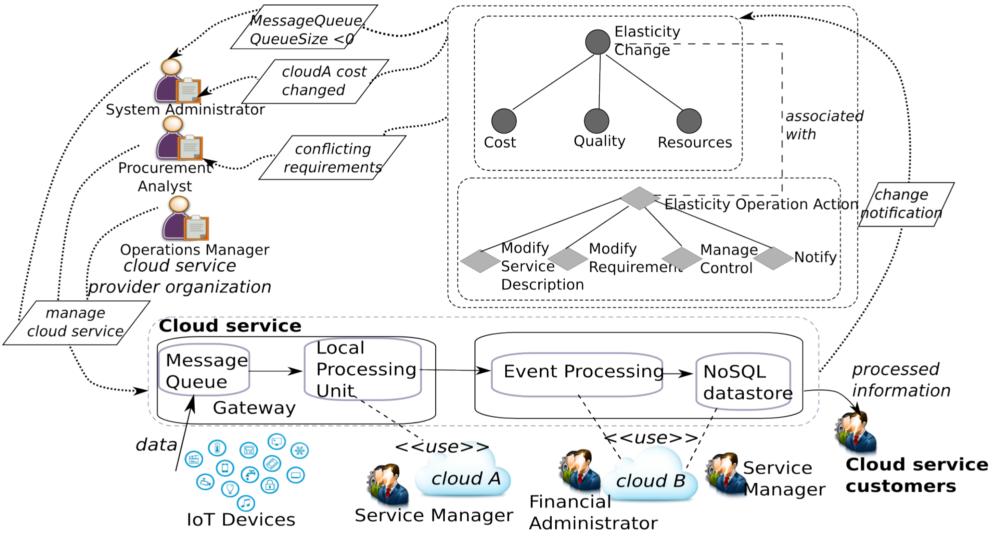
# #

Introducing eOMP



Overview

This part presents an overview of eOMP, the platform we propose for managing service elasticity operations.

Using eOMP

In this section we detail the necessary steps for using eOMP.

Extending eOMP

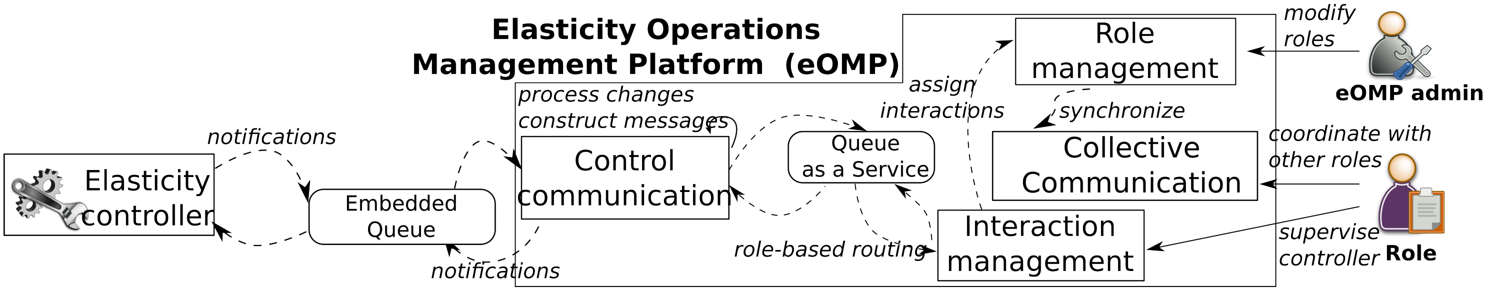
In case you want to add new features to eOMP, we explain provided hooks for easier extension of the platform.

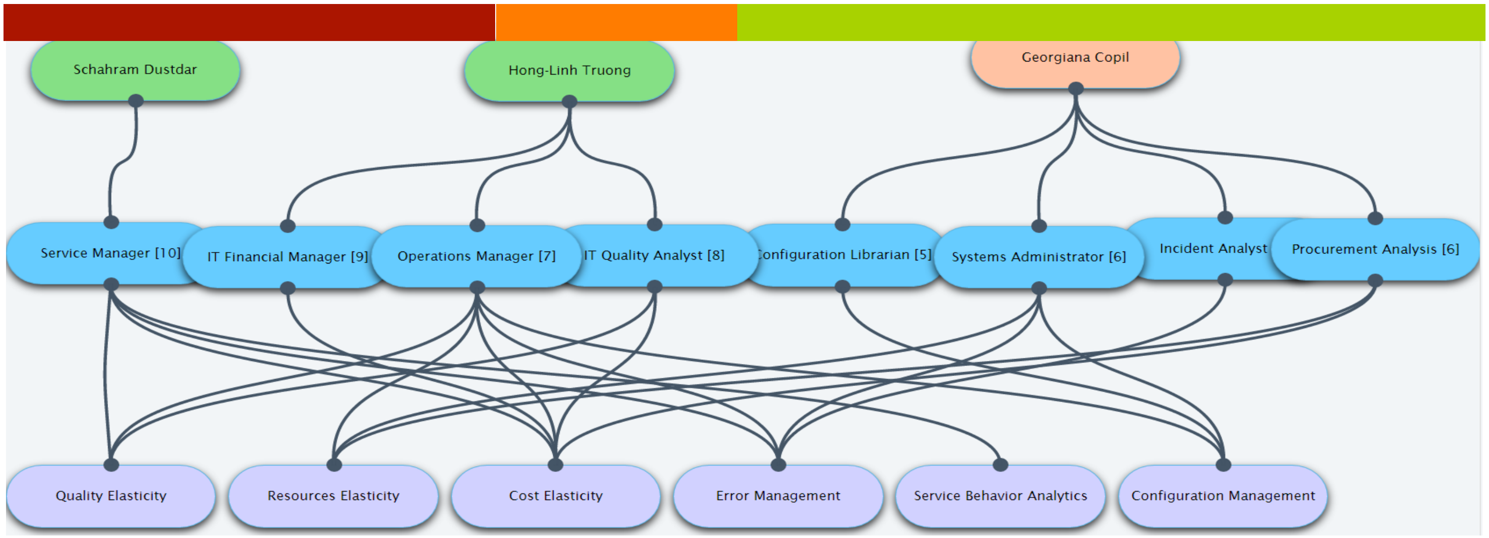
eOMP: supporting cloud service operation management for elasticity

Understanding service provider employees roles means better control for cloud services

Complex cloud services today rely on various IaaS, PaaS, and SaaS cloud offerings. Fast (re) deployment and testing cycles, and the rapidity of changes of various dependent infrastructures and services, imply a need for continuous adaptation. Although software-based elasticity control solutions can automate various decisions through intelligent decision-making processes, in many cases, such adaptation requires interactions among different cloud service provider employees and among different providers. However, decisions from stakeholders and elasticity software controllers should be seamlessly integrated.
We focus on interactions between service provider employees and elasticity controllers, and propose novel interaction protocols considering various organization roles and their concerns from the elasticity control point of view.

We introduce the **elasticity Operations Management Platform (eOMP)** which supports seamless interactions among service provider employees and software controllers. eOMP provides elasticity directives to enable notifications for complex elasticity issues solved by service provider employees, and the necessary mechanisms for managing cloud service elasticity. We show that, using eOMP, service provider employees can easily interact with elasticity controllers, and, according to their responsibilities, take part in the elasticity control to address issues that may arise at runtime for complex software services.



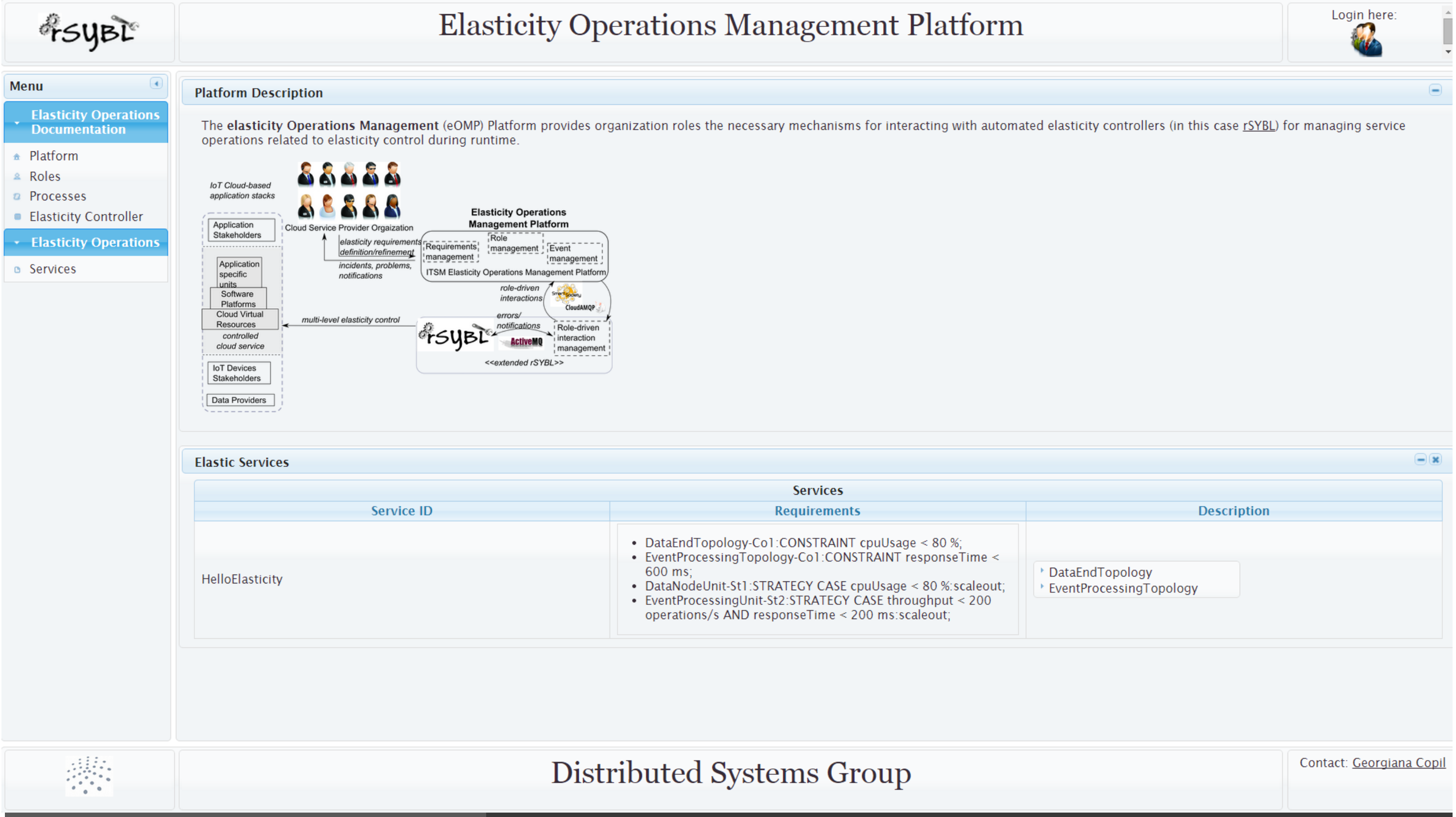


Overview

The elasticity Operations Management Platform (eOMP) is implemented as a Java enterprise application, which can be deployed either in the cloud or under service provider's premises. eOMP is open-source and available together with further experiments, details and user guide [1]. The current version integrates with the rSYBL elasticity controller, making use of the notification queue (i.e., embedded queue in the eOMP design, implemented using ActiveMQ[2]) exposing events during runtime. This can be easily extended to other elasticity controllers, by implementing an adapter for receiving and processing events. The service queue is using CloudAMQP[3], which is managed RabbitMQ[4] offered as a cloud service. Our Primefaces[5]-based frontend includes dynamically generating diagrams and charts for the cloud service provider employees.

The image above shows the roles present in our eOMP platform, with their associated responsibilities. This diagram can be accessed from the eOMP web interface, showing on the first line the service provider employees, on the second line their associated roles and on the third line the responsibilities that they have (e.g., quality elasticity management, resource elasticity, cost elasticity, error management). The current user in the snapshot above is Georgiana Copil, who has associated roles Configuration Librarian with authority 5, System Administrator with authority 6, Incident Analyst with authority 8, and Procurement Analyst with authority 6. With this view, the current employee can see his/her roles, and the roles of other employees, in order to understand the structure of the service provider organization, and the responsibilities associated with them.

A demo video of eOMP is available at https://www.youtube.com/watch?v=o-D3rXrNp4c



Elasticity Operations Management Platform

Login here:

Menu
Elasticity Operations Documentation
Platform
Roles
Processes
Elasticity Controller
Elasticity Operations
Services

Platform Description

The **elasticity Operations Management** (eOMP) Platform provides organization roles the necessary mechanisms for interacting with automated elasticity controllers (in this case rSYBL) for managing service operations related to elasticity control during runtime.

IoT Cloud-based application stacks
Application Stakeholders
Application specific units
Software Platforms
Cloud Virtual Resources
controlled cloud service
IoT Devices Stakeholders
Data Providers
Cloud Service Provider Orgaization
elasticity requirements definition/refinement
incidents, problems, notifications
Elasticity Operations Management Platform
Requirements management
Role management
Event management
ITSM Elasticity Operations Management Platform
role-driven interactions
CloudAMQP
multi-level elasticity control
errors/ notifications
rSYBL
ActiveMQ
Role-driven interaction management
<<extended rSYBL>>

Elastic Services

Services		
Service ID	**Requirements**	**Description**
HelloElasticity	• DataEndTopology-Co1:CONSTRAINT cpuUsage < 80 %; • EventProcessingTopology-Co1:CONSTRAINT responseTime < 600 ms; • DataNodeUnit-St1:STRATEGY CASE cpuUsage < 80 %:scaleout; • EventProcessingUnit-St2:STRATEGY CASE throughput < 200 operations/s AND responseTime < 200 ms:scaleout;	DataEndTopology EventProcessingTopology

Distributed Systems Group

Contact: Georgiana Copil

eOMP

Using eOMP

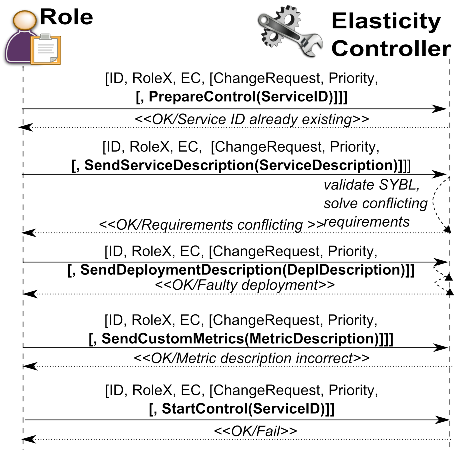
For using eOMP, some steps need to be followed firstly for starting the elasticity controller, and then for starting eOMP.

1. Download, configure and start rSYBL elasticity controller: http://tuwiendsg.github.io/rSYBL
2. Download eOMP https://github.com/tuwiendsg/rSYBL/tree/master/eOMP
3. Configure eOMP:
    a. **ControllerCommunication** - The controller communication interacts with rSYBL and with ElasticityOperationsManagementPlatform via 2 types of queues: (1) local activemq queue and (2) CloudAMQP service.
        i. rSYBL queue configuration
            1. QueueUrl =tcp://localhost:61616
            2. QueueName=events
            3. EventQueueEnabled=trueElasticityOperationsManagementPlatform
        ii. CloudAMQP configuration
            1. InteractionTopicName=eOMP
            2. CloudAMQPUri=

(Continued)

        3. CloudAMQPPassword=
        4. CloudAMQPVirtualHost=
        5. CloudAMQPUsername=
  - **b. ElasticityOperationsManagementPlatform**
    - **i.** Queue configuration - it interacts with ControllerCommunication via CloudAMQP queue – the configuration in a.ii should be the same as the one in b.i for ControllerCommunication and ElasticityOperationsManagementPlatform component to be able to communicate.
      1. InteractionTopicName=eOMP
      2. CloudAMQPUri=
      3. CloudAMQPPassword=
      4. CloudAMQPVirtualHost=
      5. CloudAMQPUsername=
    - ii. Database configuration
      1. For persistence we currently use H2 database at jdbc:h2:tcp://localhost/~/omp; - the connection url can be set from src/main/resources/META-INF/persistence.xml
      2. Create further users and roles before logging in eOMP to reflect your organization
      3. Further info on H2 can be found at http://www.h2database.com/ -

4. Use eOMP
   - a. Start the ControllerCommunication, and then start the ElasticityOperationsManagementPlatform (use Tomee for deploying the platform)
   - b. The first page displays the services which are currently controlled, their requirements, the capabilities of the controller and the capabilities of eOMP
   - c. After logging in, one can visualize the other employees and their responsibilities, as well as its own responsibilities.
   - d. Its own responsibilities details can be viewed by selecting the responsibilities item in the left menu.
   - e. By selecting interactions (of different types) the user can see interactions initiated towards him/her so far, and can respond to them by clicking the respond button.
   - f. A set of interactions semantically connected is called a *dialog.* By selecting a dialog, one can select from the context menu the “Dialog” option, which loads a diagram showing visually the dialog.
   - g. The high authority roles can also see statistics on how interactions have been ongoing. Since interactions are aggregated, and the aggregation level depends on the roles authority and its responsibilities, the higher authority roles, with higher-level responsibilities, will receive less or even no interactions while the lower level authority roles can receive hundreds of interactions.

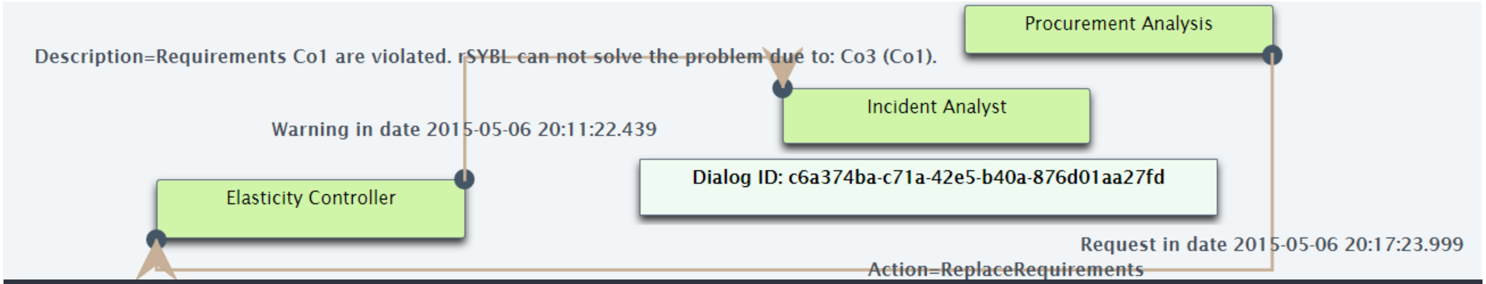

Role
Elasticity Controller
[ID, RoleX, EC, [ChangeRequest, Priority,
[, PrepareControl(ServiceID)]]]
<<OK/Service ID already existing>>
[ID, RoleX, EC, [ChangeRequest, Priority,
[, SendServiceDescription(ServiceDescription)]]]
validate SYBL, solve conflicting requirements
<<OK/Requirements conflicting >>
[ID, RoleX, EC, [ChangeRequest, Priority,
[, SendDeploymentDescription(DeplDescription)]]
<<OK/Faulty deployment>>
[ID, RoleX, EC, [ChangeRequest, Priority,
[, SendCustomMetrics(MetricDescription)]]]
<<OK/Metric description incorrect>>
[ID, RoleX, EC, [ChangeRequest, Priority,
[, StartControl(ServiceID)]]
<<OK/Fail>>

Extending eOMP

For extending eOMP in order to better map to the organization which wants to use it, the following options are available:

- In ControllerCommunication component:
  - Create a new client, for a different controller, by implementing the ControlClientInterface
  - Replace the cloudAMQP queue with a different queue, by replacing the CloudAMQPInteractions in InitiateInteractions class

- In ElasticityOperationsManagementPlatform
  - Replace the h2 database in persistence.xml with one better set for your needs
  - Split the jpa-ejb-user interface sub-modules if distribution is needed
  - Replace the cloudAMQP queue with a different queue, by replacing the CloudAMQPInteractions in



Description=Requirements Co1 are violated. rSYBL can not solve the problem due to: Co3 (Co1).

Warning in date 2015-05-06 20:11:22.439

Request in date 2015-05-06 20:17:23.999

Action=ReplaceRequirements

eOMP

For any kind of question, please contact Georgiana Copil:

- http://dsg.tuwien.ac.at/staff/ecopil
- e.copil@dsg.tuwien.ac.at

References

1. http://tuwiendsg.github.io/rSYBL/eOMP
2. http://activemq.apache.org
3. http://rabbitmq.com
4. http://cloudamqp.com
5. http://rabbitmq.com
6. http://www.primefaces.org

DISTRIBUTED SYSTEMS GROUP

Vienna University of Technology
Institute of Information Systems
184/1
Distributed Systems Group

Argentinierstrasse,8,
A-1040 Wien, Austria