Introduction

With a variety of services rapidly evolving at all architectural levels of cloud computing, there is an increasing demand for a standardized way to coordinate their interactions. Business process management, that is, more generally, the management of web-service–based workflows, could satisfy this demand and, indeed, first corresponding offerings have gained instant popularity. While from a functional perspective, these Platform-as-a-Service solutions are already quite mature, their support for fault tolerance is still very limited. Therefore they are inapplicable for critical tasks as regarded by TClouds.

Concerning the deficiencies of existing systems, we present a practical solution, called *Reliable BPEL Infrastructure* or simply *RBPEL,* for executing critical web-service–based workflows, particularly within clouds, in a fault-tolerant, highly available and highly configurable manner.

Our approach is practice-oriented, since current business process management infrastructures and workflows can be widely reused by transparently replicating them, and it is extensively configurable as well as dynamically adaptable through the use of external coordination services provided by today's cloud infrastructures. Figure 1 depicts the overall RBPEL architecture.

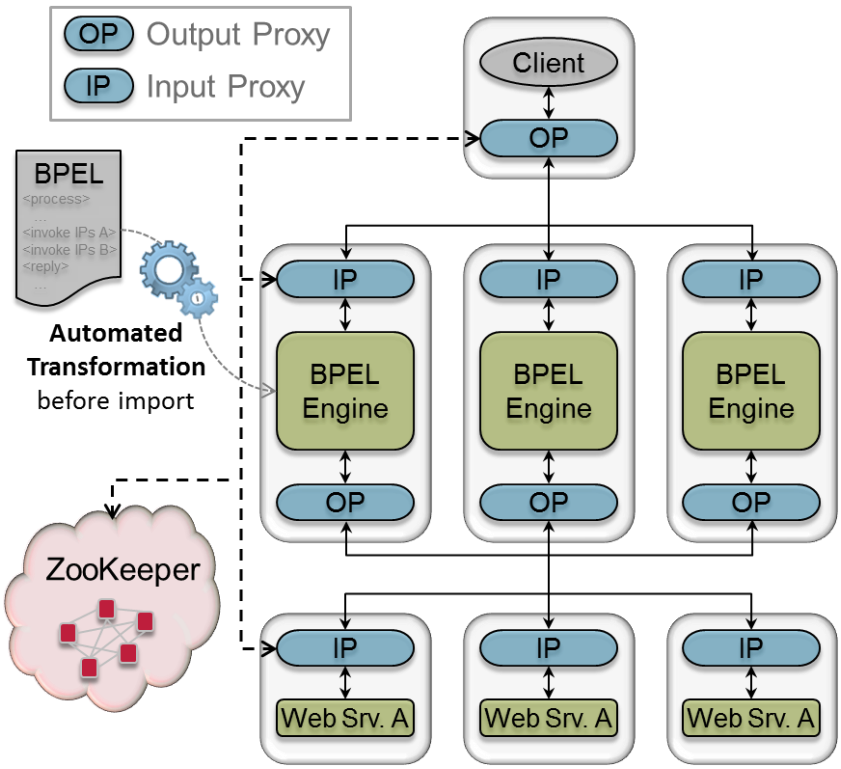

Figure: RBPEL Architecture

Standard BPEL Infrastructure

The *Web Services Business Process Execution Language* (WS-BPEL, short *BPEL*) [1] is a standardized XML-based language for describing business processes, or more generally, workflows that are based on one or more web services and that are offered as web services themselves. Thus, it would be a good candidate for providing an infrastructure for the execution of web-service–based workflows in the context of cloud computing.

However, standard BPEL engines, responsible for executing web-service–based workflows within BPEL infrastructures, log state changes to persistent storage to enable recovery of active workflows after a reboot or crash. This approach has two disadvantages: first, the need for synchronous logging slows down the execution speed during normal operation; second, the reliability of this mechanism depends on the reliability of the storage. In addition, BPEL provides only limited means to handle failures of the web services the workflows are based on. Making these web services fault tolerant is not supported at all by standard BPEL infrastructures.

This basically inhibits outsourcing of critical processes like public utility or medical services to clouds. Thus, a more reliable BPEL infrastructure is required which supports critical applications as regarded by TClouds.

TClouds — Trustworthy Clouds

RBPEL: RELIABLE WORKFLOW EXECUTION

www.tclouds-project.eu

RBPEL Architecture

We address the problems of standard BPEL infrastructures by actively replicating not only the BPEL engines, but (optionally) also the web services in a combined architecture. This architecture is designed according to three main objectives. First, all measures taken for fault tolerance have to be transparent to the workflows described in BPEL; that is, it should not make a difference whether a process definition is executed on a standard or a replicated infrastructure. Second, all measures are to be as little invasive as possible to existing BPEL infrastructures in order to reuse them. Third, for further minimizing the implementation effort, cloud services are to be used where possible.

In our architecture, these design objectives are met by means of proxies that intercept web-service calls to implement replication. In particular, the proxies distribute requests across replicas and collect results. Because web-service formats and protocols are used between clients and BPEL engines as well as between BPEL engines and web services, the replication of BPEL engines and web services can be achieved by almost the same mechanisms.

The proxies use an external coordination service, ZooKeeper [2], for general coordination, dynamic retrieval of system information and configuration, crash detection, and request ordering. Using an external service simplifies all these tasks and saves resources. It also can permit global coordination within clouds such as dynamically assigning replica virtual machines to servers and considering of quality-of-service constraints when choosing suitable web-service implementations.

To integrate a BPEL service with our architecture, the service needs to be made replication aware. This step is performed by a fully automated tool and does not require any manual modifications to the service code. In order to insert calls to the proxies handling replication, our tool transforms the BPEL process definition of the service to integrate. The transformed process definition is then loaded by the BPEL engine replicas and executed as any other BPEL process. Necessary adaptions of client requests are carried out by the proxies. Thereby, client applications don't need to be modified when invoking BPEL processes replicated on basis of RBPEL.

References

[1] http://www.oasis-open.org/committees/wsbpel/

[2] http://zookeeper.apache.org/

[3] Behl, J.; Distler, T.; Heisig, F.; Kapitza, R. & Schunter, M. Providing Fault-tolerant Execution of Web-service-based Workflows within Clouds. Proc. of the 2nd International Workshop on Cloud Computing Platforms (CloudCP '12), 2012.

Further Information

Further information about RBPEL can be found in [3] and under Deliverable „D2.2.1—Preliminary Architecture of Middleware for Adaptive Resilience".

Disclaimer

The TClouds project has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement number ICT-257243.

TClouds at a glance

Project number:
257243

TClouds mission:

- Develop an advanced cloud infrastructure that delivers computing and storage with a new level of security, privacy, and resilience.
- Change the perceptions of cloud computing by demonstrating the prototype infrastructure in socially significant application areas.

Project start:
01.10.2010

Project duration:
3 years

Total costs:
EUR 10.536.129

EC contribution:
EUR 7.500.000

Consortium:
14 partners from 7 different countries.

Project Coordinator:
Dr. Klaus-Michael Koch
coordination@tclouds-project.eu

Technical Leader:
Dr. Christian Cachin
cca@zurich.ibm.com

Project website:
www.tclouds-project.eu