FutureGateway
A new multi-infrastructure framework for customisable Science Gateways

Tomasz Zok $^{1,2}$, Marco Fargetta $^3$, Riccardo Bruno $^3$, Roberto Barbera $^3$, Marcin Plociennik $^1$, Michal Owsiak $^1$, Michal Urbaniak $^1$, Giovanni Aloisio $^4$, Sandro Fiore $^4$

$^1$ Poznan Supercomputing and Networking Center
$^2$ Poznan University of Technology
$^3$ Istituto Nazionale di Fisica Nucleare
$^4$ Euro-Mediterranean Center on Climate Change

2017-06-19
Outline

Introduction

Motivation
Design Principles

Future Gateway

Architecture
Components
Frontend
Backend

Climate Modeling

Introduction
INDIGO DataCloud
Introduction
Motivation

- Research requires an ever-increasing amount of both computational power and storage space.
- Distributed Computing Infrastructures (DCIs) is a solution, but . . .
- . . . it brings its own set of problems – users need to learn to operate DCIs and keep up with changes and novel technologies.
- Science Gateway is an answer to that problem.
Design Principles

- Easier installation and maintenance
  - Ready to use installation scripts and Ansible role
  - Public open source project at GitHub
  - Configurable and customizable

- Flexible access to DCIs
  - Plugin modules using JSaga
  - PaaS access through TOSCA

- RESTful API
  - Well known standard for many programming languages
  - Supports both desktop and mobile applications
  - Hides complexity of back-end operations
Main Entities

Application

Definition of activity to be done

Infrastructure

Environment where an application can run

Task

An instance of an application bound to specific infrastructure
View a Tasks Details

Retrieves the details of the specified task.

Create a new task

This method will create a new task with a specific ID assigned by the user. If the task already exist the task is not modified but an error is returned to the user because the tasks are not modifiable but only some parameters which have specific APIs for their update.

Modify a task

This method will modify the task with a specific ID assigned by the user. Currently only the status can be modified and the only value accepted is CANCELLED. This has the effect to stop the task, free the associated resources and clean the temporary storage.

Delete a task

Application Collection
Components

Access to FutureGateway is customizable:
- through a full portal experience or
- via separate interfaces to frontend and backend

Horizon 2020 Programme
## Components

<table>
<thead>
<tr>
<th>Database</th>
<th>Maintains information about Applications, Infrastructures and Task, but also about the queue and user roles and groups.</th>
</tr>
</thead>
<tbody>
<tr>
<td>API Server Frontend</td>
<td>Fulfills RESTful API communication. Manages AuthN/AuthZ and definitions of Applications, Infrastructures and Tasks. Accepts and queues new Tasks.</td>
</tr>
<tr>
<td>API Server Daemon</td>
<td>Polls for new Tasks. Interfaces concrete DCIs to perform actions consistently. Retrieves output. Extendable for new DCIs.</td>
</tr>
</tbody>
</table>
GUI sends a REST request

Frontend checks with Authentication and Authorization Infrastructure (AAI)

Actions to be done are queued

Response is prepared
Frontend

- Available on GitHub: https://github.com/indigo-dc/fgAPIServer
- Written in Python using Flask microframework http://flask.pocoo.org
- Uses MySQL database
- Listens to REST calls compliant with documentation
- May run standalone or as a WSGI application (e.g. Apache)
Tasks are extracted from the queue
Each command specifies Executor Interface (EI) and action
Executor is dynamically instantiated to perform action on a DCI
Backend

- Available on GitHub: https://github.com/indigo-dc/APIServerDaemon
  
  - Java web application running on top of Apache Tomcat
  
  - Polls over the queue table in DB
  
  - Interacts with DCIs
  
  - Checks for consistency and resubmits failed tasks
Climate Modeling
Introduction

- Complex processes
- Different time and spatial scales
- Largely interdisciplinary
- Inherently non-linear
- Huge computational resources and huge data volumes
INDIGO DataCloud

- Development of open source data and computing platform targeted at scientific communities
- Deployable on various software stacks provisioned over public, private or hybrid clouds

INDIGO DataCloud is a complete solution with components for:

- AuthN / AuthZ
- Computing and storage resources management
- Dynamic instantiation and configuration of VMs and containers
- Ranking & selection of providers
- End-user interfaces
Solution Design

FutureGateway in PSNC orchestrates whole process
Computing nodes in CMCC, ORNL and LLNL
Each node gets data to process
Once finished, an ensemble analysis is done
Demo