Dynamic Re-configuration of Mobile Cloud Middleware based on Traffic Load

Huber Flores, Satish Narayana Srirama
Mobile Cloud Lab
Institute of Computer Science, University of Tartu
J. Liivi 2, Tartu, Estonia
{huber,srirama}@ut.ee

1 Project Goals

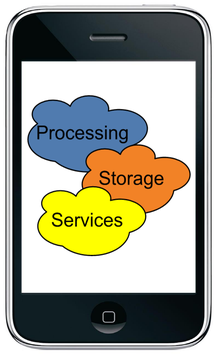
•To bring the cloud to the vicinity of a mobile

•To provide a reliable platform that handles the offloading and invocation of resource-intensive tasks from the smartphones

•To enrich the mobile applications with multi-cloud functionality

•To handle the oscillating telecommunication loads with dynamic re-configuration on the fly

• To increase the capabilities of our Mobile Cloud Middleware for managing concurrency (Erlang re-implementation)



2 Mobile Cloud Middleware (MCM)

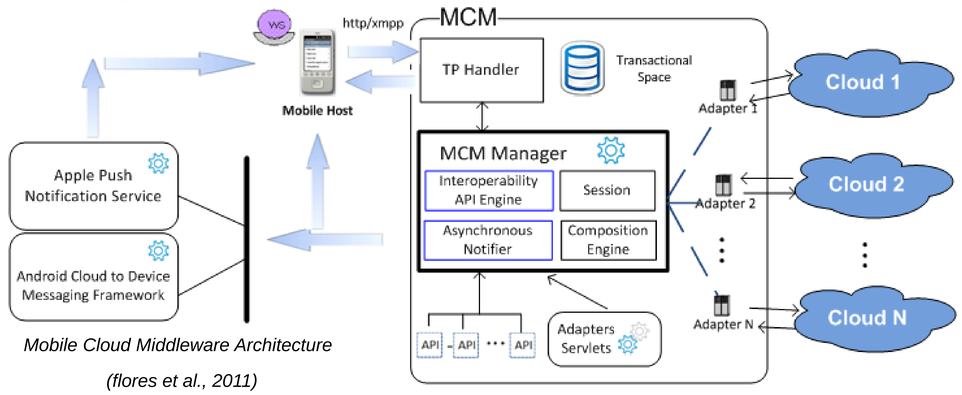
• Hides the complexity of dealing with multiple cloud vendors (abstraction of Web APIs from different cloud levels using Clojure)

•Provides hybrid cloud and cloud interoperability

•Fosters the de-coupling of the handset with a specific cloud vendor

•Manages asynchronous delegation of mobile tasks to the cloud resources by relying on push technologies.

•Enables to configure the deployment aspects of an offloaded task to the cloud



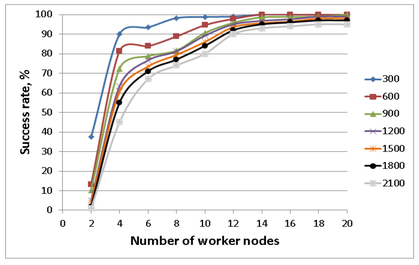
Mobile Cloud Middleware Architecture

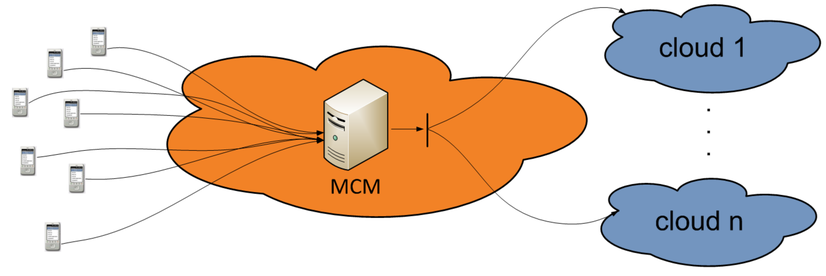
(flores et al., 2011)

3 MCM Migration to Erlang

While the prototype of MCM is working properly with the traditional Web technologies and can scale horizontally (adding more nodes rather than increasing the capabilities of a single node), as workload grows or shrinks, concurrent languages (e.g. Erlang) are preferable for cloud environments as the applications can take full advantage of the inherent distributed features of the cloud. For example, an application developed in Erlang and deployed on the cloud is highly fault-tolerant as it can be replicated by the cloud (in minutes) and easily be configured.

4 MCM Under Heavy Loads

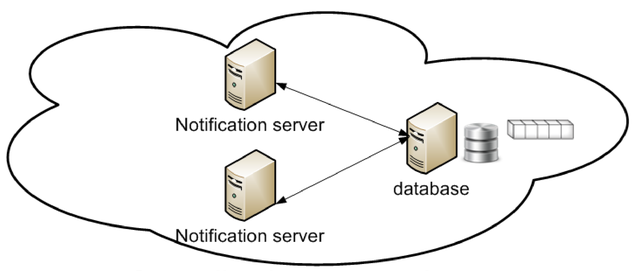




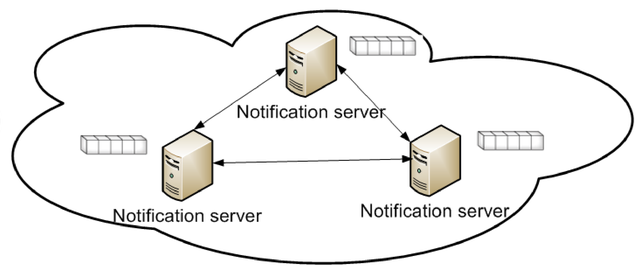
While MCM was successful in handling a multi-cloud operation from a mobile cloud application, a standalone framework, it faces the troubles of server utilization (e.g. CPU usage, memory usage etc.) and congestion of incoming mobile traffic, when managing heavy loads. In other words, each multi-cloud operation is an intensive task that overloads the server resources with long waiting network connections (cloud transaction), high data transfer and data access I/O rates

5 Re-configuration of MCM on demand

Cloud deployment is dependent on the type of services which are going to be provisioned. For example, a notification server running on a private cloud may work better, if the queue of the messages is distributed in memory among the notification cluster (if more than one server) rather than being centralized in a database which is accessed by the cluster concurrently. This kind of properties are only achieved by implementing the applications running on the cloud using the right technology (e.g. concurrent languages). In other words, different implementations behave differently and different approaches have to be applied for scaling the applications.



Centralized access to data



Distributed access to data

6 Acknowledgement

The research is supported by the European Social Fund through Mobilitas program, the Estonian Science Foundation grant ETF9287 and the European Social Fund for Doctoral Studies and Internationalisation Programme DoRa.