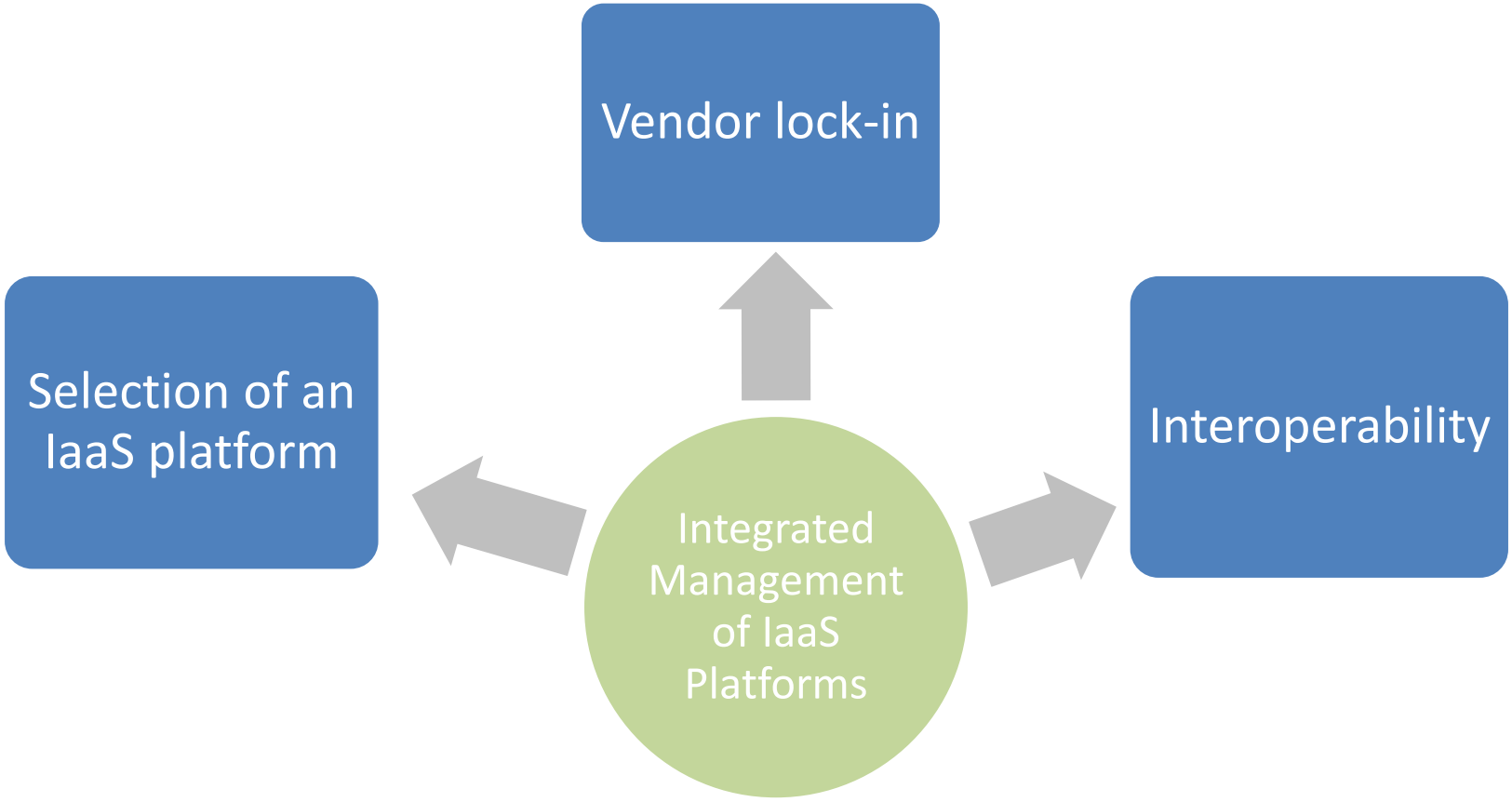
Integrated Management of IaaS Resources

Fernando Meireles (1050758@isep.ipp.pt) and Benedita Malheiro (mbm@isep.ipp.pt)
School of Engineering, Polytechnic Institute of Porto,
INESC TEC, Porto, Portugal

INESCTEC TECHNOLOGY & SCIENCE ASSOCIATE LABORATORY PORTUGAL
DEE Departamento de Engenharia Electrotécnica
isep Instituto Superior de Engenharia do Porto

Problem


Vendor lock-in
Selection of an IaaS platform
Interoperability
Integrated Management of IaaS Platforms

IaaS Platforms

INESCTEC TECHNOLOGY & SCIENCE | ASSOCIATE LABORATORY PORTUGAL

DEE Departamento de Engenharia Electrotécnica

isep Instituto Superior de Engenharia do Porto

IaaS Platform Comparison: Features

	IaaS Platforms			
Features	OpenNebula (v4.0)	OpenStack (Grizzly)	CloudStack (v4.2.1)	PACI (v5.3)
Authorization / Authentication	Password, SSH RSA keypair, X509, LDAP	Password, In-memory Key-Value Store, LDAP, X509	Password, LDAP, SSH RSA keypair, X509	Password, LDAP
Hypervisors	XEN, KVM, VMware vSphere	KVM, LXC, UML, VMWare vSphere, Xen, PowerVM, Hyper-V	VMware vSphere, KVM, Citrix XenServer and Cloud Platform	Parallels hypervisor, KVM
Management	Centralized	Scattered	Centralized	Centralized
Interfaces	Native XML-RPC API, AWS EC2 and EBS, OCCI, OCA	Native RESTful API, AWS EC2, S3 and EBS, OCCI	Native Query API, AWS EC2, Plug-in API (Java)	RESTful API
Network	Virtual router	Nova-network, Neutron	Virtual router	POA
Storage	Volumes	Volumes and Objects	Volumes	POA
Governance Model	Benevolent Dictator	Foundation	Technical Meritocracy	Proprietary

IaaS Platform Comparison: API Operations

		OpenNebula (XML – RPC API)	OpenStack (RESTful API)	CloudStack (Query API)	PACI (RESTful API)
OPERATIONS		**Instances**			
	VM Management	✓	✓	✓	✓
		Images			
	Image Management	✓	✓	✓	✓
		Storage			
	Snapshots Management	✓	✓	✓	-
	Volumes Management	✓	✓	✓	-
	Objects Management	-	✓	-	-
		Virtual Networks			
	Networks Management	✓	✓	✓	-
	Firewall, NAT and VPN Management	-	-✓	✓	±
		Other Functionalities			
	Group Management	✓	✓	✓	-
	Project and Zone Management	-	✓	✓	-
	Load Balancing Management	-	-✓	✓	✓
	VPC Management	-	-	✓	-

(- ✓) operations are available through third-party extensions to the core API

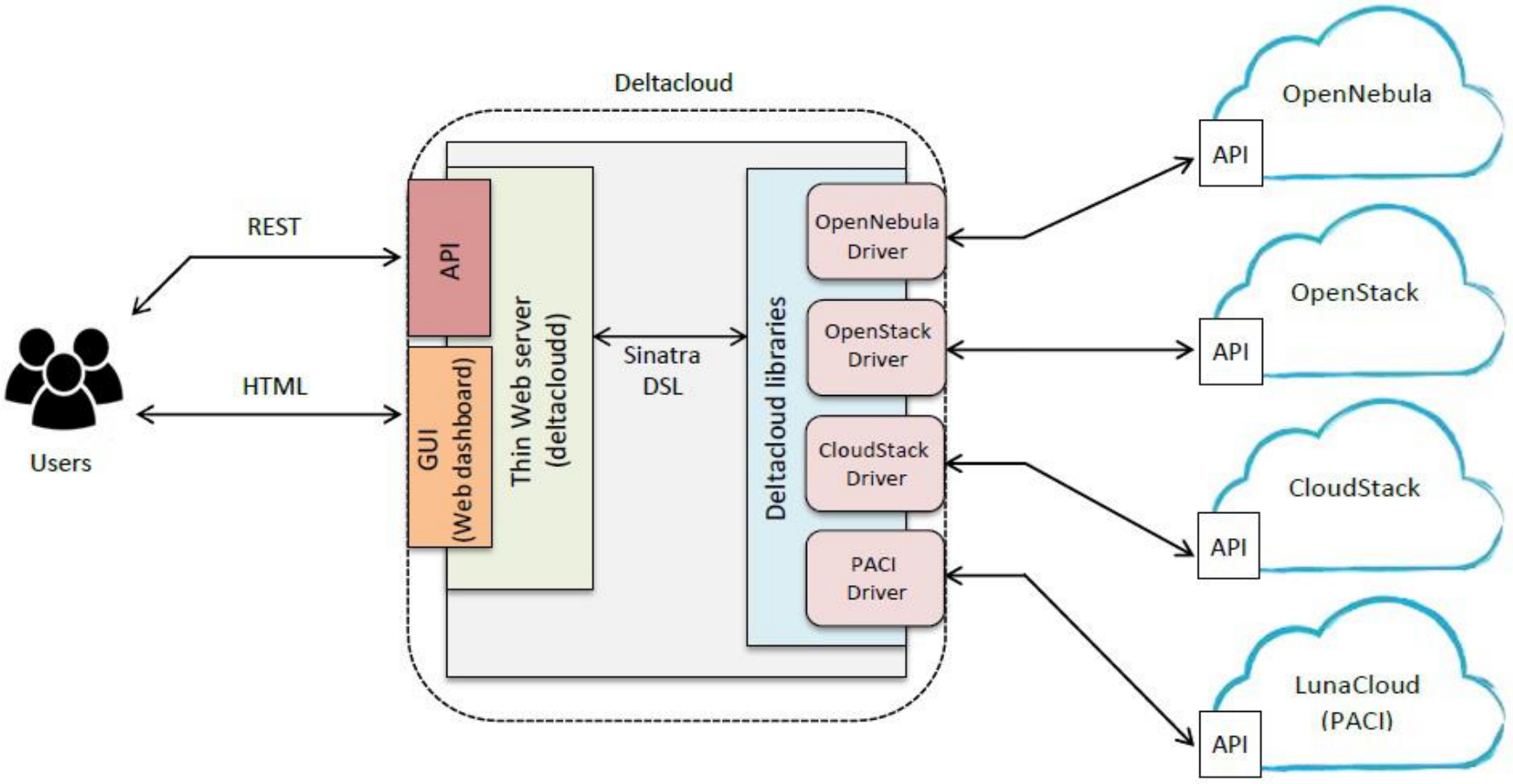
Interoperable Solution: Abstraction Frameworks/Libraries

Abstraction Solutions		
Deltacloud Framework	jClouds Library	Libcloud Library
Ruby 17 IaaS platforms Operations: • Compute • Storage • Network Drivers API: • Deltacloud REST • DMTF CIMI REST • AWS EC2 Query Web dashboard	Java 30 IaaS platforms Operations: • Compute • Storage Maven dependencies	Python 38 IaaS platforms Operations: • Compute • Storage • Network Drivers

Interoperable Service: Proposal and Implementation

- **Deltacloud framework**
  - Sinatra: Web application Ruby-based DSL
  - Thin: HTTP Ruby Web server
  - Deltacloud daemon (deltacloudd)
- **Interoperable Service API**
  - Instances, images, hardware profiles, load balancers, realms, objects, volumes, snapshots, metrics, addresses
- **Interoperable Service GUI**
  - Web dashboard
- **Driver Modules**
  - OpenNebula (included)
  - OpenStack (included)
  - CloudStack (third-party)
  - PACI (**developed**)

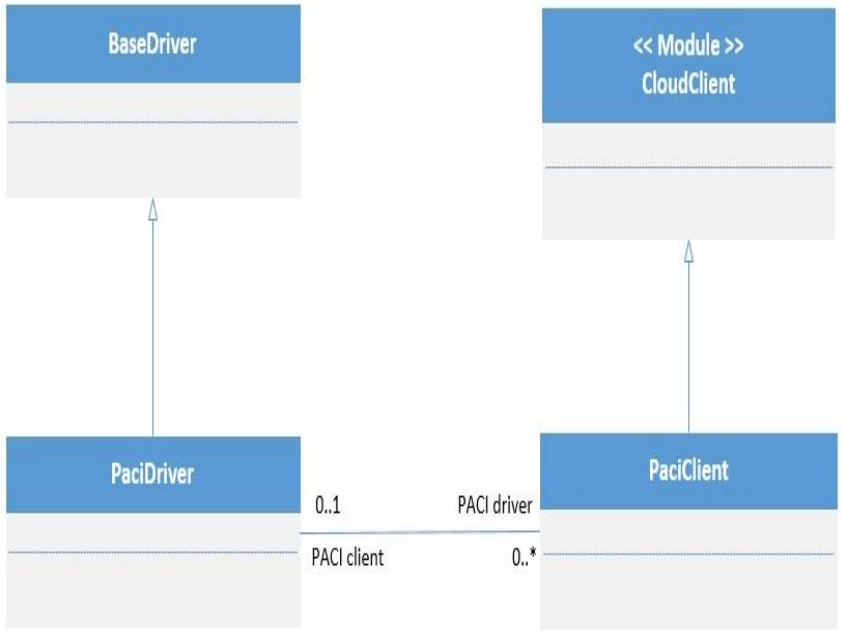
Interoperable Service: Proposal and Implementation



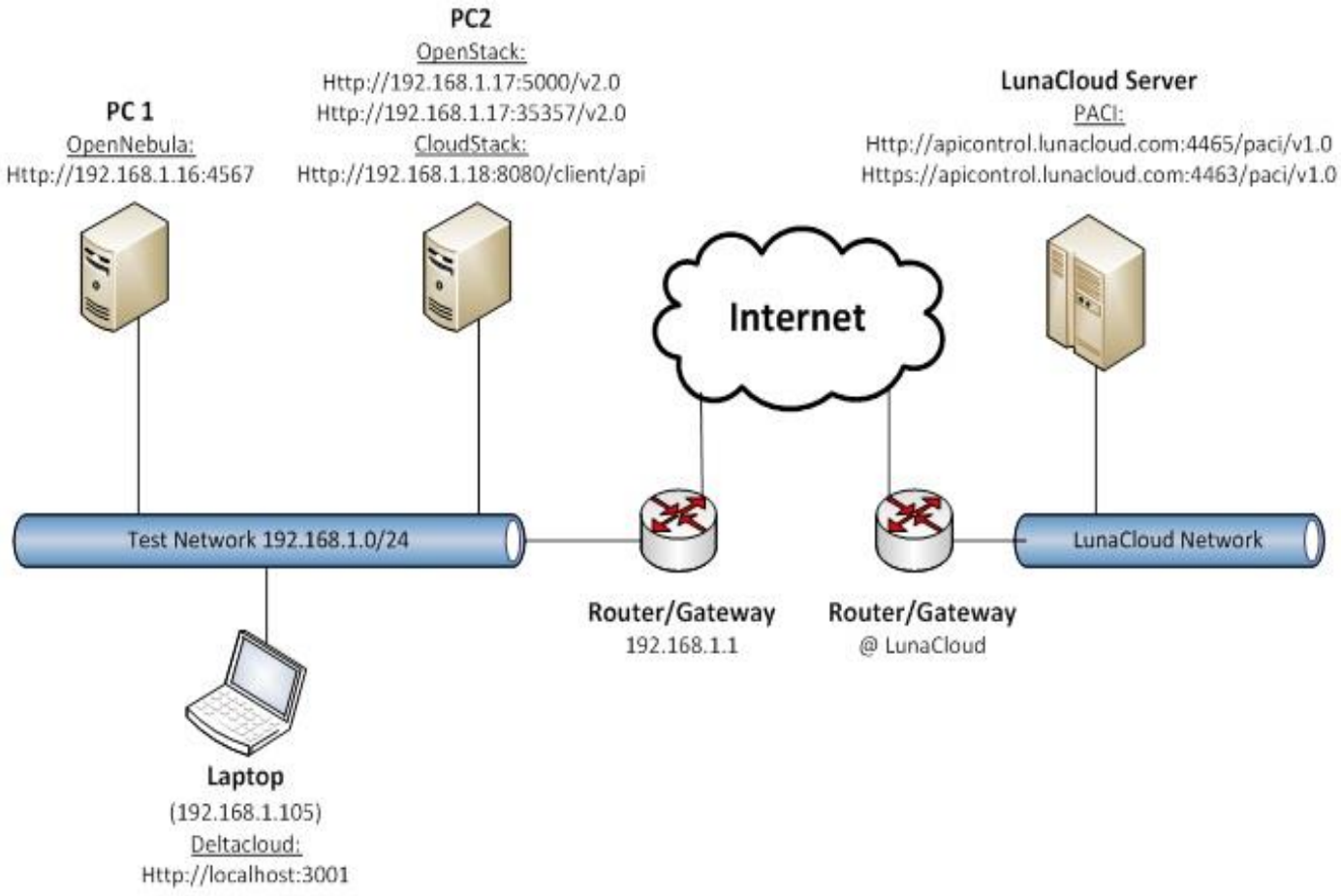
Interoperable Service Implementation: PACI Driver

- Software driver module:
  - Cloud client
  - PACI client
  - PACI Deltacloud driver

- Implemented collections:
  - Realms
  - Hardware Profiles
  - Instances
  - Images
  - Load Balancers



Tests and Results: Test Bed



Tests and Results: Problems

OpenNebula Driver Module

- Driver: Id argument mismatch in the `destroy_image` operation
- OCCI Client: instantiation of an unused argument in the `delete` method

OpenStack Driver Module

- Non-functional `start` and `stop` VM operations
- Different VM life-cycle implementation
- `stop_instance` operation defined as an alias of the `delete_instance` operation

CloudStack Driver Module

- Incomplete implementation
- Non-functional driver

Tests and Results: Discussion

Higher Time Response Delay

- Abstraction process
- Multiple HTTP calls to the back-end IaaS platform
- Implemented code

Smaller HTTP Request Packet Length

- Less request parameters
- Lower detail operations

(Considerably) Higher HTTP Response Payload

- Lists of resources (List instances, images, load balancers)
- Same granularity as specific resources information

High Availability Deployment *vs* Proof of Concept Deployment

- PACI direct API operations with the smallest time response delay

Conclusions

- **IaaS platforms:**
  - Similar features | different approaches (open source platforms)
  - Different API implementations and API operations
  - Lack of interoperability (PACI)
- **IaaS interoperability solutions:**
  - Reuse of abstraction frameworks/libraries: Deltacloud framework
- **Interoperable service:**
  - OpenStack driver requires improvement
  - Non-functional CloudStack driver
  - Large time response delay *versus* direct API calls
  - Large HTTP response payload (list operations) *versus* direct API calls

Conclusions

- **Future developments:**
  - Refine PACI driver code
  - Addition of new Deltacloud collections
  - Development of a new improved GUI
  - Resource migration service
  - CloudStack driver development

Questions?