Document	FUI-10-COMPATIBLE ONE Livrables L 38.1		
Date prévue **Date livraison**	29/11/2012 26/11/2012	**Nature**	Public
Statut	Final	**Version**	1.6

Propriétés du Document

Source du Document	FUI-10-COMPATIBLE ONE
Titre du Document	COAPS :COmpatible Application Platform Service - specifications
Module(s)	PaaS4Dev
Responsable	Institut Télécom
Auteur(s) / contributeur(s)	Sami Yangui – Sami.Yangui@it-sudparis.eu Mohamed Sellami – Mohamed.Sellami@it-sudparis.eu Mohamed Mohamed – Mohamed.Mohamed@it-sudparis.eu Samir Tata – Samir.Tata@it-sudparis.eu
Statut du Document	Final
Version	1.6
Validé par	Benoit Pelletier (Bull) - Benoit.Pelletier@bull.net
Date de la validation	Prévu Novembre 2012

Résumé

This document provides a description of COmpatible Application Platform Service (COAPS for short) the generic PaaS API based on REST/XML.

It describes the allowed actions for both User and Admin roles.

Mots Clefs

API, application, environment, descriptor.

	Document	FUI-10-COMPATIBLE ONE Livrables L 38.1		
	Date prévue **Date livraison**	29/11/2012 26/11/2012	**Nature**	Public
	Statut	Final	**Version**	1.6

Table of Contents

	Document	FUI-10-COMPATIBLE ONE Livrables L 38.1		
	Date prévue **Date livraison**	29/11/2012 26/11/2012	**Nature**	Public
	Statut	Final	**Version**	1.6

1 Introduction

This document provides a description of the generic PaaS API based on REST/XML.

This API is designed to provide an abstraction layer and a middleware for existing PaaS solutions to manage applications and environments via CompatibleOne Accords Platform (Figure 1).

Until now, there are two implementations (Cloud Foundry implementation and OpenShift implementation) of the generic API interfaces that are available.

To define a new connection between a novel PaaS and Accords, one has to simply add its specific implementation as described in Figure 1.
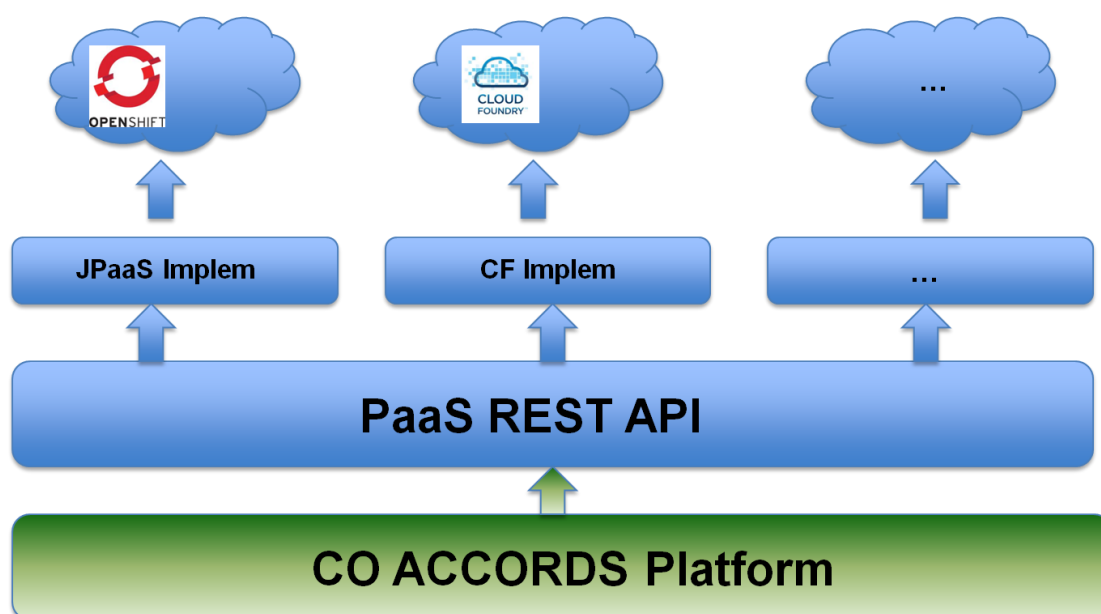

Figure 1 : Overview

The generic API interfaces describes the allowed actions for the user's roles.

The following actions of the REST protocol are used:

REST	CRUD
POST	Create
GET	Read
PUT	Create or Update
DELETE	Delete

	Document	FUI-10-COMPATIBLE ONE Livrables L 38.1		
	Date prévue Date livraison	29/11/2012 26/11/2012	**Nature**	Public
	Statut	Final	**Version**	1.6

2   API authentification

The API authentication system is "Basic". It is essentially based on adding a field "Authorization" in the header of requests containing the encoded string user: password. If the request does not contain this entry, an error "401: Unauthorized" is returned.

3   Common parameters

This section describes all the request parameters that all actions use.

Parameter	Description	Required
Action	The action to perform	Yes
Format	The format of the result. Currently, only XML is supported. Default: XML	No
AccessKeyId	The key id used to sign the request (user name)	
Signature	The digital signature you created for the request.	
SignatureMethod	The algorithm used to sign the request.	
Timestamp	The time at which the query was generated, formatted as the number of seconds since Jan 1, 1970 UTC.	No
Version	The API version to use. Expected value: 0.1.	Yes

4   Common errors

This section lists common errors that all action returns.

Error	Description	HTTP Status Code
Invalid signature	The request signature is not valid	400
Internal Failure	The internal processing has failed due to some unexpected errors.	500
Invalid Action	The action is invalid	400

InvalidKeyId	The key id is invalid	403
InvalidParameter	The parameter is not expected	400
InvalidValue	The value is not expected	400
MissingAuthentication	The key id is missing	403
MissingParameter	The parameter is missing	400
RequestExpired	The request expires	400
ServiceUnavailable	The request has failed due to a temporary failure on the server	503

5 Head command

The HEAD command provides a kind of contextual help and returns in the header a list of accessible uri.

6 API objects description

All API objects are described using the WADL specification (See http://www.w3.org/Submission/wadl/).

WADL descriptions are automatically generated with Jersey using JAX-RS annotations. The WADL is recoverable by a GET request on the url *http:// <host>: <port> /<paas-api>/application.wadl*.

An example is provided in Annex 1.

7 API objects references

We use Links and Link Relations concepts in responses to explore API objects and to navigate between them. These concepts are inspired from by the vCloud project (See http://www.vmware.com/pdf/vcd_15_admin_guide.pdf).

Link elements provide references to objects and the actions that they support. These elements are the primary mechanisms by which a server tells a client how to access and operate on an object. The server creates Link elements in a response body. They are read-only for the client.

Each Link element has the following form:

```
<Link rel="relationship"
type="application/jpaas.object_type+xml"
```

```
href="URL"
name="string"/>
```

Attribute	detail
Rel	Defines the relationship of the link to the object that contains it. A relationship can be the name of an operation on the object, a reference to a contained or containing object, or a reference to an alternate representation of the object. The relationship value implies the HTTP verb to use when you use the link's href value as a request URL.
Type	The object type, specified as a MIME content type, of the object that the link references. This attribute is present only for links to objects. It is not present for links to actions.
Href	An object reference, expressed in URL format. Because this URL includes the object identifier portion of the id attribute value, it uniquely identifies the object, persists for the life of the object, and is never reused. The value of the href attribute is a reference to a view of the object, and can be used to access a representation of the object that is valid in a particular context. Although URLs have a well-known syntax and a well-understood interpretation, a client should treat each href as an opaque string. The rules that govern how the server constructs href strings might change in future releases.
Name	The name of the referenced object is taken from the value of that object's name attribute. Action links do not include a name attribute.

For more details, the list of possible values of *rel* attribute is detailed in vCloud admin guide (See http://www.vmware.com/pdf/vcd_15_admin_guide.pdf).

8   Usage scenario

Below, in order, the necessary operations to deploy and run an application on PaaS using our API.

N°	action	Details
1	createEnvironment	Creates a new environment where the application to deploy should be run.
2	createApplication	Creates the application "envelope".
3	createApplicationVersion	Describes the application' version to deploy.
4	createApplicationVersionInstance	Describes the instance list of the application' version to deploy and its deployable.

5	DeployApplication	Uploads the application deployable to the PaaS platform.
6	startApplication	Starts the application to be available on the target PaaS.
7	stopApplication	Stops the application.

9 Application management actions

Description	Command	Specific parameters	Response
Start an application version instance	POST /app/{appId}/version/{versionId}/instance/{instanceId}/action/start		
Stop an application version instance	POST /app/{appId}/version/{versionId}/instance/{instanceId}/action/stop		
Create a new application. If the application is multitenant, it will be accessible for all tenant An Application Descriptor must be provided.	POST /app	-Application Description -Multi-tenant (yes/no).	Application Description with: - appId - Link element
Create a new version with either a file in attachment (Content-Type: multipart/form-data). or a URL. The supported artifacts are ear, bundle, war, ejbjar or a zip (war dir). An Application Version Descriptor must be provided.	POST /app/{appId}/version	- Artifact - Application Version Description	Application Version Description with: - versionId - Link element
Create a new application version instance. An Application Version Instance Descriptor must be provided.	POST /app/{appId}/version/{versionId}/instance	- ApplicationVersionInstance Descriptor	Application Version Instance Description with: - instanceId - Link element
List applications	GET /app/		Application Description List

List application versions	GET /app/{appId}/version		Application Version Description List
List application version instances	GET /app/{appId}/version/{versionId}/instances		Application Version Instance Description List
List application version artefacts	GET /artefact/{appId}/{versionId}		Application VersionArtefactsList
Describe application.	GET /app/{appId}		Application Description
Describe application version	GET /app/{appId}/version/{versionId}		Application VersionDescription
Describe application version instance	GET /app/{appId}/version/{versionId}/instance/{instanceId}		Application VersionInstanceDescription
Delete application. Removes all existing versions	DELETE /app/{appId}/delete		
Delete application version	DELETE /app/{appId}/version/{versionId}/delete		
Delete application version instance	DELETE /app/{appId}/version/{versionId}/instance/{instanceId}/delete		
Delete all applications	DELETE /app/delete		

10 Environment management actions

Description	Command	Specific parameters	Response
Start an environment	POST /environment/{envId}/action/start		
Stop an environment	POST /environment/{envId}/action/stop		
Deploy an application instance on an available environment	POST /environment/{envId}/action/deploy/app/{appId}/version/{versionId}/instance/{instanceId}		
Undeploy an application instance on an available	POST /environment/{envId}/action/undeploy/ap		

environment	p/{appId}/version/{versionId}/instance/{instanceId}		
Creates a new environment. An environment template descriptor must be provided.	POST /environment/	Environment Description	Environment Description With: - envId -Link element
Deletes an environment	DELETE /environment/{envId}		
List the available environments	GET /environment		Environment Description List.
Get the description of an environment	GET /environment/{envid}		Environment Description.
List the deployed application instances in an environment	GET /environment/{envId}/app/		Application Version Instance Description List.

	Document	FUI-10-COMPATIBLE ONE Livrables L 38.1		
	Date prévue Date livraison	29/11/2012 26/11/2012	**Nature**	Public
	Statut	Final	**Version**	1.6

11   Annex 1: WADL description

```xml
<application xmlns="http://research.sun.com/wadl/2006/10">
  <doc xmlns:jersey="http://jersey.java.net/" jersey:generatedBy="Jersey: 1.8 06/24/20
  12:17 PM"/>
  <resources base="http://127.0.0.1:8080/CF-api/rest/">
    <resource path="app">
      <method id="createApplication" name="POST">
        <request>
          <representation mediaType="application/xml"/>
        </request>
        <response>
          <representation mediaType="application/xml"/>
        </response>
      </method>
      <method id="findApplications" name="GET">
        <response>
          <representation mediaType="*/*"/>
        </response>
      </method>
      <resource path="{appId}/version/{versionId}">
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template
        type="xs:string"/>
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="versionId"
        style="template" type="xs:string"/>
        <method id="findApplicationVersionInstances" name="GET">
          <response>
            <representation mediaType="*/*"/>
          </response>
        </method>
      </resource>
      <resource path="{appId}/version/{versionId}/instance">
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template
        type="xs:string"/>
        <method id="createApplicationVersionInstance" name="POST">
          <request>
            <representation mediaType="application/xml"/>
          </request>
          <response>
            <representation mediaType="application/xml"/>
          </response>
        </method>
      </resource>
      <resource path="{appId}/version/{versionId}/instance/{instanceId}/action/start">
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template
        type="xs:string"/>
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="instanceId"
        style="template" type="xs:string"/>
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="versionId"
        style="template" type="xs:string"/>
        <method id="startApplicationVersionInstance" name="POST">
          <response>
            <representation mediaType="*/*"/>
          </response>
        </method>
      </resource>
      <resource path="{appId}/version/{versionId}/instance/{instanceId}/action/stop">
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template
        type="xs:string"/>
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="instanceId"
        style="template" type="xs:string"/>
        <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="versionId"
        style="template" type="xs:string"/>
        <method id="stopApplicationVersionInstance" name="POST">
          <response>
            <representation mediaType="*/*"/>
```

```
      </response>
    </method>
  </resource>
  <resource path="{appId}">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <method id="describeApplication" name="GET">
      <response>
        <representation mediaType="application/xml"/>
      </response>
    </method>
  </resource>
  <resource path="{appId}/delete">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <method id="deleteApplication" name="DELETE">
      <response>
        <representation mediaType="application/xml"/>
      </response>
    </method>
  </resource>
  <resource path="delete">
    <method id="deleteApplications" name="DELETE">
      <response>
        <representation mediaType="application/xml"/>
      </response>
    </method>
  </resource>
  <resource path="{appId}/version/create">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <method id="createApplicationVersion" name="POST">
      <request>
        <representation mediaType="*/*"/>
      </request>
      <response>
        <representation mediaType="application/xml"/>
      </response>
    </method>
  </resource>
  <resource path="{appId}/version">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <method id="findApplicationVersions" name="GET">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
</resource>
<resource path="environment">
  <method id="findEnvironments" name="GET">
    <response>
      <representation mediaType="application/xml"/>
    </response>
  </method>
  <method id="createEnvironment" name="POST">
    <request>
      <representation mediaType="application/xml"/>
    </request>
    <response>
      <representation mediaType="application/xml"/>
    </response>
  </method>
  <resource path="{envId}">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
```

```xml
    <method id="getEnvironment" name="GET">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
    <method id="deleteEnvironment" name="DELETE">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
  <resource path="{envId}/app/">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
    <method id="getDeployedApplicationVersionInstance" name="GET">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
  <resource path="{envId}/action/start">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
    <method id="startEnvironment" name="POST">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
  <resource path="{envId}/action/stop">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
    <method id="stopEnvironment" name="POST">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
  <resource
path="{envId}/action/deploy/app/{appId}/version/{versionId}/instance/{instanceId}">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="instanceId"
    style="template" type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="versionId"
    style="template" type="xs:string"/>
    <method id="deployApplication" name="POST">
      <response>
        <representation mediaType="*/*"/>
      </response>
    </method>
  </resource>
  <resource
path="{envId}/action/undeploy/app/{appId}/version/{versionId}/instance/{instanceId}">
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="envId" style="template"
    type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="appId" style="template"
    type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="instanceId"
    style="template" type="xs:string"/>
    <param xmlns:xs="http://www.w3.org/2001/XMLSchema" name="versionId"
    style="template" type="xs:string"/>
    <method id="undeployApplication" name="POST">
      <response>
        <representation mediaType="*/*"/>
```

```
        </response>
      </method>
    </resource>
   </resource>
  </resources>
</application>
```