The Alliance Cloud Connect Pilot

MVP Description

Digital Research Alliance of Canada | Alliance de recherche numérique du Canada

1. Cloud Connect MVP Description

1.1 Introduction

This document provides a description of the features required for the development of the Alliance Cloud Connect (ACC) Minimum Viable Product (MVP), as well as a list of desirable features. This will be used to inform the development of the calls for participation for the ACC MVP.

1.1.1 Purpose

The Alliance Cloud Connect Pilot (ACCP) will develop a single “pane of glass” or portal for researchers to access community cloud and key commercial cloud services (CSP), providing access to these services on demand and in a way that simplifies the administrative and technical challenges.

The ACCP addresses the following opportunities, all leading to better access to compute, software and data for the Canadian research community.

The ACCP requirements are described by component. While modular, these components have various levels of interdependency.

The ACCP MVP consists of:

1. **Provider: Cloud service provider (CSP) services/infrastructure:** Primarily IaaS (Infrastructure-as-as-Service) - directly usable by researchers allowing them to deploy self-service virtual compute, storage, networking, etc. A managed Kubernetes (K8S) service will be provided by the cloud providers. A number of PaaS (Platform-as-a-Service) and SaaS (Software-as-a-Service) services are also needed, such as databases, CI/CD, source code management, application platforms, HPC, AI and machine learning, etc. Researchers will be able to stand up their compute workloads, web applications, storage, virtual networking, containers, databases etc on the cloud providers’ platforms.

2. **Platform**

   The platform is made up of six components.

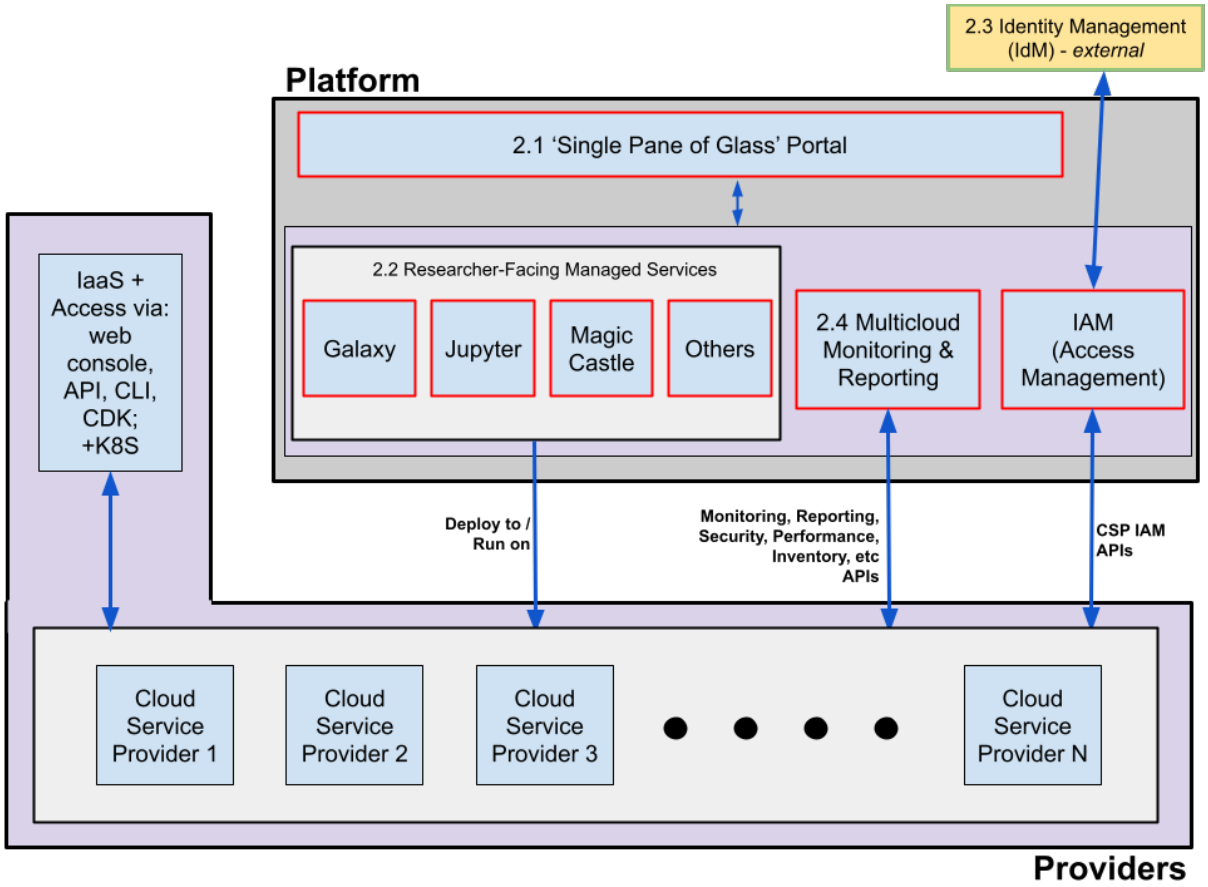
   1. **‘Single Pane of Glass’ Portal (Component #1):** A web interface (portal) providing researchers a single integrated view across all of the platform services (2,3,4 below), and pass-through links to the CSPs’ web consoles. The development of this component will require integration with the below components and links to the CSPs’ web consoles. The portal will also provide Alliance and community admins integrated administration views of all of the components.

2. **Researcher-Facing Managed Services**: A set of Alliance and/or community managed, CSP-hosted, researcher-facing specialized compute services, which will allow researchers to deploy their own instances of these services. These services (components) for this pilot stage are: Jupyter **(Component #2)**, Galaxy **(Component #3)**, Magic Castle **(Component #4)**. The user interfaces for the deployment of these services by researchers will be integrated into the portal. Additional researcher-facing services (*Others* **- Component #5**) will be considered where relevance, cost and community support justifies their inclusion in the pilot.

3. **Identity and access management (IAM) (Component #6)** that a) delegates *identity* management to a federated authority; and b) is the authority for *access* management, and propagates access management policy to the cloud providers. Access management in this context is primarily the ability to invoke, manage, view, etc CSPs' services/infrastructure. Researchers will be able to sign on with their existing credentials and access cloud resources *as per* their role in their projects. An administrative view will allow Alliance and community admins to manage accounts, CSP access, etc.

4. **A multicloud reporting and management dashboard (Component #7)** allowing researchers and Alliance and community admins to monitor, control and optimize cloud costs, infrastructure and services. The dashboard will offer a unified view across the cloud providers, supporting roll ups by researcher, project, cloud provider, etc. The dashboard will support reporting and alerting for cost management, intrusion and anomaly detection, compliance, security policy, network issues, performance issues and access policy issues.

This diagram provides a general overview of the ACCP.



1.1.2 General Principles

1. Users will be issued user accounts under the Alliance organization (billing account) for each ACCP CSP they choose to use. Users in this context are users that *instantiate* resources and *deploy* software on those resources. Researchers *using* the ACCP-deployed resource do not necessarily have to have user accounts on the ACCP.
2. Users will run all of their CSP services/infrastructure under their Alliance CSP account.
3. Researchers' access to all CSP infrastructure and services will be managed by the ACCP IAMcomponent.

1.2 Components

1.2.1 Infrastructure and Services

1. **Provider**
    a. Provide base cloud services supporting the ability to design, secure, test, deploy, monitor, manage and cost-out/cost-manage cloud infrastructure, services and applications, including hosting public facing web applications

b. IaaS (Infrastructure-as-a-Service), PaaS (Platform-as-a-Service), FaaS (Function-as-a-service), *some* SaaS (Software-as-a-Service) services are to be provided.
c. CSPs must provide access to their cloud services and infrastructure via web console, API, CLI and Cloud Development Kit (CDK)
d. CSP services *should* include automated (AI and other) services to analyse user deployments and make optimization and best practice recommendations regarding performance, budgeting, cost management, cost safeguards, efficiency, security, configuration, sustainability (carbon footprint).
e. CSPs' must make available consulting services to the Alliance and researchers, supporting architecture, security, application development, deployment and cost management/optimization.
f. CSP services must support the enforcement of tag policies (in order to implement the ACCP tagging strategy).
g. Specialized mandatory CSP services:
    i. Managed Kubernetes (K8S) Service: A fully managed and CSP supported Kubernetes (K8S) service (e.g. Google Kubernetes Engine (GKE); Amazon EKS; Azure Kubernetes Service (AKS)
    ii. Quantum computing: Managed cloud quantum computing service (e.g AWS Braket; IBM Quantum Platform; Azure Quantum)
    iii. Artificial Intelligence/Machine Learning (ML) Services: VMs supporting GPUs and managed AI/ML services

2. **Platform**
    a. **IAM (Identity and Access Management)**
        i. Identity Management: Will initially use CILogon, supporting registration of the portal via EduGain. If successful, the pilot will extend support to include CDDB as well as other IdPs, such as AWs, Google, Azure, ORCID.
        ii. Authority Management: An authority provider will be implemented to manage access by researchers to the CSP services/resources
            1. Is the authoritative source for *access* management to the CSPs
            2. Manages access managements of the CSPs (propagates auth policies to CSPs)
            3. Initiates cloud provider account creation and management
            4. Centralized ACCP user authorization/access for CSP deployments by researchers
            5. Delegates and maps access control to CSP native IAM
        iii. Federated
        iv. SSO
        v. Must support SAML
        vi. Optional
            1. Ability to create and manage: Teams, PIs, Projects
            2. 2FA

b. **Researcher-Facing Managed Services**

Principles

Two types of deployments of these services are required:

i. General users

The researcher-facing managed services are to provide simple, web access via the portal for the customization, setup, deployment, management and monitoring of their services. Researchers and their teams can instantiate these services (self-serve) for their use using the portal web interface exclusively.

ii. Advanced users

The researcher-facing managed services are also to provide IaC deployments, to be used by researchers with advanced technical abilities and specialized requirements that might not be easily fulfilled using the portal deployment.

Infrastructure-as-Code

1. Implementations are required to use Infrastructure as Code (IaC), defined here as provisioning using one or more of the following:
    a. Terraform/Open Tofu
    b. Pulumi
    c. CFEngine
    d. Chef
    e. Puppet
    f. SaltStack
    g. Ansible
    h. Helm
    i. Kustomize
2. For CSP-specific deployment, the CSP-native IaC may be acceptable. Examples: AWS CloudFormation, Azure ARM templates
3. Deployment using CSP-specific CLI (command line interface), API and CDK (cloud development kit) may be considered with justification
4. The deployment IaC *should* provide cost (and potentially carbon footprint) estimates for deployment, at the time of *pre-deployment*, to users and administrators. Example: Terraform Cost Estimation
5. Must adhere to the ACCP tagging strategy (to be developed by the pilot)
6. All IaC code, scripts, templates etc. must be in code repositories accessible to Alliance admins.

The Services

1. **Galaxy Service**

   Allows users to deploy and manage any number of Galaxy instances for their work onto a CSP's infrastructure.

   Two deployment implementations:

   1. General users: Definable and deployable Galaxy instances using a simple web interface
      a. Must be deployable on at least two of the CSPs
   2. IaC (advanced users)
      a. Must be deployable on at least one of the CSPs

2. **Jupyter Notebooks Service**

   Allows users to deploy and manage any number of Jupyter Notebooks instances for their work onto a CSP's infrastructure.

   Two deployment implementations:

   1. Average users: Definable and deployable using a simple web interface
      a. Must be deployable on at least two of the CSPs
   2. IaC (advanced users)
      a. Must be deployable on at least one of the CSPs

3. **Magic Castle Service**

   Allows users to deploy and manage any number of Magic Castle instances for their work onto a CSP's infrastructure.

   Two deployment implementations:

   1. General users: Definable and deployable using a simple web interface
      a. Must be deployable on at least two of the CSPs
   2. IaC (advanced users)
      a. Must be deployable on at least two of the CSPs

4. **Other**

   Additional researcher-facing services that can provide important services to the community are solicited. Justification is primarily based on researchers' needs/demands, the cost/time/resources that would be needed for implementation and operation in the pilot, and that the service meets the needs of researchers at the national or international level (ie. is not restricted to regional or institutional use). For research groups that are "advanced users" and are able to provide their

platform as an instance of an IaC workflow (e.g. a Terraform deployment), it is desirable to make that deployment easily accessible to end users through a simple web interface. Examples might include a genomics service (examples: iReceptor, virtTool, IRIDA). See also use case document.

c. **Multi Cloud Monitoring and Reporting, Including Community Cloud**

One or more web applications (dashboard(s)) supporting unified monitoring and reporting views across all CSPs. User views: by researcher, by project, by CSP, etc. Views for Alliance and community admins for advanced usage. The following three areas are to be supported and may be delivered with a single solution, or more than one solution:

i. Financial dashboard, reporting, monitoring, alerting, and safeguarding
ii. Performance/Operational/Infrastructure dashboard, reporting, monitoring and alerting
iii. Security dashboard, reporting, monitoring and alerting

1.2.2 Misc (catch all)

- Development and documentation of the ACCP tagging strategy

Out of Scope

1. Financial transactions with universities / funders (but *tracking* of cloud costs is strongly in-scope)
2. Protected B (sensitive information / data)
3. Marketplace (Third-party services running on the CSP). Example: AWS Marketplace https://aws.amazon.com/marketplace