Chapter VI
Consistency and Modularity in Mediated Service-Based Data Integration Solutions

Yaoling Zhu
Dublin City University, Ireland

Claus Pahl
Dublin City University, Ireland

ABSTRACT

A major aim of the Web service platform is the integration of existing software and information systems. Data integration is a central aspect in this context. Traditional techniques for information and data transformation are, however, not sufficient to provide flexible and automatable data integration solutions for Web service-enabled information systems. The difficulties arise from a high degree of complexity in data structures in many applications and from the additional problem of heterogeneity of data representation in applications that often cross organisational boundaries. The authors present an integration technique that embeds a declarative data transformation technique based on semantic data models as a mediator service into a Web service-oriented information system architecture. Automation through consistency-oriented semantic data models and flexibility through modular declarative data transformations are the key enablers of the approach.

INTRODUCTION

A major aim of the Web service platform is the integration of existing software and information systems (Alonso et al., 2004). Information and data integration is a central aspect in this context. Traditional techniques based on XML for data representation and XSLT for transformations between XML documents are not sufficient to provide a flexible and automatable data integra-
Consistency and Modularity in Mediated Service-Based Data Integration Solutions

A data integration technique in the form of a mediator service can dynamically perform transformations based on a unified semantic data model built on top of individual data models in heterogeneous environments (Wiederhold, 1992). Abstraction has been used successfully to address flexibility problems in data processing (Rouvel-lou, Degenaro, Rasmus, Ehnebuske and McKee, 2000). With recent advances in abstract, declarative XML-based data query and transformation languages (Zhu et al., 2004) and Semantic Web and ontology technology (Daconta, Obrost and Smith, 2003), the respective results are ready to be utilised in the Web application context. The combination of declarative and semantic specification and automated support of architecture implementations provides the necessary flexibility and modularity to deal with complexity and consistency problems. Two central questions to the data integration problem and its automation shall be addressed in this investigation:

- How to construct data model transformation rules and how to express these rules in a formal, but also accessible and maintainable way is central.
- How integration can be facilitated through service composition to enable interoperability through connector and relationship modelling.

We show how ontology-based semantic data models and a specific declarative data query and transformation language called Xcerpt (Bry and Schaffert, 2002) and its execution environment can be combined in order to allow dynamic data transformation and integration. We focus on technical solutions to semantically enhance data modelling and adapt Xcerpt and its support environment so that it can facilitate the dynamic generation of Xcerpt query programs (in response to user requests) from abstract transformation rules.

BACKGROUND

Information integration is the problem of combining heterogeneous data residing at different sources in order to provide the user with a unified view (Lenzerini, 2002). This view is central in any attempt to adapt services and their underlying data sources to specific client and provider needs. One of the main tasks in information integration is to define the mappings between the individual data sources and the unified view of these sources and vice versa to enable this required adaptation. Fig.1 shows two sample schemas, which might represent the views of client and provider on a collection of customers, that require integration. The integration itself can be defined using transformation languages.