WP3 - Marketplace Functions

Dissemination Level: Limited to Programme Participants

Lead Editor: Pablo Arozarena, TID

31/01/2013

Status: Final version

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 258862

Seventh Framework Programme
FP7-ICT-2009-5
Service and Software Architectures, Infrastructures and Engineering
This is a public deliverable that is provided to the community under a Creative Commons Attribution 3.0 Unported License: http://creativecommons.org/licenses/by/3.0/

You are free:

  to Share — to copy, distribute and transmit the work
  to Remix — to adapt the work

Under the following conditions:

  Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

With the understanding that:

  Waiver — Any of the above conditions can be waived if you get permission from the copyright holder.

  Public Domain — Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.

  Other Rights — In no way are any of the following rights affected by the license:

    Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;

    The author’s moral rights;

    Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.

  Notice — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this Web page.

For a full description of the license legal terms, please refer to:

http://creativecommons.org/licenses/by/3.0/legalcode
Contributors:

Michael Graumann, SAP
Frederic Junker, SAP/HSG
Andrea Giessmann, SAP/HSG
Anna Gatsioura, NTUA
Vrettos Moulos, NTUA
Pablo Arozarena, TID

Internal Reviewer(s):

Steve Strauch, University of Stuttgart
Johannes Wettinger, University of Stuttgart
Joerg Niemoeller, Ericsson

Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Authors</th>
<th>Sections Affected</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>31/01/2012</td>
<td>Pablo Arozarena</td>
<td>Version 1.0.</td>
</tr>
<tr>
<td>1.1</td>
<td>16/11/2012</td>
<td>Pablo Arozarena</td>
<td>Updated ToC</td>
</tr>
<tr>
<td>1.2</td>
<td>30/11/2012</td>
<td>Vrettos Moulos</td>
<td>Update on business resolver</td>
</tr>
<tr>
<td>1.3</td>
<td>05/12/2012</td>
<td>Pablo Arozarena</td>
<td>Updated User Guide chapter (ratings, comments, reports and bills).</td>
</tr>
<tr>
<td>1.4</td>
<td>13/12/2012</td>
<td>Michael Graumann, Frederic Junker, Andrea Giessmann</td>
<td>Updated on Business Model Simulator and Price Components</td>
</tr>
<tr>
<td>1.5</td>
<td>19/12/2012</td>
<td>Pablo Arozarena</td>
<td>Review version.</td>
</tr>
<tr>
<td>1.6</td>
<td>25/01/2013</td>
<td>Andrea Giessmann</td>
<td>Improved BMC screenshots.</td>
</tr>
<tr>
<td>2.0</td>
<td>31/01/2013</td>
<td>Pablo Arozarena</td>
<td>Final version.</td>
</tr>
</tbody>
</table>
# Table of Contents

Executive Summary .................................................................................................................. 10  

1. Introduction.......................................................................................................................... 11  
   1.1. Purpose and scope ........................................................................................................... 11  
   1.2. Document Overview ....................................................................................................... 11  

2. Marketplace prototype description ...................................................................................... 12  
   2.1. Virtual machines deployed ........................................................................................... 12  
   2.2. Relationship to other 4CaaSt Work Packages ................................................................. 14  

3. Installation and configuration ............................................................................................... 15  
   3.1. Market Place Frontend ..................................................................................................... 15  
      3.1.1. Configuration files ...................................................................................................... 16  
      3.1.2. Access to the 4CaaSt Marketplace frontend .............................................................. 17  
   3.2. Marketplace backend ...................................................................................................... 17  
      3.2.1. Configuration files ..................................................................................................... 18  
   3.3. Price Editor ..................................................................................................................... 18  
      3.3.1. Configuration file ...................................................................................................... 18  
      3.3.2. RESTful API .............................................................................................................. 18  
   3.4. Multidevice Reporting Tool ............................................................................................ 19  
   3.5. Business Model Component .......................................................................................... 20  
      3.5.1. Business Model Simulation ....................................................................................... 20  
      3.5.2. Price Model Simulation ............................................................................................. 20  
      3.5.3. Social Analytics ........................................................................................................ 20  
   3.6. Price Aggregator ............................................................................................................ 20  
   3.7. Business Resolver ......................................................................................................... 21  
   3.8. Business Charging Engine ............................................................................................. 21  
   3.9. Settlement Engine ......................................................................................................... 21  
   3.10. Marketplace Repository, Products Catalogue and Contracts Inventory .................. 22  

4. User guide ............................................................................................................................. 23  
   4.1. Marketplace Management ............................................................................................. 23  
   4.2. Publishing products in the Marketplace ......................................................................... 24  
   4.3. Contracting products in the Marketplace ....................................................................... 26  
   4.4. Price Editor .................................................................................................................... 28  
      4.4.1. Edit a price plan ......................................................................................................... 28  
      4.4.2. View a price plan ..................................................................................................... 29  
   4.5. Ratings and comments .................................................................................................... 30  
      4.5.1. Product ratings .......................................................................................................... 30  
      4.5.2. Product comments ................................................................................................... 31  
   4.6. Marketplace Reports .................................................................................................... 32
4.6.1. Predefined reports.................................................................32
4.6.2. User defined reports..........................................................33
4.7. Simulations ............................................................................36
  4.7.1. Business Model Simulation ..............................................36
  4.7.2. Price Model Simulation ....................................................45
4.8. View Bill ................................................................................52
5. Conclusions ............................................................................53
6. References ..............................................................................54
List of Figures

Figure 1. Marketplace physical architecture ................................................................. 14
Figure 2. Tomcat administration page ............................................................................. 16
Figure 3. Downloading the reporting app ....................................................................... 19
Figure 4. Adding a new organization ............................................................................. 23
Figure 5. Creating a new user ....................................................................................... 24
Figure 6. Selection of a blueprint template ..................................................................... 24
Figure 7. Selection of blueprint characteristics ............................................................. 25
Figure 8. Publication of a product in the Marketplace ...................................................... 25
Figure 9. Browsing the 4CaaSt catalogue ....................................................................... 26
Figure 10. Selection of product characteristics ............................................................. 26
Figure 11. Resolved Blueprint ...................................................................................... 27
Figure 12. Creation of price plan ................................................................................... 28
Figure 13. Adding a price component ............................................................................ 29
Figure 14. View of a price plan ..................................................................................... 30
Figure 15. Rate product ................................................................................................. 31
Figure 16. View ratings ................................................................................................. 31
Figure 17. Comment product ........................................................................................ 31
Figure 18. View comments ............................................................................................ 32
Figure 19. Access to reports ......................................................................................... 32
Figure 20. Contracted products report .......................................................................... 33
Figure 21. Ratings report ............................................................................................... 33
Figure 22. Customer reports ........................................................................................ 34
Figure 23. Selection of fields for a report ....................................................................... 34
Figure 24. Favourite reports ........................................................................................ 35
Figure 25. Bar chart .................................................................................................... 35
Figure 26. Pie chart ...................................................................................................... 36
Figure 27. Meta-model ................................................................................................. 37
Figure 28. Business model types .................................................................................. 38
Figure 29. Data Collection ............................................................................................. 38
Figure 30. Simulation types ............................................................................................ 39
Figure 31. Procedure model ........................................................................................ 39
Figure 32. Business Model Type Setup ........................................................................ 40
Figure 33. Defining attributes ...................................................................................... 41
Figure 34. Business Model Edition .............................................................................. 41
Figure 35. Data Acquisition ......................................................................................... 42
Figure 36. Add Business Model ................................................................................. 42
Figure 37. Edit Business Model .................................................................................... 43
Figure 38. Simulation ................................................................................................. 43
List of Tables

Table 1 4CaaSt Marketplace VMs ........................................................................13
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4CaaSt</td>
<td>Building the PaaS Cloud of the Future</td>
</tr>
<tr>
<td>BMC</td>
<td>Business Model Component</td>
</tr>
<tr>
<td>BMS</td>
<td>Business Model Simulation</td>
</tr>
<tr>
<td>CDR</td>
<td>Charging Data Record</td>
</tr>
<tr>
<td>FLEXISCALE</td>
<td>Public cloud platform provided by Flexiant Ltd.</td>
</tr>
<tr>
<td>HTML</td>
<td>HyperText Markup Language</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>JSON</td>
<td>JavaScript Object Notation</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>PaaS</td>
<td>Platform as a Service</td>
</tr>
<tr>
<td>PMS</td>
<td>Price Model Simulation</td>
</tr>
<tr>
<td>REST</td>
<td>Representational State Transfer</td>
</tr>
<tr>
<td>RESTful</td>
<td>Conforming to REST constraints</td>
</tr>
<tr>
<td>SSH</td>
<td>Secure Shell</td>
</tr>
<tr>
<td>URL</td>
<td>Unified Resource Locator</td>
</tr>
<tr>
<td>VM</td>
<td>Virtual Machine</td>
</tr>
<tr>
<td>WAR</td>
<td>Web application ARchive</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
</tbody>
</table>
Executive Summary

The 4CaaSt Marketplace offers the capability to trade the applications and services provided by 4CaaSt [1]. As such it is intended as a sort of front end between 4CaaSt users and the system itself. Three types of Marketplace users are considered: Marketplace administrators, service providers and service consumers.

The second release of the 4CaaSt Marketplace has been prototyped, building on the first release [2], and deployed on a cloud testbed provided by Flexiant (FLEXISCALE\(^1\)). The different components that make up the Marketplace are deployed on a set of virtual machines supported by this testbed.

This prototype supports the publication, selection, customization and contracting of cloud services. It also provides simulation tools which allow service providers experiment with different business models and pricing strategies. For those purposes, 4CaaSt Marketplace features a Web based user interface which offers all the necessary operations. Besides, a set of backend components implement the business logic of the prototype while a persistence layer stores relevant information in a database.

Actual service deployment, execution and monitoring rely on the capabilities provided by other 4CaaSt work packages, in particular WP2, WP4 and WP5. For more information, please, refer to deliverables [3], [4] and [5].

In this document, a set of guidelines on how to set up and configure the 4CaaSt Marketplace are given. In addition, a short user guide covering the main capabilities of its user interface is included. This information, together with the corresponding software release, should suffice to install and use the prototype.

Note this document only refers to the functionality implemented in the scope of the two first 4CaaSt iterations. When the third and last iteration of this prototype becomes available the document will be updated accordingly (project month 39).

\(^1\) http://www.flexiscale.com/
1. Introduction

1.1. Purpose and scope

This document complements the software release of the 4CaaSt Marketplace prototype by providing useful installation and use guidelines. Thanks to them, 4CaaSt Marketplace users will be able to properly install the software components needed to realise the second iteration prototype. In addition, the main capabilities provided by the 4CaaSt Marketplace to its users are described by means of relevant screen captures and textual descriptions.

For a detailed description of the requirements and architecture of this prototype, please, refer to [6], [7] and [8].

1.2. Document Overview

The document is structured as follows:

- Chapter 1 gives an introduction to the document.
- Chapter 2 provides a high level description of the 4CaaSt Marketplace prototype.
- Chapter 3 goes on to describe how the different components that make up the prototype can be installed and configured.
- In chapter 4 guidelines on how to operate the 4CaaSt Marketplace, by means of its user interfaces, are provided.
- Finally, Chapter 5 gives some conclusions about the results obtained in the second iteration of the 4CaaSt Marketplace.
2. Marketplace prototype description

The 4CaaSt Marketplace prototype is implemented as a set of software components that are deployed on a cloud infrastructure provided by FLEXISCALE. Several virtual machines are created in this testbed to host the Marketplace prototype.

Users of the Marketplace have access to its user interface by means of a Web-based front end deployed on a Tomcat application server, while persistency is granted by a MySQL database. Communications between different components are mainly based on REST interfaces and Web services.

A detailed description of the functionality addressed by the second release of the 4CaaSt Marketplace can be found in deliverables [6] and [7].

2.1. Virtual machines deployed

The 4CaaSt Marketplace prototype is deployed on a set of virtual machines provided by the FLEXISCALE testbed. Although all the Marketplace components could have been deployed on a single virtual machine, it has been decided to take advantage of the flexibility and scalability provided by the cloud infrastructure available in the testbed by using several virtual machines instead.

Even though different testbed configurations are possible, this document focuses on describing the actual set-up used to validate the Marketplace prototype. In the following paragraphs each of the virtual machines used are described.

- **Virtual Machine 1: Marketplace repository**
  - IP address: 109.231.82.98
  - Operating System: Ubuntu 10.04 32-bit
  - Database: MySQL 14.14 Distrib 5.1.41
    - Port: 3306
    - user/password: root/root
    - Schema: 4CAAST
    - /etc/mysql/my.cnf (configuration file)
    - Basedir = /usr/share/mysql
    - Datadir = /var/lib/mysql
  - 4CaaSt Components: Marketplace Repository

- **Virtual Machine 2: Marketplace front-end and back-end**
  - IP address: 109.231.82.99
  - Operating System: Ubuntu 10.04 32-bit
  - Application Server: Tomcat 6.0.32
    - /home/ubuntu/apache-tomcat-6.0.32/webapps
    - http://109.231.82.99:8080
    - admin password: 12345678
  - 4CaaSt Components: Marketplace front-end and back-end

- **Virtual Machine 3: Business resolver**
  - IP address: 109.231.77.106
  - Operating System: Customized Version Based on Debian (Testing Edition) and Linux Mint Debian Edition
• Application Server: Tomcat 7.0.29
• 4CaaSt Components: Business resolver
• Credentials:
  ▪ user/password: root/4caast

• Virtual Machine 4: Price components and simulations
  o IP address: 109.231.77.108
  o Operating System: Windows Server 2008 R2 Standard (SP 1)
  o Application Server: Tomcat 6.0.35
  o 4CaaSt Components:
    ▪ Business Model Component
    ▪ Charging Component
    ▪ Price Editor
    ▪ Price Model Simulator
    ▪ Price Aggregator

Table 1 summarizes the information related to 4CaaSt Marketplace VMs, which is also depicted as a diagram in Figure 1.

<table>
<thead>
<tr>
<th>VM #</th>
<th>OS</th>
<th>Applications</th>
<th>4CaaSt Components</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM1</td>
<td>Ubuntu</td>
<td>MySQL</td>
<td>• Marketplace Repository</td>
<td>109.231.82.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Products Catalogue</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Contracts Inventory</td>
<td></td>
</tr>
<tr>
<td>VM2</td>
<td>Ubuntu</td>
<td>Tomcat</td>
<td>• Marketplace front-end</td>
<td>109.231.82.99</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Marketplace back-end</td>
<td></td>
</tr>
<tr>
<td>VM3</td>
<td>Debian Testing</td>
<td>Tomcat</td>
<td>• Business Resolver</td>
<td>109.231.77.106</td>
</tr>
<tr>
<td>VM4</td>
<td>Windows Server 2008</td>
<td>Tomcat</td>
<td>• Business Model Component</td>
<td>109.231.77.108</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Charging Component</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Price Editor</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Price Model Simulator</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Price Aggregator</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 4CaaSt Marketplace VMs
2.2. Relationship to other 4CaaS Work Packages

The 4CaaS Marketplace relies on the functionality provided by other work packages. The main interactions between the second iteration Marketplace prototype and other 4CaaS work packages are as follows:

- **BluePrint access**: WP2 provides the list of blueprints resolved from a technical perspective.
- **Deploy Service**: once the service is contracted, WP4 is in charge of actually deploying it on top of the cloud infrastructure provided by the FLEXIANT testbed.
- **Monitoring and accounting**: in order to rate and charge services the 4CaaS needs accounting data related to the contracted services, which is provided by WP5.
- **WP6 and WP7**: although there is not a direct relationship between WP3 and WP6 and WP7 in terms of software interfaces, all services and building blocks provided by WP6 and WP7 are offered to 4CaaS users via the WP3 Marketplace.
- **WP8**: the three use case prototypes developed in WP8 also make use of the Marketplace for the commercialization of the services implemented. To do so, relevant products need to be published in the Marketplace for each scenario, together with corresponding price models and additional business attributes.
3. **Installation and configuration**

In the following paragraphs the installation and configuration of the different components that make up the second release of the 4CaaSt Marketplace prototype is described (see [8] for a detailed description of these components).

3.1. **Market Place Frontend**

This component groups the functionalities referred to as:

- Product Definition
- User Profile Management
- Product Contracting
- Search Product
- View Product Summary
- View Contracts/ Contract Detail
- Ratings and comments

In addition, the Business Model Component (3.5) can be launched from the Marketplace frontend.

The Marketplace frontend is implemented as a Web Application deployed on a Tomcat application server. It is deployed as a virtual machine on the FLEXISCALE testbed with the following IP address: 109.231.82.99.

To log in via SSH use user/password: ubuntu/pnZ5FrGVe28h0YUe.

Tomcat must be installed in the server where the frontend is to be deployed. The path to Tomcat is: /home/ubuntu/apache-tomcat-6.0.32/bin

To start and stop Tomcat the following commands can be used:

- ./startup.sh
- ./shutdown.sh

The name of the WAR file which implements the front end is: 4CAAST_Marketplace_V4.war.

This component can be downloaded from:

- https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/FrontEnd/lib

In order to deploy it, the Tomcat administration interface (Figure 2) has to be used. Access data are as follows:

- URL: http://109.231.82.99:8080/manager/html
- user/password: admin/12345678.

The location of the WAR file after deployment is:

- /home/ubuntu/apache-tomcat-6.0.32/webapps
3.1.1. Configuration files

The 4CaaSt Marketplace uses two configuration files that are embedded in the 4CaaST_Marketplace_V4.war file. In case changes are needed, they can be accessed in the following directory:

/home/ubuntu/apache-tomcat-6.0.32/webapps/4CaaST_Marketplace/WEB-INF/classes

In the following sections details about the contents of these files are provided.

3.1.1.1. File jdbc.properties

It contains information related to the data base (MySQL) used as Marketplace repository. The parameters that need to be configured and appropriate values are as follows:

- BBDD_PATH = jdbc:mysql://109.231.82.98:3306/4CAAST_DEMO
- BBDD_USER = root
- BBDD_PSW = root
- BBDD_SCHEMA = 4CAAST_DEMO
- BBDD_DRIVER = com.mysql.jdbc.Driver
- BBDD_TYPE = mysql
3.1.1.2. **File app.properties**

It contains information related to access to other applications as the price viewer and service deployment. If they are changed Tomcat has to be restarted. The parameters that need to be configured and appropriate values are as follows:

- **deployment.url=**  
  http://109.231.87.130:8080/com.sap.deploymentmanager.frontend/rest/deployment

- **resolver.url=**  

- **priceviewer.url=**  

- **priceviewerWeb.url=**  
  http://109.231.77.108:8080/PriceEditor/VAADIN

- **getBlueprint.url=**  
  http://blueprinthq.host-for.me/4CaaSt/blueprint_hq.php

3.1.2. **Access to the 4CaaSt Marketplace frontend**

In order to access the 4CaaSt Marketplace frontend installed in FLEXISCALE, with a regular web browser, use the following data:

- **URL:**
  - http://109.231.82.99:8080/4CAAST_Marketplace_V4

- **Users:**
  - Marketplace management:
    - user/password: admin/admin
  - Service Provider:
    - user/password: sptel/sptel
  - Customer:
    - user/password: customer/customer

3.2. **Marketplace backend**

This component groups the functionalities referred to as:

- Blueprints Access
- Contracts Manager
- Delivery Manager

The Marketplace backend is implemented as the server part of a Web Application deployed on a Tomcat application server. It is deployed on a virtual machine on the FLEXISCALE testbed with the following IP address: 109.231.82.99.

Basically the Marketplace backend acts as an intermediary between the frontend and the components in charge of handling blueprints and the actual deployment of services in the 4CaaSt cloud. Communication with these components, which are not part of the Marketplace, is conducted via Web services.

Details on how to use Tomcat to deploy the Marketplace can be found in section 3.1. This component can be downloaded from:

- https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/FrontEnd/lib
The name of the WAR file which implements the back-end is: 4CAAST_Marketplace_V4.war and, after deployment, is located at:

- /home/ubuntu/apache-tomcat-6.0.32/webapps

### 3.2.1. Configuration files

Refer to section 3.1.1.

### 3.3. Price Editor

This component groups the functionalities referred to as:

- Price Editor
- Price Viewer
- RESTful API for the Price Editor / Viewer

The Price Editor is implemented as a Vaadin Web Application deployed on a Tomcat application server. It is deployed on a virtual machine on the FLEXISCALE testbed with the following IP address: 109.231.77.108/109.231.82.99. It can be accessed at the following URL:


Basically the Price Editor acts as a form editor to view and manipulate price models associated to products. Communication with these components, which are not part of the user interaction, is conducted via RESTful Web services.

The name of the WAR file which implements the front end is: PriceEditor.war. This component can be downloaded from:

- [https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/PriceEditor/deploy](https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/PriceEditor/deploy)

In order to deploy it, the Tomcat administration interface has to be used (see section 3.1).

#### 3.3.1. Configuration file

The Price Editor uses one configuration file that is embedded in the PriceEditor.war file. In case changes are needed, they can be accessed in the following directory:

/WEB-INF/classes/org/fourcaast/priceeditor

It contains information related to the data base (MySQL) used as Marketplace repository. The parameters that need to be configured and appropriate values are as follows:

- **DB_PATH** = jdbc:mysql://109.231.82.98:3306/4CAAST_DEMO
- **DB_USER** = root
- **DB_PSW** = root
- **DB_DRIVER** = com.mysql.jdbc.Driver

#### 3.3.2. RESTful API

The API of the Price Editor can be accessed via the URL mentioned in 3.3 by adding the following parameters as query string:

- **product**: ID of the product for which the price model will be displayed
- **bp_repository**: specifies the repository of the blueprints and can take the following values:
  - **internal**
  - **localfile**
- `localfile_serialized`
- `wp2`

- **editable:**
  - `true`: launch Price Editor (default)
  - `false`: launch Price Viewer (no editing possible)

Example:


### 3.4. Multidevice Reporting Tool

This component provides Reports to service providers in two ways:

- As an integrated tool in the Marketplace.
- As an Android app installed on a mobile Device (tablet, mobile phone)

This tool is composed of two different parts:

- A client side which can be used as an integrated tool in the Marketplace or as an Android App installed on a mobile device:
  - Marketplace frontend
    - The functionality is integrated in the Marketplace frontend, so no additional installation is needed (see section 3.1).
  - App (tablet or smartphone):
    - Logged as Service Provider, the Android App can be downloaded from the 4CaaSt Marketplace’s My Mobile App tab (Figure 3).

![Marketplace](image)

**Figure 3. Downloading the reporting app**

- A server side, implemented as a back-end rest service that sends data in JSON format to the client side of the tool.

  The Server tool is packaged in a war file and is deployed on Apache Tomcat 6.0.32 with JVM Version 1.6.0_26-b03 as a REST API. It can be downloaded from:

  - `https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/FrontEnd/lib/rest-json-xml-consumer.war`
3.5. **Business Model Component**

The Business Model Component comes as a WAR archive file containing all necessary libraries. It is deployed on a virtual machine running Tomcat as application server and can be accessed at:

http://109.231.77.108/BMC

The BMC is a wrapper holding the Business Model Simulation, Price Model Simulation as well as Social Analytics. These components can be accessed directly by adding a parameter called tab to the URL mentioned above that can take the following values:

- 0: Business Model Simulation
- 1: Price Model Simulation
- 2: Social Analytics

Example: http://109.231.77.108/BMC?tab=0

3.5.1. **Business Model Simulation**

The Business Model Simulation can be accessed at

http://109.231.77.108/BMC?tab=0

3.5.2. **Price Model Simulation**

The Price Model Simulation can be accessed with either the (1) blueprint-id as parameter or the (2) product-id:


It is deployed as a single Web Application on a Tomcat application server being embedded in the Business Model Component.

3.5.3. **Social Analytics**

Social Analytics can be accessed at:


3.6. **Price Aggregator**

The Price Aggregator comes as a WAR archive file containing all necessary libraries. PriceAggregator.war is deployed e.g. to an Apache Tomcat server like any other WAR file. It is deployed on a virtual machine provided by FLEXISCALE with the following IP address: 109.231.77.108. It can only be accessed via POST-Requests at the following URL:

http://109.231.77.108/PriceAggregator/PriceAggregator

The POST-Request must contain the following parameters:

- **method**: priceaggregator
- **blueprint**: ARB (.xml - file) of the product whose Price Model Components will be aggregated

The response consists of the aggregated Price Model as .xml-File.

The Price Aggregator component can be downloaded from:

https://svn.forge.morfeo-project.org/4caast/tags/Release_1.0/WP3/PriceAggregator/deploy/
3.7. Business Resolver

In this section, the Business Resolver that makes up the second release of the 4CaaSt Marketplace prototype is described.

The Business Resolver component has two different parts:

1) The Business Resolution Engine which is the key component for the resolution
2) The Business Interface Framework that is the REST API that implements the interfaces and transactions between the components.

This component can be downloaded from:

- https://svn.morfeo-project.org/4caast/trunk/WP3/BusinessResolutionEngine/archive/PlatformBusinessResolver_DEMO_2Year.zip

The Resolution Server is implemented as a Web Application deployed on a Tomcat application server (version 6). It is deployed on a virtual machine provided by FLEXISCALE with the following IP address: 109.231.77.106.

The whole procedure that takes place in order to start the server is fully automated due to maven. So the coyote, the tomcat, the java and the dependencies' libraries are automatically downloaded from the repository. All the packages are described into the pom file (pom.xml).

The tomcat server is installed into /root/tomcat7 directory.

In order to make it easier to maintain it, one script is implemented. In order to run and deploy the server the build.sh script should be executed.

The credentials are into the property file that provides the username and the password for the server as well as the location of the files.

The server has a modified version of Debian Testing Edition. Also few packages are from Linux Mint Debian Edition Repositories.

3.8. Business Charging Engine

The Business Charging Engine comes as a WAR archive file containing all necessary libraries. ChargingComponent.war is deployed e.g. to an Apache Tomcat server like any other WAR file. It is deployed on a virtual machine provided by FLEXISCALE with the following IP address: 109.231.77.108. It is accessed via web browser / GET-Request to the following URL:


The response consists of the computed bills in HTML format for display in a web browser. The bills are also sent in CDR format to the Settlement Engine (see section 3.9).

3.9. Settlement Engine

The Settlement Engine processes the information returned by the Business Charging Engine as a file containing one or more CDRs. After processing and storing these files, the Settlement Engine generates a bill, in pdf format, including all relevant information in an appropriate format so that it can be easily understood by the customer (see section 4.8).

The Settlement Engine is installed as part of the Marketplace backend WAR file (see section 3.2), containing all necessary libraries.

The bill generator REST service can be accessed at the following URL:

3.10. Marketplace Repository, Products Catalogue and Contracts Inventory

The 4CaaSt Marketplace Persistence Layer is implemented as a relational database. It hosts the three repositories described in [8]:

- Marketplace Repository
- Products Catalogue
- Contracts Inventory

In particular, MySQL is used, hosted by a different server than the Web application server. Details about the server and repository, as deployed in the FLEXISCALE testbed, are as follows:

- Server:
  - IP address: 109.231.82.98
  - User/password: ubuntu/08U8ovMGX8fio0ba

- MySQL:
  - Port: 3306
  - User/password: root/root
  - Schema: 4CAAST_DEMO
  - Configuration at: /etc/mysql/my.cnf
    - Basedir = /usr/share/mysql
    - Datadir = /var/lib/mysql
4. User guide

The 4CaaSt Marketplace is a Web application that gives users access to the different capabilities provided. Several types of users are identified and are granted different access rights depending on their role. The main user roles are:

- Administrators. They are in charge of the overall management of the Marketplace features, adding and removing users, etc.
- Service providers can specify and publish their service offerings using the Marketplace capabilities.
- Consumers can browse services published in the Marketplace, negotiate some service details and finally contract them.

The goal of this chapter is to give an overview of the main capabilities the Marketplace user interface offers. Note it is not intended as a scenario description, which can be found in other documents like [9].

4.1. Marketplace Management

Marketplace administrators can add new organizations and users to the system, as shown in the following figures.

![Figure 4. Adding a new organization](image-url)
4.2. Publishing products in the Marketplace

Service providers can use the Marketplace to specify and publish their services so they can be easily found and used by consumers. The first step to specify a new product is to import the selected BluePrint template from the available ones (defined outside the Marketplace) as shown in Figure 6.

Once this is done, the service provider can fix some Blueprint characteristics to fit its own needs. In particular, different price, availability and security levels can be selected as shown in Figure 7.
In addition, sophisticated price plans can be assigned to services, as it is explained in section 4.4. See also section 4.7.2 for price model simulations.

Once the service is fully specified, it can be published in the Marketplace (Figure 8).

---

**Figure 7. Selection of blueprint characteristics**

**Figure 8. Publication of a product in the Marketplace**
4.3. Contracting products in the Marketplace

Consumers of the services provided by the 4CaaSt platform can select, contract and deploy services from the 4CaaSt catalogue (Figure 9).

Figure 9. Browsing the 4CaaSt catalogue

Once a given service is selected, the consumer can set requirements on a group of parameters (Figure 10).

Figure 10. Selection of product characteristics

Taking into account the information received, 4CaaSt tries to find the service proposition that best matches the consumer’s needs. This is presented to the consumer as a resolved blueprint (Figure 11).
The consumer can also access detailed price information (Figure 14) before finally contracting the service.

Consequently, there is an interactive negotiation process that involves several blueprint resolutions according to consumer’s requirements until a final service proposal is agreed between 4CaaSt Marketplace and the consumer. At this stage the consumer can contract the product which is subsequently instantiated and deployed.
4.4. Price Editor

Service providers can create and edit the price plans for their products. Consumers can view the price plans for published products.

4.4.1. Edit a price plan

After the service provider has specified a product in the Marketplace its price plan can be accessed under the following URL template:


*Note: replace “70” with the product ID*

Therein the service provider can set the name, description and currency of the price plan as well as an optional maximal price for the summed up price for all its price components (Figure 12). All entries marked with an asterisk are mandatory.

![Figure 12. Creation of price plan](image)

Furthermore it is possible to upload an existing price plan as well as downloading the current price plan as .xml-file.

A click on the *Add Price Component* button adds a price component to the price plan as shown in Figure 12.

To specify that the price component prices at a monthly fee subscription of 50 Euro, the price component *Monthly subscription fee for taxi app* can be configured as shown in Figure 13.

The created or adapted price plan can be saved to the product database through a click on the *Save Plan* button (Figure 12 and Figure 13).
4.4.2. View a price plan

Then the consumer can view the price plan for the taxi app under the following URL template:

  
  Note: replace “70” with the product ID

Therein the consumer can see the price plan for the taxi app in a read only form (Figure 14).
4.5. Ratings and comments

The 4CaaS Marketplace allows its users to share opinions on the services offered by means of textual comments and ratings.

4.5.1. Product ratings

Customers can rate products they have contracted from the 4CaaS marketplace (Figure 15) on a scale from 1 to 10.
Besides, both customers and service providers have access to ratings from other customers (Figure 16).

4.5.2. Product comments

Customers can put comments on products they have contracted from the 4CaaSt marketplace by adding a short textual description of their opinion (Figure 17).
Both customers and service providers have access to comments from other customers (Figure 18).

![Figure 18. View comments](image)

### 4.6. Marketplace Reports

A graphical reporting tool allows users of the 4CaaSt marketplace access relevant information about platform usage, products, contracts, etc. (Figure 19).

This tool can be used from different types of devices (PCs, tablets, smartphones) and operating systems (iOS, Android). It is also integrated in the 4CaaSt marketplace front-end.

![Figure 19. Access to reports](image)

Two types of reports are available, as described in the next sections.

### 4.6.1. Predefined reports

Predefined reports are configured in advance, so the Marketplace users only have to select one of them and set its parameters as required. The main predefined reports available are:

- Products contracted over time, filtered by customer or product (Figure 20).
4.6.2. User defined reports

Users of the Marketplace can use this tool to create their own reports (Figure 22).
To do so, the tool provides a guided navigation through the database structure so that the user can pick up the tables and fields he wants to include in his report (Figure 23).

Once the report is completed, it can be executed and stored for further use in the “favourites” tab (Figure 24).
Figure 24. Favourite reports

The user can also select the type of graph he wants (bar or pie chart) and switch between them (Figure 25 and Figure 26).

Figure 25. Bar chart
4.7. Simulations

The 4CaaSt Marketplace allows service providers running simulations to optimise their business models in different ways as described in the next sections.

4.7.1. Business Model Simulation

Based on the data collected within the “Market and Competition Analysis” task 9.1 (see [10]), we have developed a structured approach that not only allows a systematic design of business models for cloud computing, but also provides a tool to facilitate the business model design process through simulation/analysis.

BMS provides a structured approach that not only allows a systematic design of business models for cloud computing, but also provides a tool to facilitate the business model design process through simulation/analysis. The resulting method is specified by means of a (1.) meta-model and (2.) a procedure model as well as (3.) a software prototype.

Service Providers can launch the Business Model Simulation (BMS) from the main page of the 4CaaSt Marketplace frontend. The User Guide for this simulator, which is embedded in the Business Model Component (BMC), is described in the next paragraphs.

4.7.1.1. Meta model

The meta model (Figure 27) defines the concepts used within the method. Having been developed in iterations with the implementation of the prototype and the procedure model, its left part is influenced by the physical data representation of the business model. The rest of the meta-model, the right and bottom parts, refer to the analysis and simulation.
The business model is the main design object of the method. It always belongs to one business model type (Figure 28). A business model type describes a business model through defining several attributes. An attribute is either a free text value, or it has fixed values the attribute could take. Each attribute belongs to one of the six business model components.

Data Collection Activities (Figure 29) contribute to the data set. A Data Collection Activity refers to certain attributes and uses different channels of data acquisition. The ways to acquire data could be by filling a survey, posting micro-surveys, adding competitive business model instances, utilizing usage data from marketplaces or performing market research activities.

Currently we are using the data collected by the online survey for our analyses, as well as the business model instances investigated [10].
An analysis or simulation can be run for a business model. It typically considers a subset of attributes of the business model to be analysed or changed and a subset of business models to be considered as the environment or the competition. Four different types of simulation variants exist (1.) Attribute tuning, (2.) direct benchmark and (3.) competition ranking (Figure 30).
4.7.1.2. **Procedure Model**

The process of representing, designing, and simulating/analysing cloud business models is specified with a procedure model (Figure 31). It serves as the overall view on all activities in their hierarchical structure and in their sequential order.
The procedure model is divided into three parts:

- Phase I involves the tasks required by the marketplace provider to prepare the method for the service providers.
- Phase II involves the service providers and facilitates the modelling of their business models.
- In Phase III, service providers analyse, simulate, and optimize their business models.

4.7.1.2.1. Phase I: Method Setup and the Acquisition of Data

The first consideration the marketplace provider will take is to define the target users of the tool, i.e., identify scope and customers. This means deciding what types of business models are that the tool should consider, and how they are hierarchically structured (Figure 32). A business model type can be a sub-type of another business model type.

![Business Model Type Setup](image)

**Figure 32. Business Model Type Setup**

Here, three main business model types have been defined, based on the architectural concepts for clouds: SaaS, PaaS and IaaS. In addition to this, the PaaS layer has been differenced into the three sub-types: Development-, Application- and distribution focused, as introduced in [10].

The activity of specifying the cloud-based business meta-model means to define attributes and attribute levels. The marketplace provider thinks about which attributes best describe the respective offering in the respective component (Figure 33).
Attributes of Business Model Types are inherited to sub-types. But sub-types can also have additional attributes. For instance, the PaaS Business Model Type has five additional Attributes. Like for example the Attribute “Development Environment”, with its three Attribute levels “online”, “SDK” and “None” (Figure 34).

A conjoint analysis on PaaS business models serves as an initial data-set. Within the final step of phase I, the marketplace provider instantiates the data set by importing the collected
data into the data structure for PaaS business models so that it can be used as a basis for simulation (Figure 35).

Figure 35. Data Acquisition

4.7.1.2.2. Phase II: Business Model Design

A service provider now can model his business model by selecting a name and description, and assigning a business model type to it (Figure 36).

Figure 36. Add Business Model

The service provider of 4CaaSt platform specifies his PaaS business model and sets the attribute levels for all the attributes that are determined through the PaaS business model type.
At this point the procedure model suggests the first feedback loop which goes back into the method setup and would allow the service providers to suggest or add new attributes and attribute levels to their respective business model type (Figure 37).

Figure 37. Edit Business Model

4.7.1.2.3. Phase III: Simulating Cloud Business Models

The Simulation builds upon existing instances of business models and the data specified and instantiated by the marketplace provider. A service provider can run different kinds of simulations, which can give hints on potential improvements of the business model. Currently, three specific kinds of simulation are suggested (Figure 38).

Figure 38. Simulation

With the competition analysis (Figure 39), cloud providers can compare their own business models with all the business models of the same type. The first main output this type of simulation can generate is the similarity in per cent of the competing business models relative to the service provider’s business model. The second output is a set of market shares for each business model of the respective type. The assumption here is that business models of the same type are in direct competition to each other and split the market among them. The calculation of virtual market shares requires a set of virtual consumers who can
vote for the business model that maximizes their utility. Any kind of method that creates individual utility vectors on a subset of the meta-model’s attribute levels would be suitable here.

Figure 39. Competition analysis

Within the direct benchmark analysis a service provider can compare his own business model with another business model (Figure 40). Just like in the competition analysis case, the other business model has to be of the same type, and the result is one similarity value and two market shares. This kind of analysis allows for more detailed attribute-wise comparison views between two business models.

Figure 40. Direct benchmarking

The attribute variation analysis (Figure 41) is an extension of the competition analysis: The user can select one or more attributes of an existing business model and see how the market share in the competition analysis would change when the attribute levels change. Each combination of attribute levels, for the selected attributes, creates one temporary business model, which is simulated on. For each of these combinations, the user is then provided with the value of the predicted market share.
For each of these combinations, the service provider receives the predicted market share. Here, the highest market share could be reached, if the platforms offers an offline development environment and provides a test environment.

Figure 41. Attribute variation analysis

4.7.2. Price Model Simulation

Service Providers can launch the Price Model Simulation (PMS) as part of the product definition workflow of the 4CaaSt Marketplace frontend. The User Guide for this simulator, which is embedded in the BMC, is described below.

4.7.2.1. Overview

The Price Model Simulation (PMS) is also part of the BMC, and hence, reachable via the start page of the BMC. The whole Price Model Simulation consists of 11 collapsible portlets (Figure 42) that are ordered chronologically ascending to their position in the simulation process.

Through these steps the provider can find an optimal price model of his product for a given usage of the single components that has to be defined (or predicted). After setting an Optimization Goal the results and detailed analysis including costs and revenue will be shown. Finally the simulated (optimal) price model can be edited or directly stored in the Products Catalogue database.

Besides showing the product name the analysis can be started here including the hint that the usage has to be specified before. This can be done in the portlet “Select Price Model Components”.

Finally the results of the analysis are shown in the portlet “Breakdown of Revenue and Cost".
Figure 42. Price Model Simulator

The product graph (Figure 43) gives an overview of all third-party-products the inspected product depends on whereby different layouts can be chosen.
Another portlet shows the specified price models for all products shown in the “View Product Graph” by embedding the Price Editor (see section 4.4). Naturally price models of third party products cannot be edited.

4.7.2.2. **Setting usage of Price Model Components**

For the simulation depends on the specified usage, this component allows that definition by communicating with the portlet “Enter Usage Function”. According to the price models shown in the portlet “View Price Models”, the usage has to be set for each component of each price model (Figure 44).

![Select Price Model Components](image)

**Figure 44. Select Price Model Components**

The Predict Usage Function portlet (Figure 45) allows specifying the usage of the Price Model Component (PMC) being currently selected. It can be defined as constant value or as a function depending on the variables that are listed there. Finally the function is illustrated as a 3d-graphic.

Alternatively, in case of not being able to express the usage function, the portlet “Predict Usage Function” can be used to predict the function by regression.
Alternatively to an accurate usage function this portlet allows predicting that function by taking key-value pairs and performing a polynomial regression (Figure 46).
4.7.2.3. **Price Model Simulation**

Finally, before executing the PMS, a goal has to be set (Figure 47). That can either be minimizing or maximizing revenue, cost, profit or usage. Furthermore there are two conditions that can be selected to (1) make profit and (2) determining maximal usage for Price Model Components of the inspected product.

![Figure 47. Optimization goal](image)

Afterwards, the simulation can be started in this portlet. During the simulation the progress including currently results are displayed (Figure 48).

![Figure 48. Optimize Price Model](image)
4.7.2.4. **Simulation Result**

The result of the simulation consists of an optimal Price Model for the inspected product by optimizing according to the Optimization Goal and considering the conditions and restrictions. This price model can be modified or directly saved.

![Simulation Result](image)

**Figure 49. Optimization Result**

4.7.2.5. **Analysis**

The “Breakdown of Revenue and Cost” portlet (Figure 50) that was started in the portlet “Inspected Product”. Details about the profit, revenue and costs are shown.

Then, in the “Analyze Revenue, Cost, Profit and Break Even” portlet (Figure 51), details are shown regarding revenue, cost, profit and the breakeven point whereby the impact of the single Price Model Components is shown in comparison to the total values.
Figure 50. Breakdown of Revenue and Cost

Figure 51. Analyze Revenue, Cost, Profit and Break Even
4.8. **View Bill**

Customers of the 4CaaSt marketplace can access their bills from the user interface (Figure 52). Each bill is related to a given contract over a period of time.

![Figure 52. View bills](image)

Note the bill contains detailed information about the different concepts the customer is being charged for. Since 4CaaSt deals with composite products, a single contract may encompass charges for the different sub products that make up the final offering (Figure 53).

![Figure 53. Customer bill](image)
5. Conclusions

The information contained in this document allows deploying and operating the second release of the 4CaaSt Marketplace prototype. This prototype, which focuses on trading aspects, needs to be connected to other 4CaaSt building blocks in order to fulfil the overall cloud platform.

Different types of users can interact with the prototype by means of its user interface, which is briefly explained in this document. The main capabilities it provides to each user group (Marketplace administrators, service providers and end users) are described and illustrated by means of screen captures.

In this second iteration, the 4CaaSt Marketplace prototype allows defining and contracting services, simulating business models and accessing statistics about user interactions. Over the next months the prototype will be enhanced with additional functionality in the scope of the third 4CaaSt iteration (see [7] for a description of planned new features).
6. References

[1] FP7 Grant Agreement - Annex II General Conditions Grant Agreement No. 258862, October 2009.


