Delivering integrated high-level services at Bari IBiSCo-ReCaS site through the PaaS dashboard and the Orchestration system

Mauro Patano on behalf of:
Marica Antonacci, Giovanni Savarese, Ettore Serra, Marco Tangaro

IBiSCo Workshop
Napoli - April 18-19, 2024

Table of Contents

Overview

Cloud Computing

is a de facto standard for modern data centre implementation and a foundation for cloud-native application deployments. It is used by the majority of organisations worldwide.
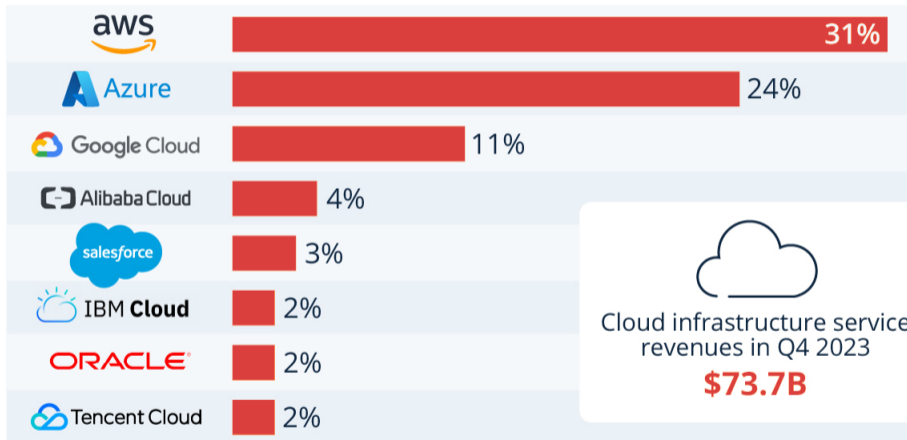
WorldWide Keyplayers

Amazon Web Services, Microsoft Azure, Google Cloud Platform, Alibaba Cloud, offer a wide service assortment, readily accessible to customers.

World Market

[1]

aws	31%
▲ Azure	24%
Google Cloud	11%
Alibaba Cloud	4%
salesforce	3%
IBM **Cloud**	2%
ORACLE®	2%
Tencent Cloud	2%

Cloud infrastructure service revenues in Q4 2023
$73.7B

[1]Worldwide market share of leading cloud infrastructure service providers in Q 2023. Includes PaaS and IaaS as well as hosted private cloud services. Source: Synergy Research Group

Services

Delivery Modalities

Software as a Service (SAAS), Data as a Service (DAAS), Hardware as a Service (HAAS), Platform as a Service (PAAS), and Infrastructure as a Service (IAAS), provide a heterogeneous set of resources with varying degrees of ease in use and operation.

Data Centers Research Oriented

This model of service supply and access is becoming increasingly important, even in general purpose Data Centers with a significant focus on research, such as ReCaS Bari, or in data center federations like INFN-Cloud, of which ReCaS is a member.

Advantages

Users can access resources in a transparent and easy way, with very short waiting times. Admins and support team can limit the manual operations and configurations needed to support the users in their daily activities.

- Cost Effective
- Time Savings
- Future-Proof
- Increase Security
- Dynamically Scale
- Custom Solutions
- Flexibility

User Needs

Our users

come from different scientific communities (high energy physics, bioinformatics, medical physics, engineering...) and some local small and medium-sized enterprises.

Virtualization

Ever-increasing demand for virtual computer hardware platforms, storage devices, and computer network resources.

Containers

this lightweight virtualization technology dramatically simplifies the distribution and deployment of the software encapsulating the runtime dependencies in a single package.

Container Orchestration

system for automating software deployment, scaling, and management.

The ReCaS PaaS Orchestrator: three facts

1. The PaaS Orchestrator allows access to distributed cloud compute and other resources on top of cloud in a transparent and federated way

2. Users can easily deploy services without having to worry about where the resources are available and how to create and configure the resources they need

3. A set of "pre-cooked" service templates are available through the Orchestrator Web Dashboard: once the user is logged in, she/he can choose different categories of services: from the instantiation of virtual machines, to the automatic installation of softwares like docker, docker-compose, to the deployment of complex architecture such as kubernetes clusters.

Foundations and key enablers

- Develop and integrate open-source and open standard-based components to ensure portability and interoperability
- Adopt a modular micro-service architecture
- Use Infrastructure as Code (IaC), DevOps and containers to reduce manual processes and increase flexibility and portability across environments
- Leverage federated Authentication and Authorization technologies based on OpenID-Connect (supporting also legacy AAI solutions) like ReCaS-Bari-IAM
- Enable flexible service composition and re-use

ReCaS PaaS - Main features

- Enabling the federation of distributed and heterogeneous compute environments: clouds, docker orchestration platforms
  - Further integrations can be easily included implementing new plugins (adapters)
- Smart scheduling $\rightarrow$ Automatic selection of the provider
  - First level based on the deployment type (Cloud, Docker, Kubernetes)
  - Second level based on SLAs and compute/storage requirements vs provider capabilities
    - Support for specialized hardware (GPU, Infiniband)
    - Data location
  - Best provider selection based on:
    - Resource quotas (SLA)
    - Monitoring data
  - Automatic retry of failed deployments
- Client interfaces for advanced users (REST APIs, CLI, python bindings) and end-users (web dashboard - no skills required)

ReCaS PaaS Platform

The Orchestrator ensures the transparent access and exploitation of the resources.
RECAS-IAM (based on INDIGO IAM) is used to manage the users identities and permissions

The Dashboard

Nodes and Services

The system architecture consists of five virtual machines on which the microservices (docker containers) run:

- Node recas-paas, services:
    - orchestrator-dashboard
    - redis
    - nginx
    - vault

- Node recas-orchestrator, service: orchestrator
- Node recas-cmdb, services:
    - slat
    - cpr
    - cmdb
    - couchdb

- Node recas-im, service im
- Node recas-paas-db, service: Mysql db

Key Points I

The Orchestrator interacts with the provider services through:

- the Infrastructure Manager for deploying complex and customized virtual infrastructures on multiple IaaS Cloud backends (Openstack. . .)
- direct APIs for deploying dockerized workloads on container platforms.

ReCaS Bari IAM manages the authentication through the whole stack:

users are organized in different IAM groups, each group can access a specific set of services from the dashboard.

Key Points II

The catalogue is a graphical representation of the TOSCA templates:

- Each card in the catalogue is associated to one or more templates
- The configuration form allows the user to specify requirements for the deployment in a straightforward way checking the mandatory fields hiding the complexity of TOSCA

A notification system is implemented in the Dashboard:

the user receives an automatic email as soon as the deployment is ready. Then, the details about the deployed service can be accessed through the Dashboard.

The Dashboard is integrated with Hashicorp Vault (Secrets Manager)

to support some functionalities (e.g. ssh key pair management, service credentials store)

Conclusions

ReCaS PaaS facilitates the deployment of complex services both for end users with different levels of technical skill and for maintainers who see most repetitive work completed automatically.

The dashboard allows the deployment of the required services by choosing a set of options available in easy-to-understand forms. The technologies behind the platform (tosca templates, tosca types, OpenID identity and access management, Ansible . . .) are transparent to the user.

We will gradually add other "plugins" to enable other features/services.