

Amenesik

Cloud Engine

User Guide Version 1.0a

Iain James Marshall
19/06/2016

Table des matières

Introduction	1
Components.....	1
Database Services	1
Publication Service.....	1
Security Service	2
The Kernel Services.....	2
Cloud Deployment Services	2
Remote Command Interface.....	2
Deployment Models.....	3
Small Model	3
Medium Model	3
Large Model	4
References	6
OCCI.....	6
TOSCA.....	6
CIMI.....	6
CORDS	6
AMENESIK	6

Introduction

This component of the Amenesik Enterprise Cloud, based on an industrialisation of the CompatibleOne Accords Platform, is responsible for the management of all aspects of cloud application descriptions and their run time deployment. It presents the traditional OCCI interface allowing direct and authenticated access to all components of the CORDS model.

Components

The engine has been entirely refactored for operational efficiency, using the Amenesik Platform Editor, resulting in the following individual functional components.

Database Services

This component provides the centralised data management system used for the storage of instance data for all category managers of the Amenesik Cloud Engine components. This is usually MYSQL but POSTGRESQL or CASSANDRA could be used instead if so required.

In a standard deployment configuration, the MYSQL Database Server will usually be listening on the TCP port number 3306.

Publication Service

This component provides the centralised service registry through which all OCCI categories are made known and made available for use by all other components of the system.

In a standard deployment configuration, the Publication Service will usually be listening on the TCP port number 8086 and expecting HTTPS secure communication.

Security Service

This component provides the categories required for the implementation and support of the Amenesik Security Policy. This is based on the account, user and authorisation categories of the standard OPENID model working in conjunction with the role, rule and policy categories of the ORBAC model.

In a standard deployment configuration, the Security Service will usually be listening on the TCP port number 8087 expecting HTTPS secure communication.

The Kernel Services

This component provides the CORDS model management categories for manifest definition, agreement definition, generic service deployment and management and for monitoring purposes. The kernel also contains the scheduling and timer categories responsible for driving the various events of the Service Instance Life Cycle and for the general maintenance of the Amenesik Cloud Engine.

In a standard deployment configuration, the Kernel Services will usually be listening on the TCP port number 8088 expecting HTTPS secure communication.

Cloud Deployment Services

This component provides the various cloud provisioning categories for IAAS, PAAS and SAAS levels of service deployment.

In particular, the following cloud technologies are supported:

- Amazon Elastic Cloud Compute EC2
- Google Compute Engine GCE
- Microsoft Windows Azure
- OpenStack
- OpenNebula
- SoftLayer
- CloudSigma

In a standard deployment configuration, the Cloud Services will usually be listening on the TCP port number 8089 expecting HTTPS secure communication.

Remote Command Interface

The engine is accompanied by the Amenesik Remote Command Interface which can be accessed either over HTTP through a SOAP interface and its adjacent WSDL definitions document or through the standard HTML Web Browser FORMS interface.

In a standard deployment configuration, the Remote Command Server will usually be listening on the TCP port number 8186 expecting HTTPS secure communication.

The Remote Command Service allows operations at the document level to be performed such as manifest and agreement parsing and the execution of CORDSCRIPT programs and behaviour. The Remote Command Server also offers document conversion operations from TOSCA to MANIFEST format and from UML to AGREEMENT formats. In addition, the Remote Command Server offers a Service Catalogue interface compatible with the requirements of the Cloud Foundry Platform allowing the Amenesik Cloud Engine to be integrated with this platform as a Service Brokerage Engine.

Deployment Models

An operational instance of the Amenesik Cloud Engine, resulting from the industrialisation of the CompatibleOne Accords Platform, will be deployed by the Amenesik Platform Service based on a specific deployment model chosen amongst those models made available by the Amenesik Platform Editors. Other than the collection of category instances of which the platform is comprised, three distinct variations of the deployment model are available. In all deployment models the database name and access credentials may be specified in the service level agreement. If they are not specified they will default to “accords_database”, “root” and “accords-platform” respectively. In all cases access to the Amenesik Cloud Engine will be performed through the various Amenesik User Interface tools by the appropriately trained personnel. Direct access to the tool will be performed only by members of Amenesik or representatives having followed an intensive training.

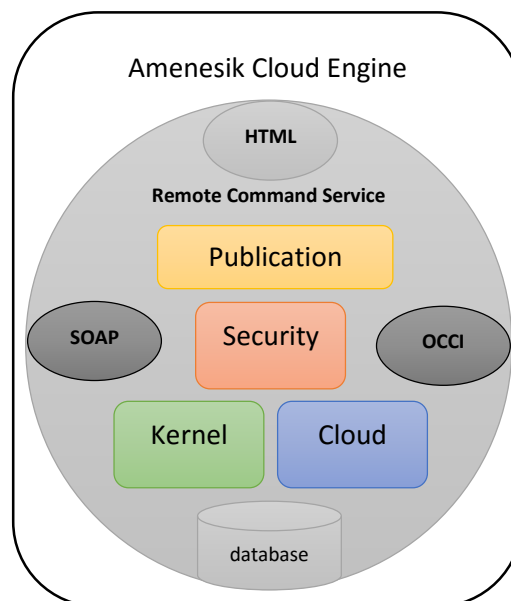
Small Model

In the case of the small model the platform components comprising the model will all be deployed as individual processes within the same single virtual machine.

The size of the hosting machine will be determined by the compute characteristics of the defined platform model in terms of RAM, DISK and processor cores.

In the Small Model the underlying database will be private and the standard user interface packages will also be deployed for use on the same machine.

The following diagram depicts the components of the Small Model within a single virtual machine.



Medium Model

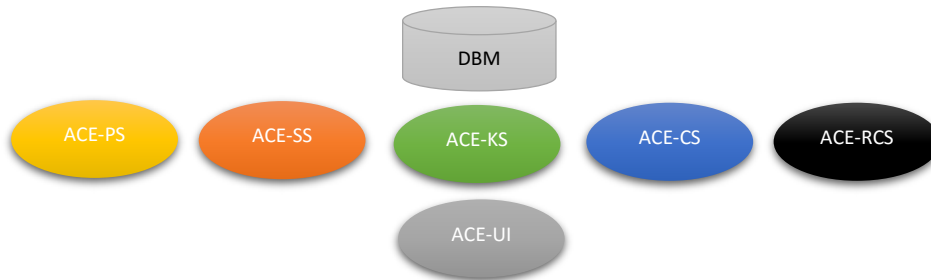
In the case of the Medium Model, the platform components comprising the model will all be deployed as individual virtual machines.

The size of the hosting machines will be by the compute characteristics of the defined platform model in terms of RAM, DISK and processor cores.

The underlying database will be deployed on its own individual virtual machine and will therefore be public.

The user interface packages and associated apache server will be installed in an additional front end machine.

The following diagram depicts a standard seven virtual machine Medium Model deployment:



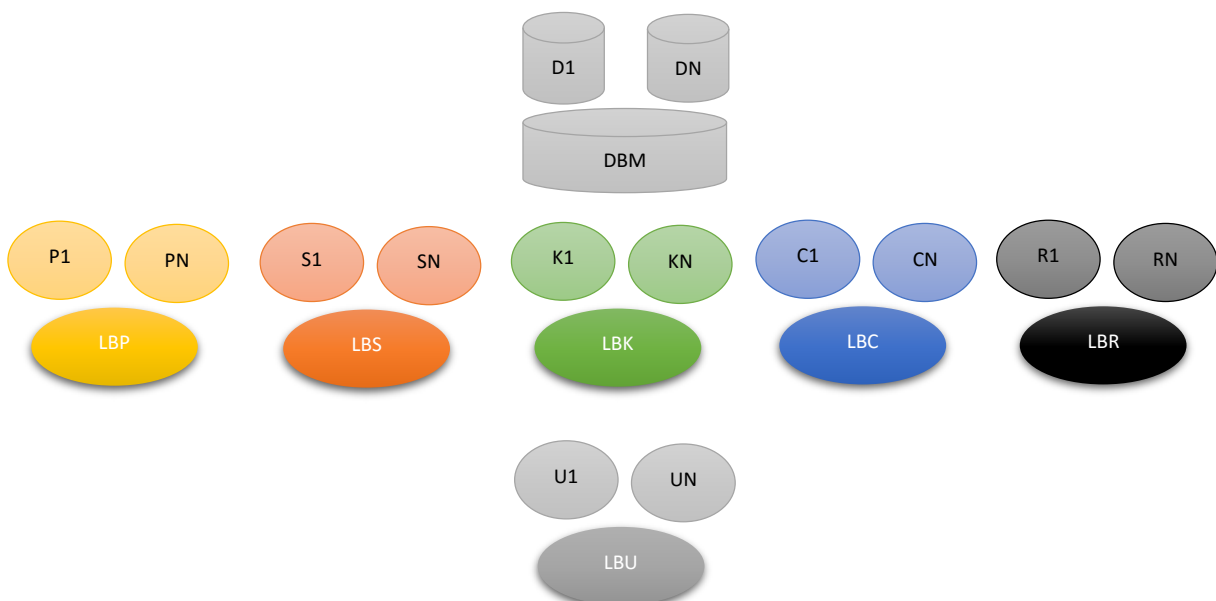
Large Model

In the case of the Large Model the platform components comprising the model will all be deployed as load balanced clusters of virtual machines. The size of the hosting machines will be by the compute characteristics of the defined platform model in terms of RAM, DISK and processor cores. The underlying database will be deployed on its own cluster of virtual machine and will therefore be public. The user interface packages and associated apache server environment will also be installed on a load balanced cluster of machines.

The overall scalability of the different clusters will be defined in terms of their elastic floor and elastic ceiling and the associated elastic threshold values.

The following diagram depicts a large model deployment where:

- D1, DN and DBM represent the elements of the database array
- P1, PN and LBP represent the elements of the load balanced publication service cluster.
- S1, SN and LBS represent the elements of the load balanced security service cluster.
- K1, KN and LBK represent the elements of the load balanced kernel service cluster.
- C1, CN and LBC represent the elements of the load balanced cloud service cluster.
- R1, RN and LBR represent the elements of the load balanced remote command service cluster.
- U1, UN and LBU represent the elements of the load balanced user interface cluster.



In the Large Model the URL of the load balancer endpoint of each load balanced cluster will be used as the component identity for inter-component communication ensuring that load balancing is correctly performed across the available worker nodes of the cluster. With an elastic floor value of 10 and an elastic ceiling value of 100 this would require the initial deployment of 77 and a maximum deployment of 770 virtual machines.

References

This section of the document provides a collection of links to cloud standards documentation and Amenesik support documents.

OCCI

The following documents are available from the OGF web site:

- OCCI CORE Version 1.1:
<https://www.ogf.org/documents/GFD.183.pdf>
- OCCI INFRASTRUCTURE Version 1.1 :
<https://www.ogf.org/documents/GFD.184.pdf>
- OCCI http Version 1.1 :
<https://www.ogf.org/documents/GFD.185.pdf>

TOSCA

The following documents are available from the OASIS web site

- TOSCA Version 1.1:
<http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.pdf>
- TOSCA Namespace:
<http://docs.oasis-open.org/tosca/ns/2011/12>

CIMI

The following documents are available from the DMTF web site

- CIMI Version 1.1 :
http://www.dmtf.org/sites/default/files/standards/documents/DSP0263_1.0.1.pdf

CORDS

The following documents are available from the CompatibleOne community web site:

- CORDS Version 1.1 :
<http://www.compatibleone.com/community/wp-content/uploads/2014/05/CordsReferenceManualV2.15.pdf>

AMENESIK

The following documents are available from the AMENESIK web site:

- Amenesik Enterprise Cloud (AEC) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikCloud.pdf>
- Amenesik Cloud Engine (ACE) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikCloudEngine.pdf>
- Amenesik Manifest Editor (AME) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikManifestEditor.pdf>
- Amenesik Agreement Editor (ASE) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikServiceEditor.pdf>
- Amenesik Service Dashboard (ASD) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikServiceDashboard.pdf>
- Amenesik Cloud Operator (ACO) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikCloudOperator.pdf>
- Amenesik Platform Editor (APE) Version 1.1:

<http://www.amenesik.com/cloud/AmenesikPlatformEditor.pdf>

- Amenesik Platform Service (APS) Version 1.1:
<http://www.amenesik.com/cloud/AmenesikPlatformService.pdf>