Chapter 16
Privacy-Conscious Data Mashup: Concepts, Challenges and Directions

Mahmoud Barhamgi
Claude Bernard Lyon 1 University, France

Chirine Ghedira
Claude Bernard Lyon 1 University, France

Salah-Eddine Tbahriti
Claude Bernard Lyon 1 University, France

Michael Mrissa
Claude Bernard Lyon 1 University, France

Djamal Benslimane
Claude Bernard Lyon 1 University, France

Brahim Medjahed
University of Michigan-Dearborn, USA

ABSTRACT

Modern enterprises across all spectra are increasingly adopting SOA-based data integration architectures to rapidly respond to transient data business needs. In this chapter, the authors analyze a new class of enterprise data integration application, called Data Mashup, in which data services are composed on the fly to answer new data business demands. The chapter reviews the different approaches to data mashup, discusses their limitations, and identifies the main requirements to data mashup. The authors next propose a declarative data mashup approach addressing the identified requirements. Finally, the chapter presents some research directions that must be followed in order for data mashup technology to mature.

1. INTRODUCTION

Since the dawn of the information age, data has been at the center of enterprise applications. The introduction of relational database management systems in the 1970’s created a productive new world that enabled developers of data-centric enterprise applications (i.e., data integration applications) to work much more efficiently than ever before. Data-centric application development in the pre-relational era meant hand-writing, tuning, and maintaining large procedural programs to access and manipulate the application’s data. Relational database systems made it possible for developers to write much simpler, declarative queries to accomplish the same tasks. Physical
system details such as indexes and clustering were hidden by the relational model, enabling developers to focus first on the logical tasks at hand. Developers of data-centric applications face a new and different barrier today. Relational databases have been so successful that there are many of them available (e.g., Oracle, DB2, SQL Server, and MySQL, to name a few of the most prominent ones). A typical enterprise is likely to have a number of relational databases within its corporate walls, and information about key business entities such as Customers or Employees is likely to reside in at least several of these systems. In addition, much of the information, even if it is stored relationally, will be relationally inaccessible because it is under tight application control. Access to application controlled information must come through the application APIs, as they enforce the rules and logic of the “business objects” of the application. As a result, enterprise data-integration application developers currently face a huge integration challenge: a given business entity of interest is now likely to reside in a mix of relational databases, packaged applications, and perhaps even in files or in legacy mainframe systems and/or applications. When a new, “composite” data-centric application needs to be created from these parts, we have to come back to procedural programming. This time, however, it is procedural programming against a wide variety of different subsystems, APIs, and data formats – essentially hand-coding what amounts to a distributed query or update plan. This situation has undermined the enterprise’s ability to respond to rapidly changing business requirements.

To face this new challenge, a recent trend has been to employ the SOA principle for data integration in modern enterprises (Spiess, Karnouskos, Savio, & Baecker, 2009; Janner, Siebeck, Schroth, & Hoyer, 2009). Data is provided to data-integration application developers as Data Services (a.k.a., DaasSs: Data-as-a-Service) that access and manipulate transparently data pieces (about the same business object) at the different data sources (Gilpin, Yuhanna, Leganza, Heffner, Hoppermann, & Smillie, 2007; Carey, 2006; Truong & Dustdar, 2009; Dan, Johnson, & Arsanjani, 2007). That is, data services are a new class of services that sit between the enterprise application developers and the enterprise’s heterogeneous data sources. They provide a well-documented, platform (and data source) independent, interoperable and uniform method of interacting with data. They shield the enterprise application developers from having to directly interact with the various data sources that give access to business objects (i.e., Customers, Orders, Invoices, etc.), thus enabling them to focus on the business logic only.

Similarly to traditional Web services (a.k.a., SaaSs Software-as-a-Service), different data services can be composed rapidly to respond to transient business data needs. This new class of data integration that relies on data services and the SOA principle was coined as Data Mashup (Jhingran, 2006; Bouguettaya, Nepal, Sherchan, Zhou, & Wang, 2010). Originally, the concept of Mashup (Bouguettaya, Nepal, Sherchan, Zhou, & Wang, 2010) has originated from the observation that the number of applications available on the Web is growing very rapidly, and so is the need to combine them to answer the business requirements. Mashups are also known as the Situational Applications; i.e. applications that come together “on the fly” for solving some immediate business problems. The Web site HousingMaps.com is an example of a site that “mashes-up” two other Web sites: Craigslist and Google Maps; it displays the available houses in an area by combining a listing from Craigslist with a display map from Google. The ProgramableWeb.com Web site lists more than /4800/ online mashups created by Web users.

Data Mashup (a.k.a., Information Mashup) is a special class of applications mashup that is concerned with combining information from several data sources to construct value added information for business needs. Access to data sources is often carried out through Web services
Related Content

Delivering Superior Customer Perceived Value in the Context of Network Effects
[www.igi-global.com/article/delivering-superior-customer-perceived-value/1874?camid=4v1a](www.igi-global.com/article/delivering-superior-customer-perceived-value/1874?camid=4v1a)

E-Commerce Business Models: Part 1
[www.igi-global.com/chapter/commerce-business-models/41196?camid=4v1a](www.igi-global.com/chapter/commerce-business-models/41196?camid=4v1a)

Electronic Risk Management
[www.igi-global.com/chapter/electronic-risk-management/9408?camid=4v1a](www.igi-global.com/chapter/electronic-risk-management/9408?camid=4v1a)

An Approach to Engineer Communities of Web Services: Concepts, Architecture, Operation, and Deployment
[www.igi-global.com/article/approach-engineer-communities-web-services/37434?camid=4v1a](www.igi-global.com/article/approach-engineer-communities-web-services/37434?camid=4v1a)