Chapter VI

XSLT: Common Issues with XQuery and Special Issues of XSLT

Sven Groppe  
*University of Lübeck, Germany*

Jinghua Groppe  
*University of Lübeck, Germany*

Christoph Reinke  
*University of Lübeck, Germany*

Nils Hoeller  
*University of Lübeck, Germany*

Volker Linnemann  
*University of Lübeck, Germany*

ABSTRACT

The widespread usage of XML in the last few years has resulted in the development of a number of XML query languages like XSLT or the later developed XQuery language. Today, there are many products like databases in the area of XML processing that support either XSLT or XQuery, but not both of them. This may prevent users from employing their favourite XML query language. In this chapter, the authors show that both languages have comparable expression power and present a scheme for translating XQuery to XSLT and vice versa. This translation scheme enables a user to employ either XSLT or XQuery for each product which supports one of these languages. They also summarize in this chapter both current and future trends and research issues and also consider those that might emerge in the common area of XSLT and XQuery and which are particular to XSLT.
INTRODUCTION

Today, XML is increasingly used to label the information content of diverse data sources including structured and semi-structured documents, relational databases, and object repositories. Furthermore, an increasing number of real XML databases have been developed. The World Wide Web Consortium (W3C) developed the extensible stylesheet language (XSL) (W3C, 2001; 2004b) and especially, its XSL transformations (XSLT) (W3C, 1999; 2007c) part, which is developed to express transformations of XML data. The W3C later developed the XQuery (W3C, 2007b) language, which was developed to express queries. However, many tools support either XSLT or XQuery, but not both.

This chapter clarifies which issues of XSLT are common with XQuery. For the researcher and developer, it is important to know which issues to be resolved are common for XQuery and XSLT, since they can determine which contributions for XQuery also work in principal for XSLT and vice versa. We clarify this by not only comparing the two languages XQuery and XSLT, but also by proposing translation schemes from XQuery to XSLT and vice versa. This clearly indicates which kind of contributions and solutions can be transferred from XQuery to XSLT and from XSLT to XQuery. After this has been determined, the open and novel issues of both XML querying languages XQuery and XSLT together can be considered. We also explain the differences of XSLT in comparison with XQuery, and thus we can determine which research directions apply in particular to XSLT because they are more useful for it.

Although the XSLT and XQuery languages have been developed with different aims, they have comparable expression power and similar applications. Both languages use XPath (W3C, 1999a; 2007a) as the path language for retrieving XML node sets, and have corresponding language constructs for iterations on an XML node set, definitions of variables, XML node constructors and definitions and calls of user-defined functions. Therefore, we can construct a translation from XSLT stylesheets into XQuery expressions and vice versa. A translation of an expression formulated in XSLT or XQuery into an expression formulated in the other language is correct for all input XML documents if, for each input, the results of applying the expressions are the same.

In the following, we describe application scenarios, where the applications of a translation from XQuery expressions into XSLT stylesheets and of a translation from XSLT stylesheets into XQuery expressions offer considerable benefits.

1. Translation from XQuery expressions into XSLT stylesheets: Using our proposed translation from XQuery expressions into XSLT stylesheets, the XQuery language can be used whenever commercially or freely available products support the processing of XSLT stylesheets, but do not support the XQuery language. Examples of such tools are BizTalk (Microsoft, 2007a), Cocoon (Apache Software Foundation, 2005a) and Xalan (Apache Software Foundation, 2005b).

2. Translation from XSLT stylesheets into XQuery expressions: Many commercially or freely available products support the evaluation of XQuery expressions, but do not support the XSLT language, for example Tamino XML Server (Software AG, 2007), Microsoft SQL Server 2005 Express (Microsoft, 2007b), Galax (Fernández et al., 2007) and Qizx (Axyana Software, 2007). A translation module from XSLT stylesheets into XQuery expressions can make the XSLT language available for these products. Another usage scenario is the migration of legacy systems. Legacy systems may use sub-systems which support XSLT. Further-