2008 IEEE International Conference on Web Services

Information as a Service in a Data Analytics Scenario – A Case Study

Vishal Dwivedi, Naveen Kulkarni
SETLabs, Infosys Technologies Ltd
{ Vishal_Dwivedi, Naveen_Kulkarni}@infosys.com

Abstract

In this work we present a case study of a SOA realization exercise at a business information provider firm, which deals with disparate sources of data in-order to provide reliable reports to its clients. Unlike typical enterprise scenarios, where applications are required to be service enabled, the key requirement here was to service enable its data acquisition, quality check, reporting and other processes which are either mostly manual or ETL based workflows. This paper also addresses how shared services, business processes, rules, and semantics are used to provide quality and agility to the internal processes many of which are entirely dependent on the type of data received. The case and the scenario are chosen specifically to emphasize the fact, that mere web-services implementation does not lead to service oriented architecture, but it is the appropriate usage of them.

1. Introduction

A leading US information provider Omega[†] has been a key provider of reliable and accurate financial information to their customers. Their key process entails capturing raw information across sources, its storage and validation and providing analytical reports to the clients. All this involves business processes which span across multiple processing hubs at different locations. Due to the high number of transactions spanning multiple geographic locations, Omega also requires a differentiated level of services in a secure environment.

Over time, the number of transactions that Omega processes has grown exponentially and the new scenario necessitates putting into place a flexible and reusable architecture based on service oriented principles. Though the basic business model of the client has had very few changes, what they see in terms of change is the ability to speed the process of acquiring data, providing accurate information and increase in data quality. The key requirement to introduce a single global platform for data acquisition and consumption and also ensuring minimal disruption while they move towards SOA adoption requires a certain level of challenges. In this work we present a brief description about their SOA adoption scenario and some of our experiences in the exercise.

2. Omega Business Context

The business information provided by Omega depends primarily on the data collected across different sources. Omega had developed and defined systems, processes, and validation procedures that ensure data received by its customer could readily be used for making decisions. In such processes quality of data is one of the most critical measures. At present, Omega has built solutions that comprises of ETL[‡] to perform the core operations. ETL jobs are defined to act upon the data feed from various sources and transform it such that it is ready for consumption by its customers. Apart from ETL, there have been few custom developed applications that offer capabilities like cleansing, verification, reporting, management etc. Figure1 presents one such sample process. However Omega is facing difficulties, with rigidly integrated components which provide very less flexibility for bringing in new partners (sources of data, external data validation agencies, probing agencies etc). It is estimated that after having identified a reliable source of data feed, it normally takes atleast three to four months to set up policies, rules, validation procedures, schemas, transformation and reports. Omega is also seeing a majority of their existing data sources have adopted newer technologies like XML and Web services to speed up the ability for Omega to pull the data. However, even with such newer technologies Omega is finding difficult to manage the ever increasing complexities.

Omega is looking at disruptive way of transforming their existing solution to offer an improved quality of data consistently and timely, reduce the cost, improve the efficiency and to manage the globalization effect. SOA has proven to be one such means to achieve flexibility and agility. Service orientation of its systems will provide abstractions over the core capabilities, common procedures, rules and management. For example, with globalization, service abstraction can provide a uniform

[†] Not the real name.

[‡] ETL here refers to the *Extract, Transform and Load* warehouse processes which range from extracting data from external resources, to their transformation and loading into the end target warehouses while maintaining the required quality standards.

978-0-7695-3310-0/08 $25.00 © 2008 IEEE
DOI 10.1109/ICWS.2008.119

view of data across different stores. Apart from adopting SOA, to achieve the desired effect, it is necessary to marry other technologies like Business Process Management (BPM), Business Rules Management (BRM) and Enterprise Service Bus (ESB).

The Figure1 represents the processes for acquiring data from various sources. This process could be modeled with BPM tools and could be triggered by a data event that arises out of an incoming feed (data is pushed to Omega), or a news in a market (data is pulled by Omega). Once triggered, the process is executed and various activities as described in the process are performed. In the traditional approach, all these activities were hardcoded as part of some custom application or within ETL. However, since the process is externalized it would be easy for adding or remove specialized activities as needed. Also with BRM, the rules can be externalized and centrally managed providing an opportunity for business to create and maintain rules. BPM, BRM and ESB all could be combined to facilitate optimized routing capability for the solution. This will enable improved efficiency and reduction in cost as some of the activities would be reused. As shown in Figure 1 is that activities within process also represent core ETL functions. With ETL tools today offering the possibilities of providing a service interface to manage their jobs and workflows, it would be imperative to expose them as services. These services are attached to the processes activities that are related to ETL. Hence, there would be a handoff from the process layer to ETL layer and the later taking care of actual extraction, transforming and loading.

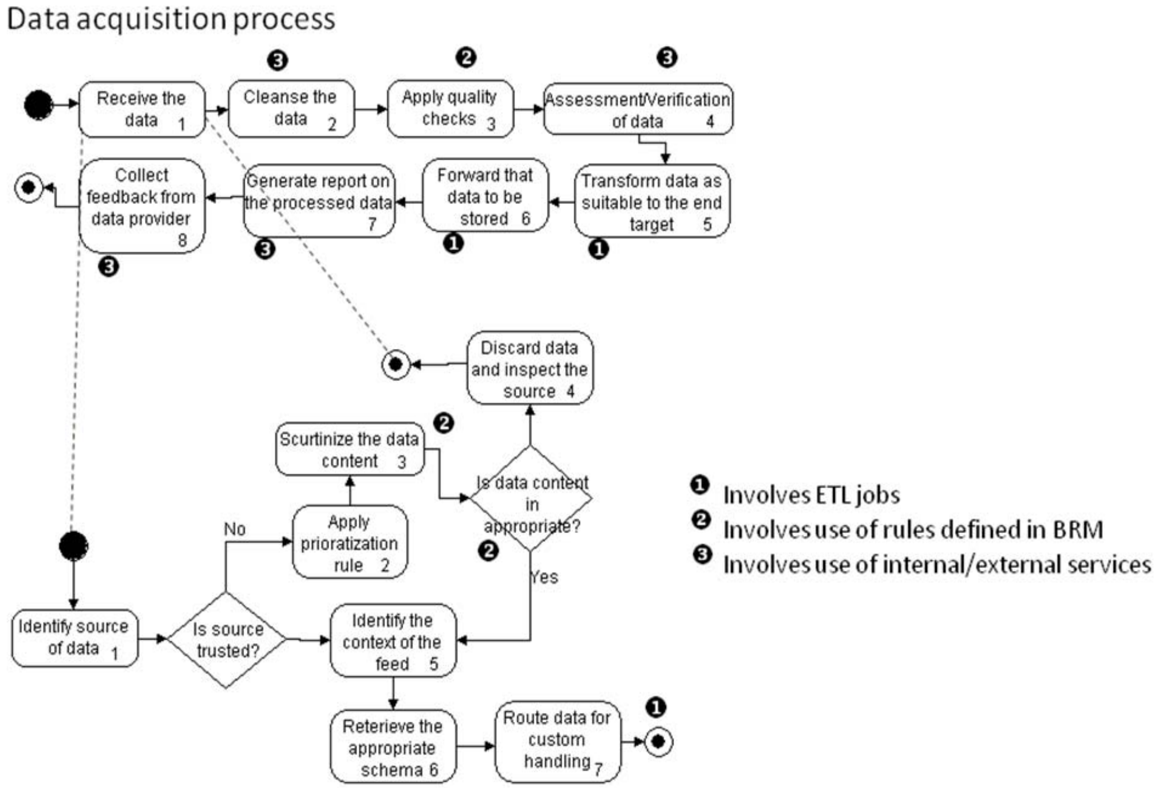
Data acquisition process
Receive the data 1
Cleanse the data 2
Apply quality checks 3
Assessment/Verification of data 4
Transform data as suitable to the end target 5
Forward that data to be stored 6
Generate report on the processed data 7
Collect feedback from data provider 8
Discard data and inspect the source 4
Scurtinize the data content 3
Is data content in appropriate?
Apply prioratization rule 2
No
Yes
Identify source of data 1
Is source trusted?
Identify the context of the feed 5
Reterieve the appropriate schema 6
Route data for custom handling 7
❶ Involves ETL jobs
❷ Involves use of rules defined in BRM
❸ Involves use of internal/external services

Figure1: An example cross section of Omega Process map**

3. Background Work

There has been quite a lot of work recently in academia and industry for SOA migration. At Infosys, we have our own architecture-centric framework, named Infosys Service Oriented Analysis/Adoption Process (InSOAP)[1]. InSOAP provides a systematic approach and a well-defined process to guide the design, evaluation and development of an Enterprise Architecture (EA) for SOA adoption scenario. One of the closest works to our approach is IBM's Service-Oriented Modeling and Architecture (SOMA) [2]. SOMA is a methodology for the identification, modeling and design of business aligned services at a proper level of granularity while leveraging existing systems.

** Illustrative

There exist other works like: Business Applications to Legacy Systems (BALES) methodology as proposed in [4, 5] and pattern based approached as proposed by Dabous in [8]. While BALES was based on Web-Services development using "objectified" legacy data collectively derived from both forward engineering (of the enterprise model) and reverse engineering approaches (to recover the legacy applications UML model), Dabous' approach was based on quantitative models to estimate the impact of service realization options. Even some of our work in [3] presents a decision framework for modernization which guides architects in the systematic ranking/selection of the most appropriate combination of service realization approaches for a given problem while taking into account the consequences on the desired quality attributes.

Although, there have been many SOA realization frameworks but their key focus has been on application migration to a service oriented architecture primarily focusing on business processes and rules. There has been some work on modeling Information as a service [6]. Scenarios such as one addressed in this paper present the need for such an approach. In particular for a domain like data analytics the processes typically have been human centric with very less work towards SOA adoption mainly due to challenges as per scale and real time concerns. Web-services in practice have not been good in scenarios where a lot of data is involved. Data analytics in general and Data as a service kind of scenarios necessitate not only appropriate service architecture but also a suitable service granularity. We address some of these issues in this case study and present a scenario which requires data as a service kind supply chain, involving millions of transactions most of which depend on the source and the type of data.

4. Preliminaries

4.1. Objectives of the exercise

The key objective of the exercise is to enable SOA migration of Omega applications in order to ensure agility of its enterprise architecture. Since Omega deals with a huge number of clients which traditionally have required customized processes, it requires a single platform to uniformly acquire data from sources across the world and processing of data, avoiding data redundancy and thus improving the accuracy of information provided to its customers.

Omega also intends to grow through acquisitions and in recent year has acquired quite a few companies whose solutions used different technologies. One of the objectives is also to quickly integrate the acquired data and processes and optimally use them. Data Analytics which has been the key operation of Omega requires ETL processes many of which are manual and require to be moved to BPM processes with integration of rules through standard BRMs.

4.2. Key Complexities

Unlike most enterprise SOA realization scenarios where applications require service enabling, Omega requires its data to be service enabled. Most of the operations in the exercise are ETL jobs which need to be well integrated with BPM and rule engines. The key complexities include:

Scale

- Business expects growth of at least 10 million records per week or more in acquiring data
- High number of cleansing checks/rules
- Consumers must be able to retrieve data with least number of refinement iterations in less than a couple of seconds

Data Diversity

- Data acquisition is done in diverse data formats in both batch and online modes from multiple sources

Redundancy

- High possibility of redundancy of information in current solution affecting the accuracy of data
- Multiple applications for the same business functions acting on different input type.

Quality of Data

- Quality management of data is critical as business adds different modes/sources of acquiring data
- Quality checks done through external partners as well as data providers through loop back processes

4.3. Guiding Principles for service enabling the data supply chain

Omega requires a meta-data based architecture where data semantics are captured along with the data chunks, irrespective of the source of the data. The processes in the ETL are required to be exposed to the BPM systems which provide an automated non-intrusive approach to access of all data. Whether data is pulled by Omega or it is pushed to its systems, it allows for “data on demand” paradigm by allowing services for access and semantic integration. Service would be modeled to handle volume, velocity and variety. This would help in automation by allowing for tracking of the data supply chain processes, by providing a composite view of data which may be required across several locations or may exist in chunks across several locations, thus avoiding redundancy. The ETL activities can thus have a shared data access, allowing content and context based routing across several engines based on rules which can scale the number of transactions processed.

5. Omega's approach

5.1. Service Architecture

As depicted in Figure 2, Omega implements a two tier architecture: first, the infrastructural services of granularity varying from system services to business services; second, the shared data services (SDS) to work for data coordination and aggregation across different data sources and location. The SDS stack is based on standards and is linked to ETL and they utilize the meta-data information of data elements.

5.2. Using an event based architecture

Omega has implemented event based architecture for triggering its internal processes. As shown in Figure 3 events can arise for different sources of data, feedbacks, when information is accessed, when data is scrutinized or during auditing. Data from a source could be in any form which when received by its data processing hubs usually triggers its processes. Although for most of the processing steps, the parameters of data transformation and data analysis have to individually defined for most of the organizations, standard rules.

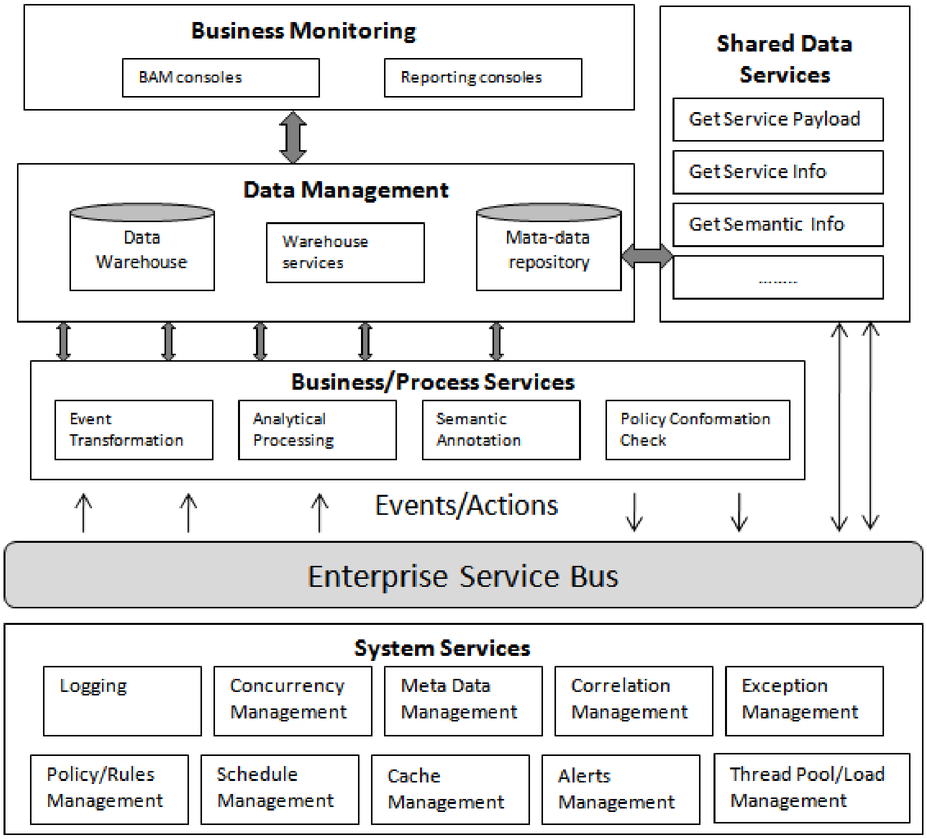

Figure2: Omega Service Architecture

This event based architecture is in charge of the following:
- Standard services like event transformations, semantic annotation, policy conformation etc
- Interface to external/internal systems for receiving data and triggering business transformations.
- Content and context based routing of the correct workflow specific to the data.
- Standardization of events for data analysis and transformations
- Defining and enacting exceptional behavior through BRM interfaces.

5.3. Coexistence of ETL and BPM

Omega, has built all its solutions around ETL workflows. These workflows automated the task of working on the data to make it more suitable for consumption, but are rigid as they need to be define per data type or source. Service architecture though provided the ability to manage the complexities described in Section 4.2, they still did not introduce the flexibility into the workflows. BPM hence is used along with the Service architecture for enabling the data supply chain processes such as the one described in Figure 1.

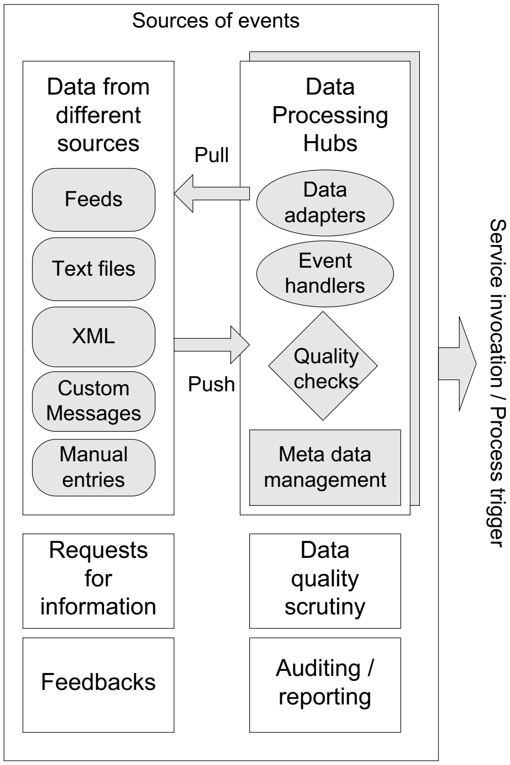

Figure 3: Sources of data and events for Omega

ETL currently represents an important investment for Omega and is central to their current solution. Further ETL tools available in the market are efficient and can handle variety of data types. They come with out-of-the-box features like highly sophisticated transformations, history and audit trails on positive and negative process runs, processing capabilities for large voluminous data etc. Further to this Omega's data sources provide data in a variety of types and format which the current ETL jobs are efficient in processing.

Though the current ETL jobs were rigid, Omega opted to model their data supply chain processes using BPM technologies. However, it was necessary that the existing ETL job were needed to be involved during the process execution. As shown in Figure 4, services interfaces are provided to ETL jobs. ETL tools today also offer different means of web service enabling the jobs like –

1. Consumer interface for ETL jobs to access information provided by other services, for example to validate a data item, ETL job might want to fetch data from warehouse
2. Provider interface for ETL jobs to provide massaged data, for example given an address an ETL job could cleanse it
3. Management interface for ETL job to provide management activities.

Processes that were modeled within BPM tools used the management interfaces exposed by ETL to initiate job. This enabled Omega to take advantage of processing capabilities of ETL and flexibility provided by BPM. While BPM and ETL shared the responsibility of all data supply chain process, any trigger to start the process or ETL job is handled by BPM layer and data feeds are handled by ETL. Further to introduce a uniform communication, ESB is used between ETL and BPM.

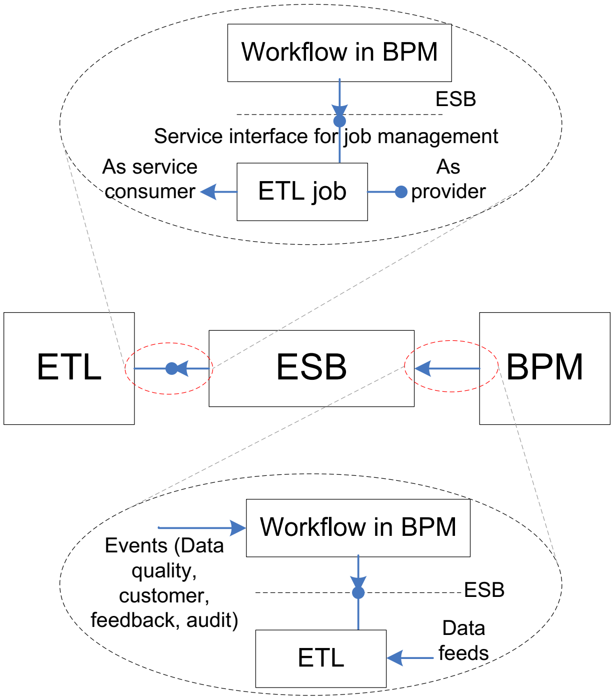

Figure 4: Process Management for Omega with service realization

5.4. Providing an Information as a service model

Based on the guiding principle of volume, velocity and variety as described in Section 4.3, Omega adopted to implement Information as a service (IaaS) model [6]. Though the key challenge for them was to define processes flexible enough to handle different data types, services defined using IaaS model complimented it as they are configurable to handle particular client information in the data supply chain processes. The processes thus utilize the routing provided by ESB and business rules, to trigger the associated workflow.

The key idea lies in their ability to decouple ETL activities, shared services and BPM/workflows providing for an automation of these jobs. This allows Omega to put in place workflows to handle complex events related to data quality, customer interaction, data management etc. The data quality process itself is automated to handle exceptions and track performance providing a centralized monitoring of business activities through BAM consoles.

Although, most of the concepts used at Omega are quite old, the key here is effectively using them in a data analytics scenario. This type of data analytics jobs are mostly human centric, but this approach has allowed Omega to move towards a high level of automation. They are even looking to put in place semantic information about data by integrating referential data based on ontology. This would allow for further automation and scale up their operations.

5.5. Real time semantics based processing

Omega captures its data from sources varying from text file dumps to automated RSS feeds. As shown in Figure 3, the data from different sources is received by data processing hubs, where it's processed. While most text data requires semi-automated processing, the structured data just requires a quality check approval. Data of all forms is uploaded to their respective repositories (through push/pull operations) which are polled after a fixed interval for contents. A case-id is assigned for each data type and a workflow instance is triggered for its processing.

Omega maintains meta-data information for most of its clients and the type of data they upload. Not only does this semantic information help in quick validation of the data, it also provides for its effective management by allowing for a shared data service implementation. Omega thus can manage with generic SDS implementations for its multiple clients.

Omega implements a real time OLAP over the client's data which is used for generating warehouse reports. A business activity monitoring console acts as a frontend for most of Omega's clients where they can access these reports and other details.

6. Conclusions

In this work we present a case study of an information provider firm – Omega, which seeks to automate its data supply chain processes to enable SOA. Unlike scenarios like legacy migration which involve building an appropriate service portfolio, Omega's business process encompasses activities varying from manual to system and ETL jobs each of which require a different approach for automation. This paper presents the key complexities involved in such a SOA realization scenario and presents the architecture for such a realization.

7. References

[1] Abdelkarim Erradi, Sriram Anand, Naveen N. Kulkarni: "*SOAF: An Architectural Framework for Service Definition and Realization",* IEEE SCC 2006: 151-158

[2] Ali Arsanjani, Abdul Allam, "Service-Oriented Modeling and Architecture for Realization of an SOA", IEEE SCC, 2006

[3] A. S. Erradi, A. and N. Kulkarni. "*Strategies for Integrating Legacy Applications as Services in a Service Oriented Architecture*", 12th Asia-Pacific Software Engineering Conference (APSEC), Taipei, Taiwan, 2005

[4] H. J. v. Heuvel, W.-J. v. d. and M. P. Papazoglou, "*A methodology to support web-services development using legacy systems*", IFIP Working Conference on Engineering Information Systems in the Internet Context, Kanazawa, Japan, pages 81–103, 2002

[5] W.-J. v. d. Heuvel. "*Matching and Adaptation: Core Techniques for MDA-(ADM)-driven Integration of new Business Applications with Wrapped Legacy Systems*", Model-Driven Evolution of Legacy Systems (MELS04), Monterey, Canada, 2004.

[6] Dan, Asit; Johnson, Robert; Arsanjani, Ali, "*Information as a Service: Modeling and Realization*", International Workshop on Systems Development in SOA Environments, 2007 (SDSOA), pp. 2-2.

[7] Ali Arsanjani, Service-oriented Modeling and Architecture: "*How to Identify, Specify and Realize Services for your SOA"*, Nov 2004, IBM Developerworks.

[8] F. T. Dabous. "*A Pattern Based Approach for the Architectural Design of e-Business Applications*", PhD thesis, School of IS, UNSW, Sydney, Australia, 2005