Semana de Engenharia 2010
Guimarães, 11 a 15 de Outubro

ELASTIC ENTERPRISE APPLICATIONS

Ana Nunes and José Pereira
Department of Informatics
University of Minho
E-mail: {ananunes,jop}@di.uminho.pt

KEYWORDS
SOA, Cloud Computing.

ABSTRACT

There is a growing awareness of scalability as a key property of enterprise applications.
The current target is that IT services are elastic, i.e. that they scale to very large dimensions but also that resources can be provisioned dynamically and incrementally. The motivation for this is twofold: First, to cope with applications with an increasing number of users in a single deployment. Second, to enable the same application to be deployed in increasingly larger settings, allowing a software provider to swiftly capture an emerging market.
We are interested in characterizing the major obstacles for migrating business applications to elastic settings, namely in database-centric applications, in order to propose practical architectural and middleware solutions to overcome them.

MOTIVATION

At the infrastructure level, the need for elasticity in enterprise application is being met with the cloud computing paradigm, the combination of a new business model with highly decentralized, scalable, and dependable systems. The infrastructures initially built to meet the internal requirements of large Internet applications such as Google or Amazon.com are currently being commercialized as collections of services that together realize the vision for elastic infrastructure.

Unfortunately, currently available cloud computing proposals fall short in face of the needs of mainstream business applications. Most offerings are for low level infrastructure services like virtual machines and raw storage, which leaves much of the scale issues to the developer. Even initial proposals for multi-tiered application platforms, such as Google App Engine and Windows Azure, are targeted at specific application scenarios and offer limited functionality. Moreover, there are additional dimensions to scalability such as management and maintenance of the application itself, in which an increasingly large number of interventions to keep the system operating and to fulfill changing business needs must be performed.

A particular cause for concern for small and medium businesses is the current reliance on traditional database management systems, by making use of advanced features or simply by implementing core business logic within the DBMS itself.
A major reason for this is the implementation of intricate business process management systems as collections of triggers and stored procedures, taking advantage of the robustness and performance of the transaction processing engine.

In sharp contrast, current proposals for data management in the cloud offer very limited functionality, rely on middleware layers for most processing and lack strong transactional guarantees.

Although migrating to a Service-Oriented Architecture is often cited as the long term strategy for scaling, it requires a profound refactoring of current systems, with a large investment in making explicit reliability and performance guarantees that currently are implicit in the usage of the transactional processing engine. These issues create a large gap between current mainstream business applications and the promises of elastic computing.

A multi-tier architecture differs from the traditional RDBMS-based architecture mainly by decoupling data management and business logic (Alonso et al, 2004). This decoupling enables the implementation of business logic in a programming paradigm of choice, independently of the RDBMS, while in turn, requiring the business logic to be made explicit.

Multi-tier architectures and the current middleware used to implement them are heavily geared towards scaling out: Capacity can be increased by adding servers at the business and presentation layers. This is possible due to these layers being mostly stateless and thus little to no coordination is required to ensure consistent operation.
In fact, middleware packages include transparent clustering capabilities out-of-the-box and distributed coordination is involved mostly in the maintenance of configuration (e.g. JNDI), not during application processing (Labourey and Burke 2004).
Scalability challenges arise at the bottom-most layer, responsible for data management and within the messaging framework.
At the data layer, scalability is addressed first by off-loading the database management system by using caching. An example of this is memcached

(http://code.google.com/p/memcached/), a distributed memory caching system for which bindings exist in several programming languages. This strategy is highly successful for many current web based applications, in which most of the workload is read-only. This is not the case of small and medium businesses for which the business logic is within the DBMS, as the workload will be mainly composed of updates.
Scalability of update operations is achieved by data sharding, a technique for horizontal (or row-based) data partitioning. Hibernate (http://www.hibernate.org) provides support for this approach. This must however be considered in the application logic, which has to address different shards of data at different servers. Moreover, it requires that the application lends itself to partitioning, which is true in large scale web based applications, but not necessarily in DBMS-based ones.

OBJECTIVES

This project aims at:

(a) characterizing the major obstacles to elastic cloud computing in common business applications, namely, those emerging from business logic currently embedded in database management systems and
(b) proposing architectural and middleware solutions to overcome them, with an emphasis on the practicality and cost effectiveness of their application.

In detail, this requires seeking answers to questions such as:

- What is a sufficiently representative application that can be used as a benchmark during the project?
- What concurrency and reliability semantics, currently implicit in the transactional mechanism, are in fact required?
- Which timeliness requirements are implicit in current solutions?
- How does best practice in implementing business processes within a centralized DBMS collide with elastic computing?
- Within the current discussion on the role and boundaries of a DBMS for cloud computing, what is the necessary functionality for the target application ?
- What is the relevant abstraction and interface presented by each of the layers in a cloud computing stack?
- How do current cloud computing offers fit the proposed architecture?
- How can the proposed abstractions be efficiently implemented at different scales?
- How does the proposed abstraction cater to dependability issues?
- Is there an incremental approach to migrating to the proposed architecture and middleware solutions?

This work takes on a systems approach through creating and testing prototypes, thus assessing the practicality of the research contributions in a representative real setting.

REFERENCES

Labourey, S. and Burke, B. 2004. “Jboss AS Clustering”
Alonso, G.; Casati, F.; Kuno, H. and Machiraju, V. 2004. “Web Services”. Springer New York.

AUTHOR BIOGRAPHIES

ANA NUNES was born in Vila Real, Portugal and went to the University of Minho, where she studied Informatics Engineering and obtained her degree in 2008. Later, she obtained her MSc degree in Informatics from the same university in 2009. She has since enrolled in the MAP-i Doctoral Program and is now pursuing a PhD in in the Distributed Systems area, under the specific theme of “Elastic Enterprise Applications”. Her e-mail address is : ananunes@di.uminho.pt .