml-Linking groupwork. Further research work should contain at least two future directions: management of document and fragments versions. Another direcendnotestion is the DTD translation to schema XML.

ENDNOTES

1 OpenDoc is a registered trademark of Apple Computer, Inc
2 XML Spy 3.5 is an Integrated Development Environment (IDE) for the eXtensible Markup Language (XML), http://www.xmlspy.com

REFERENCES

Related Content

Bicluster Analysis for Coherent Pattern Discovery
www.irma-international.org/chapter/bicluster-analysis-for-coherent-pattern-discovery/112571/

Up-to-Date Summary of Semantic-Based Visual Information Retrieval
www.irma-international.org/chapter/up-to-date-summary-of-semantic-based-visual-information-retrieval/112527/

Using Maturity Model to Govern Information Technology
www.irma-international.org/chapter/using-maturity-model-govern-information/30720/

Understanding the Methods behind Cyber Terrorism
www.irma-international.org/chapter/understanding-the-methods-behind-cyber-terrorism/112557/

Stock Price Trend Prediction and Recommendation using Cognitive Process
www.irma-international.org/article/stock-price-trend-prediction-and-recommendation-using-cognitive-process/178161/