

Enabling Cloud interoperability with COMPSs

Daniele Lezzi, Francesc Lordan, Roger Rafanell, Rosa M. Badia
Barcelona Supercomputing Center, Spain

Fabrizio Marozzo, Domenico Talia
University of Calabria, Italy

Cloud Futures 2012, Berkeley, CA - May 7-8, 2012



Outline

- The COMPSs framework
- The VENUS-C platform
- COMPSs and Azure
- Application evaluation

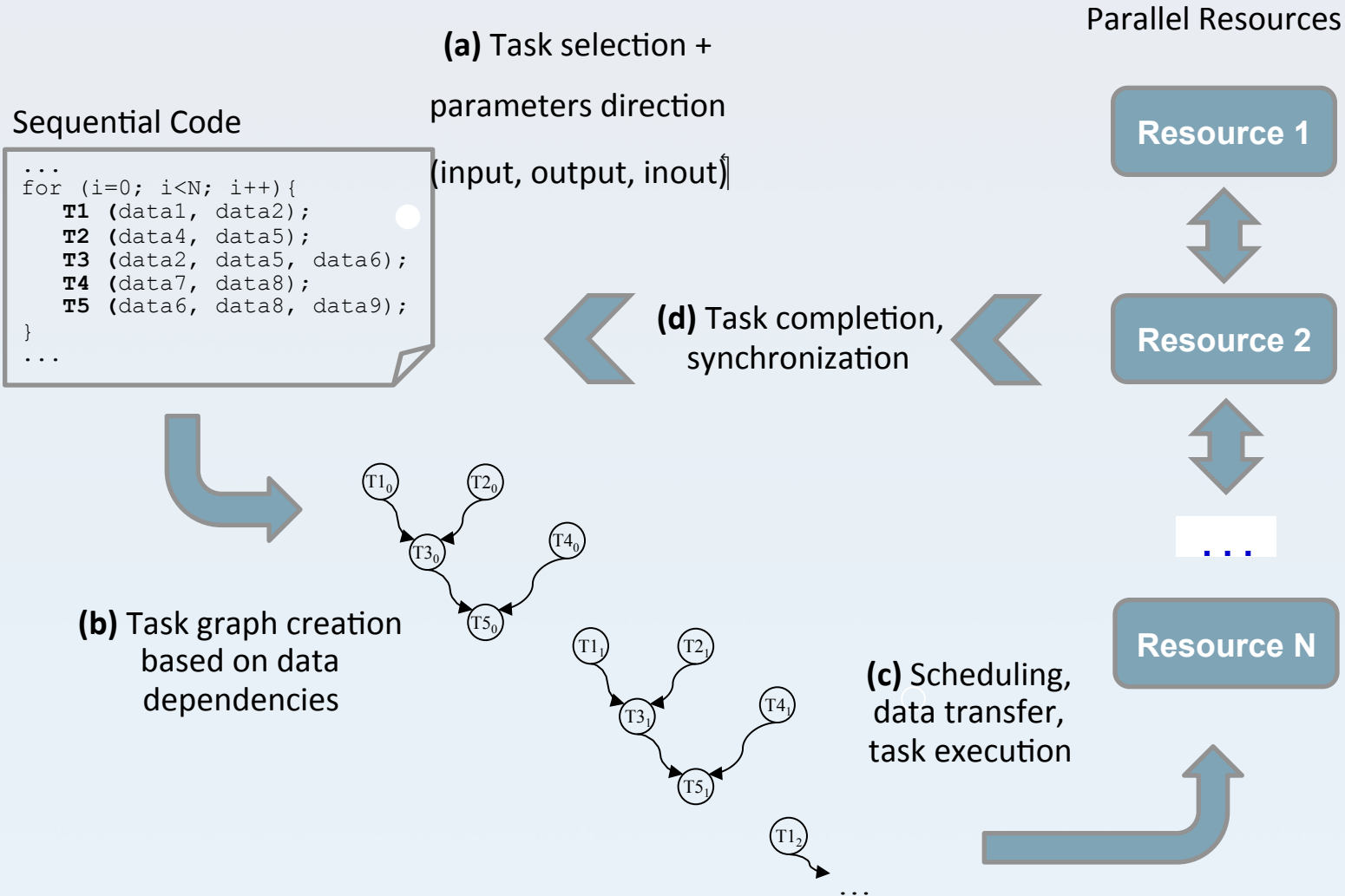


Goal of This Work

- ❑ **COMP Superscalar (COMPSs)** is a programming framework that provides a programming model and a runtime that ease the development of applications for distributed environments and their execution on a wide range of computational infrastructures.
 - COMPSs has been recently extended in order to be interoperable with several cloud technologies like Amazon, OpenNebula, Emotive and other OCCI compliant offerings.

- ❑ **Goal:** This work presents the extensions of this interoperability layer to support the execution of COMPSs applications into the **Windows Azure Platform**.

COMPSs Programming model



COMPSs Programming model

User code

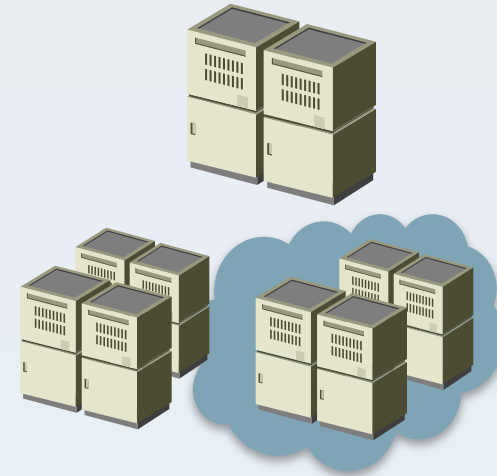
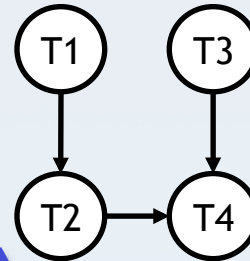
```

initialize(f1);
for (int i = 0; i < 2; i++) {
    genRandom(f2);
    add(f1, f2);
}
print(f2);
    
```

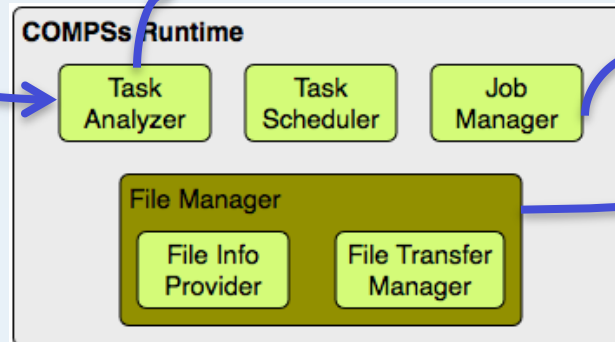
Annotated interface

Custom Loader

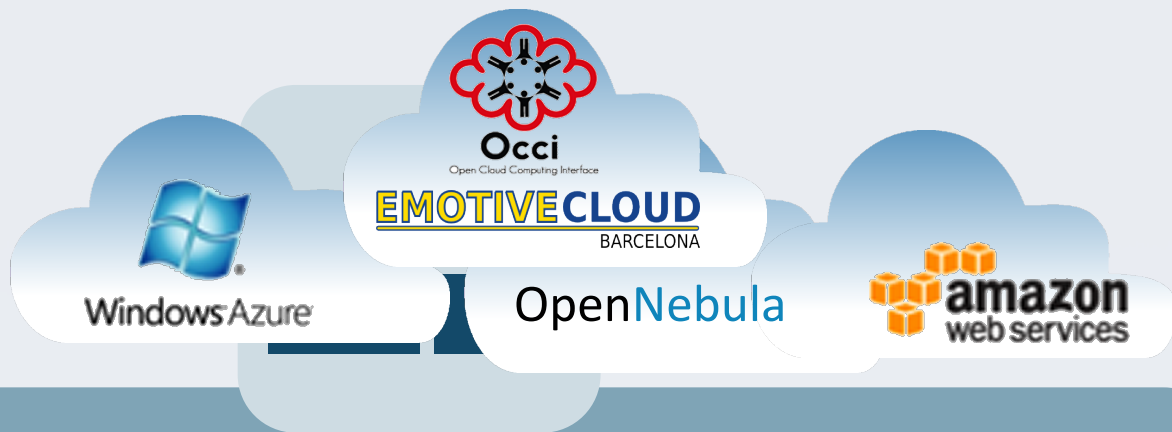
Javassist



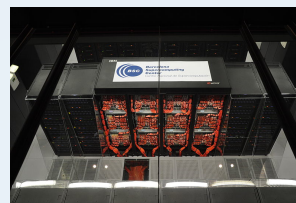
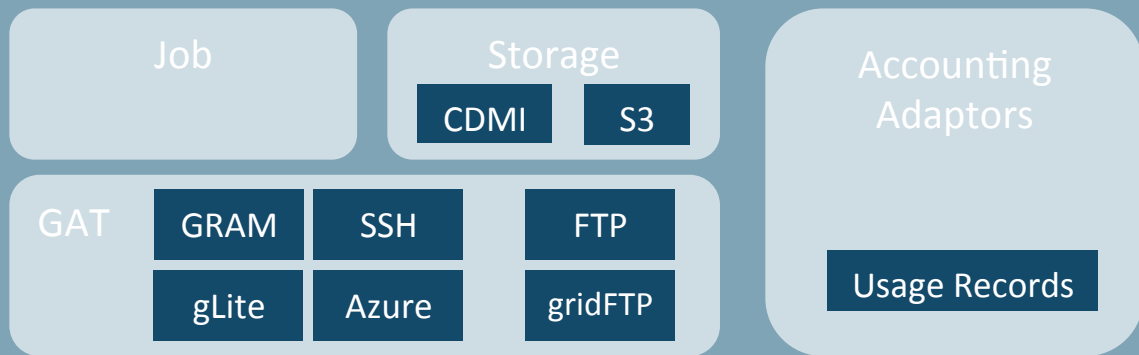
**Grids
Clusters
Clouds**

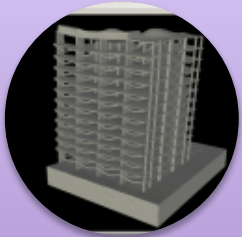


Files



COMPSs Runtime





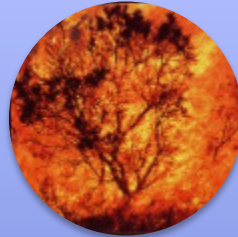
Structural
Analysis for
Civil
Engineering



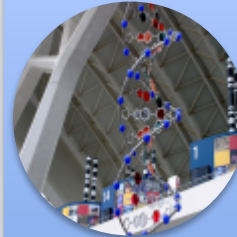
Building
Information
Management



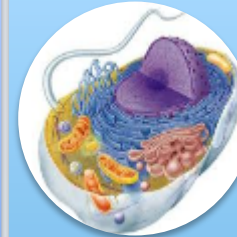
Data for
Science -
AquaMaps



Civil
Protection
and
Emergencies



Bioinformatic
s



System
Biology

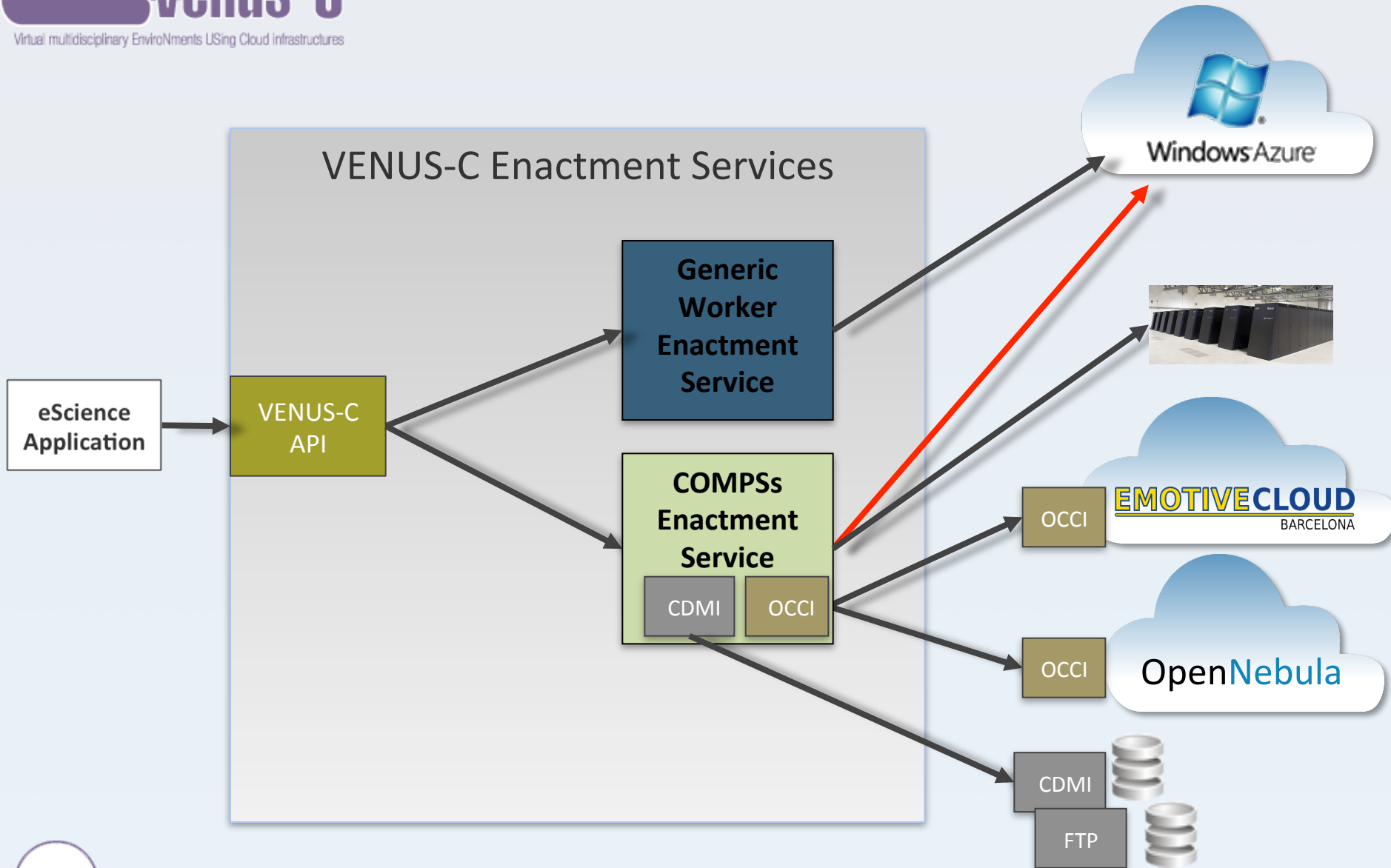


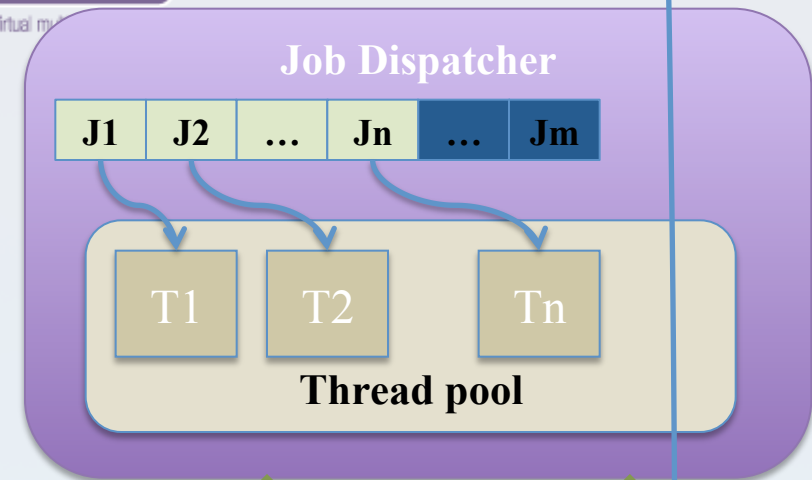
Drug
Discovery

COMPSS in VENUS-C allows the easy porting of scientific applications to Clouds

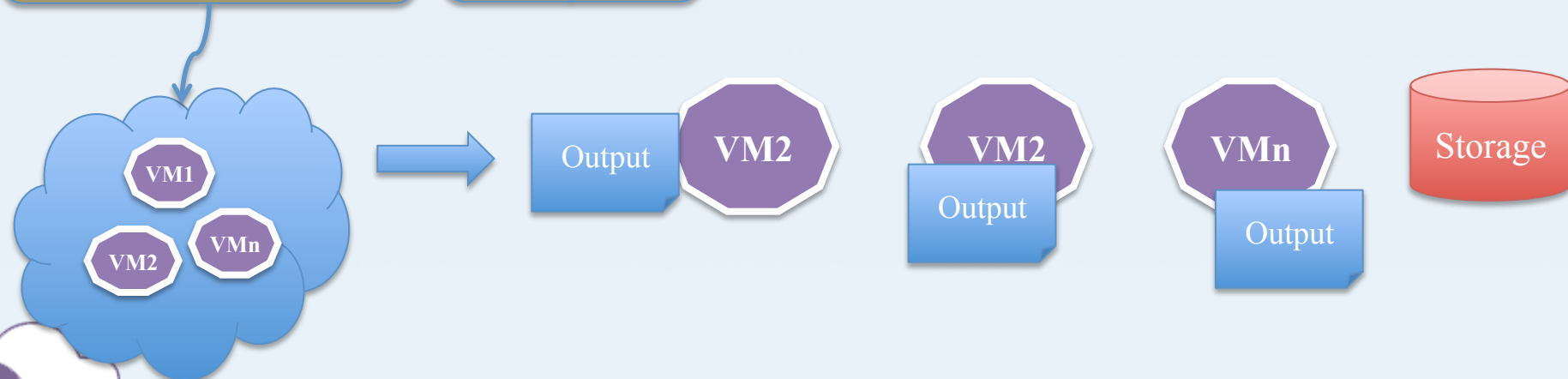
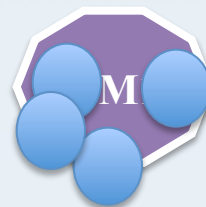
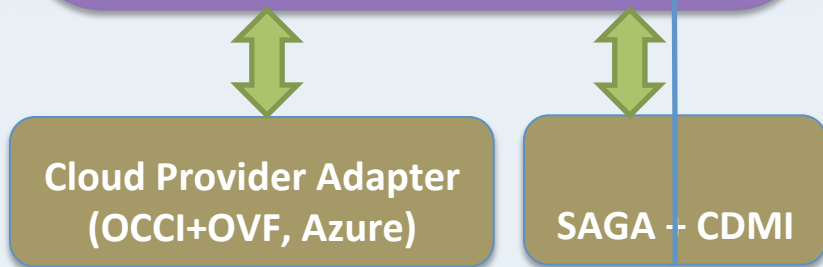
- Minimum effort required to the user
- No need of change for an already existing COMPSS application
- Provides elasticity features
- Execution results demonstrate scalability

The VENUS-C Platform

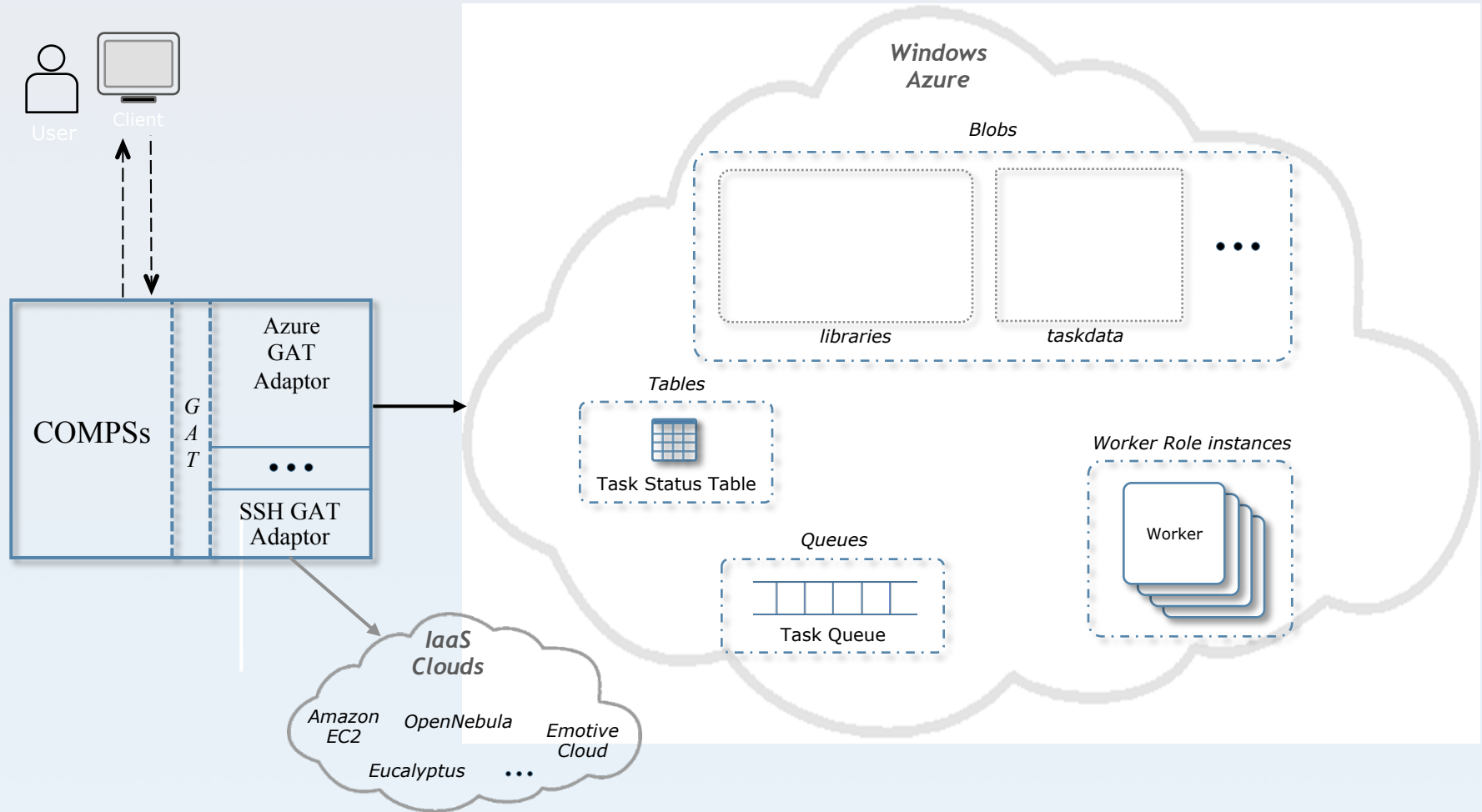




1. VMs are created on demand through the actual cloud adaptor
2. The COMPSs application is remotely executed through SAGA and the data is staged in
3. The tasks are executed on the worker nodes



Azure Java GAT Adaptor

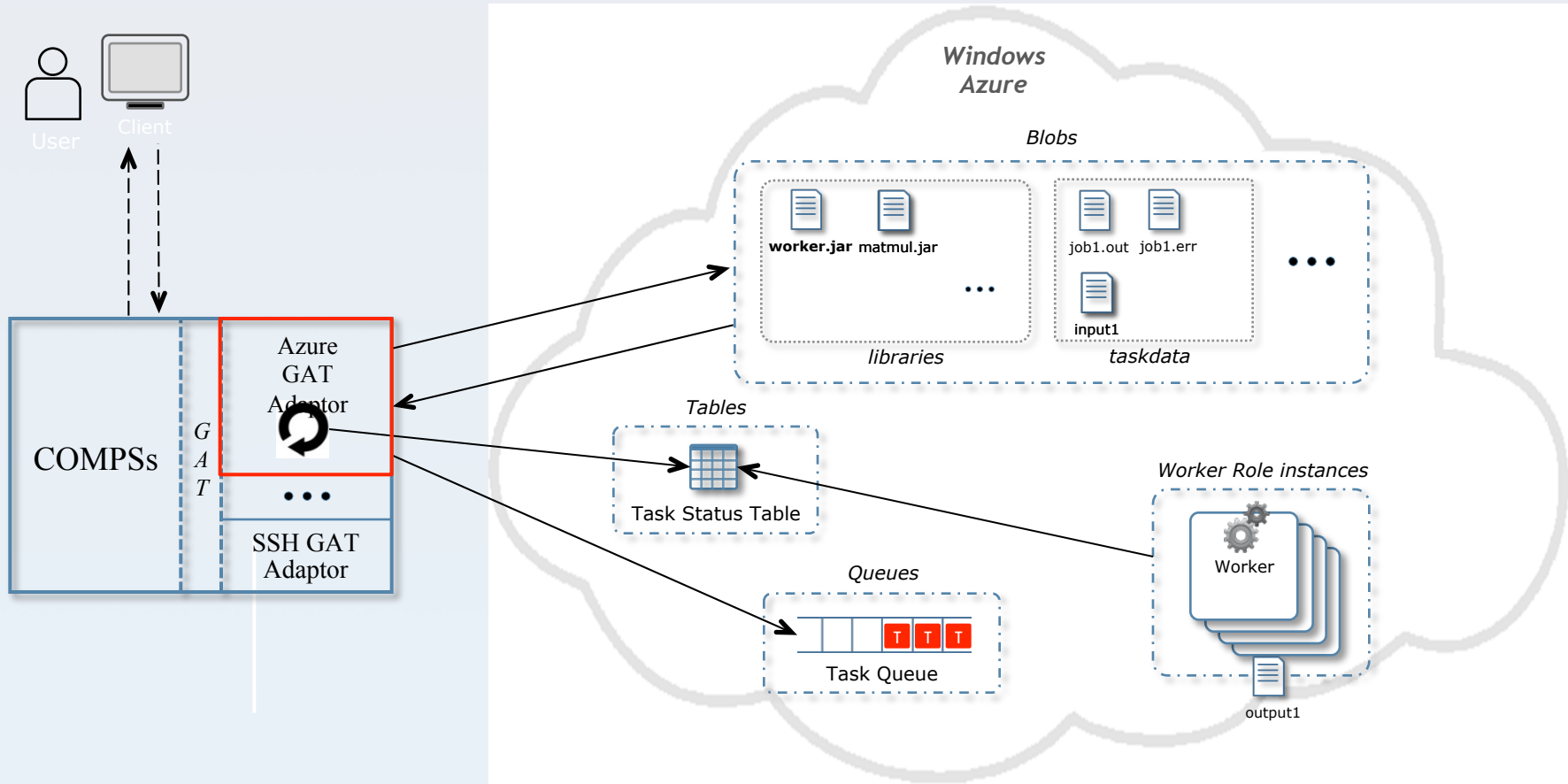


- ❑ The **Azure JavaGAT Adaptor** enriches COMPSs with data management and execution capabilities that make it interoperable with Azure and implemented using two subcomponents.
- ❑ Data management is supported by a subcomponent called **Azure File Adaptor**.

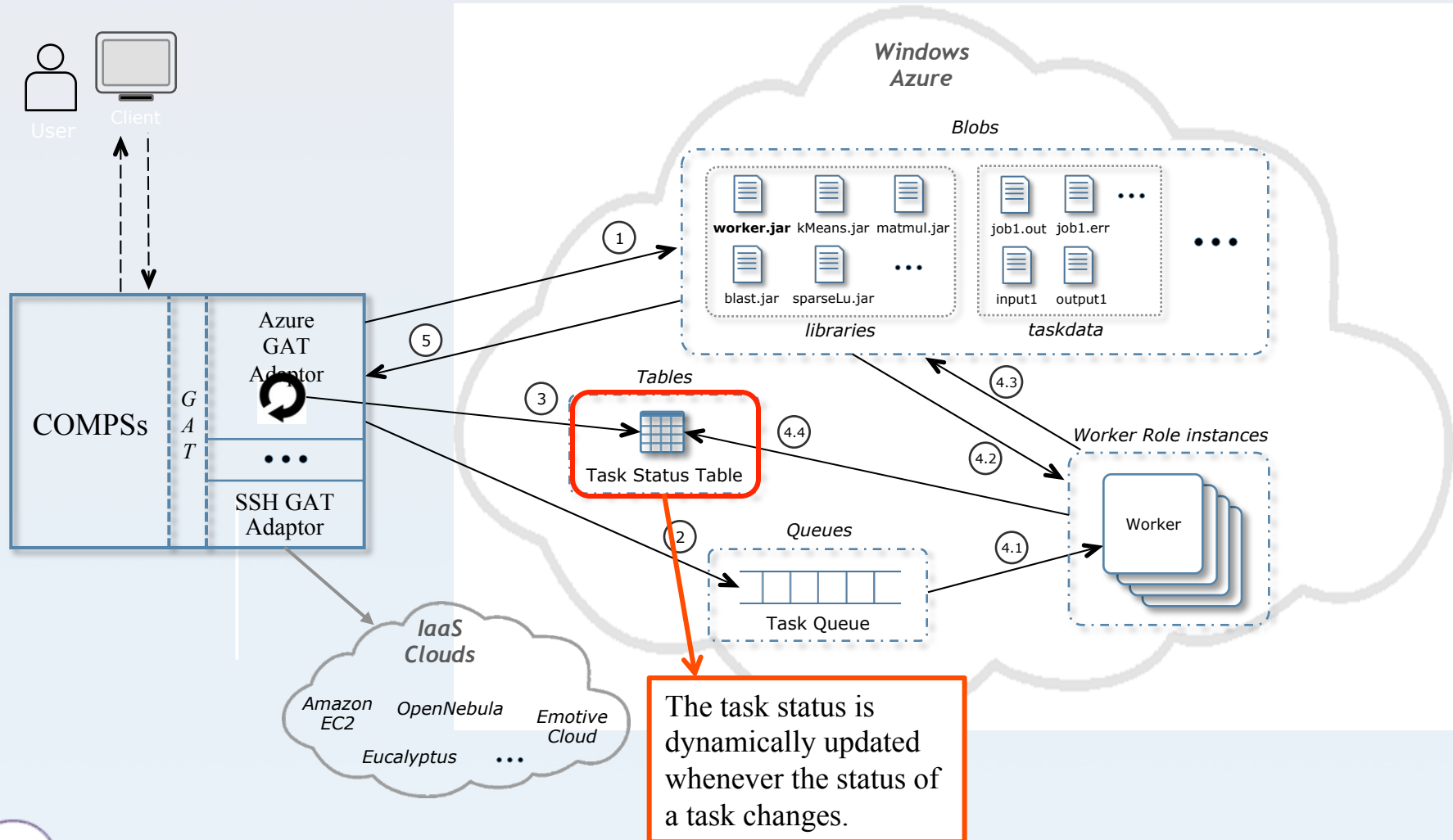
Read and write data on the Azure Blob Storage (Blobs), to deploy the libraries needed to execute on Azure and to store the input and output data (taskdata) for the tasks.
- ❑ The **Azure Resource Broker Adaptor**, on the other side, is responsible for the task submission.

Following the Azure Work Queue pattern, this subcomponent adds into a Task Queue the tasks that must be executed on an Azure resource by a Worker. In order to keep the runtime informed about each task execution, the status of the tasks is updated in a Task Status Table.

Azure GAT Adaptor: Exec. mechanisms



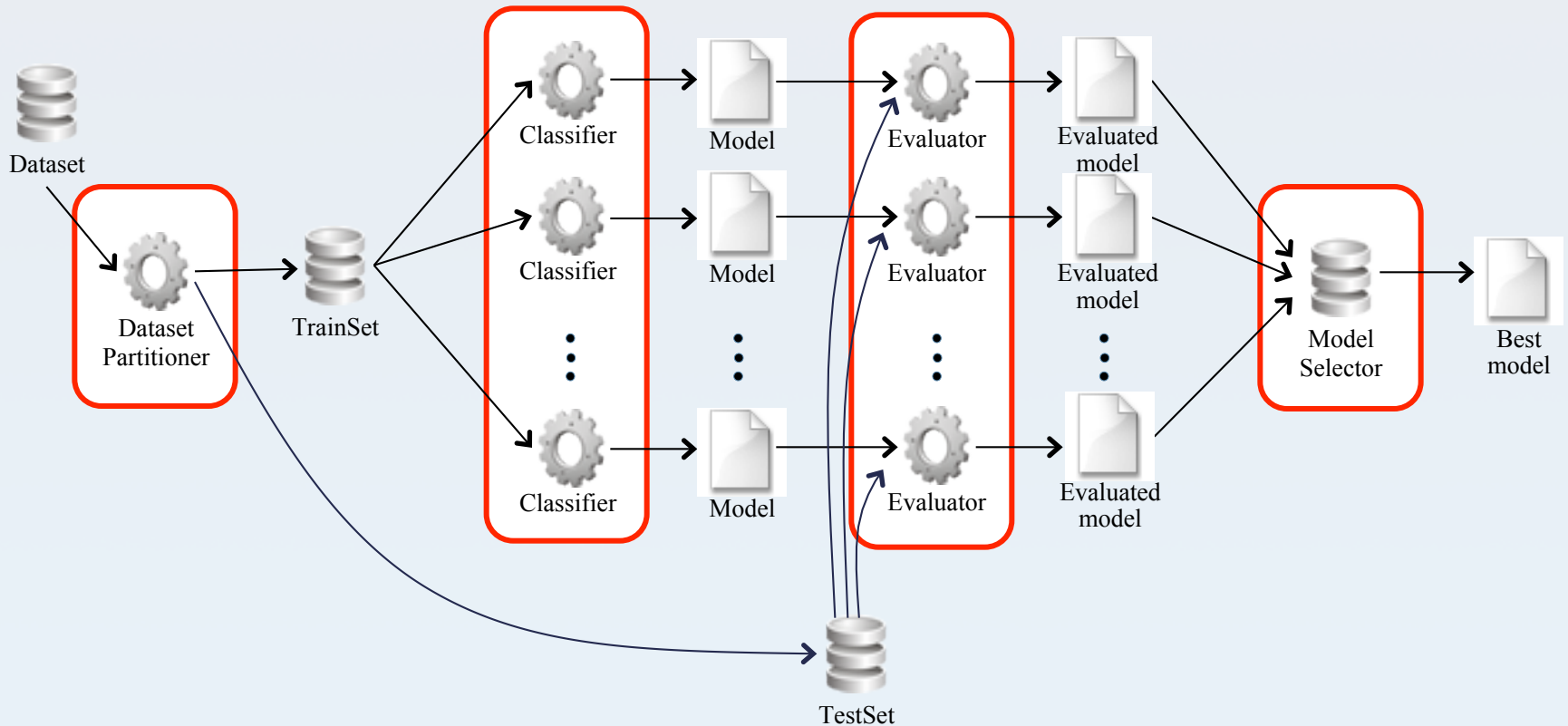
Azure GAT Adaptor: Architecture

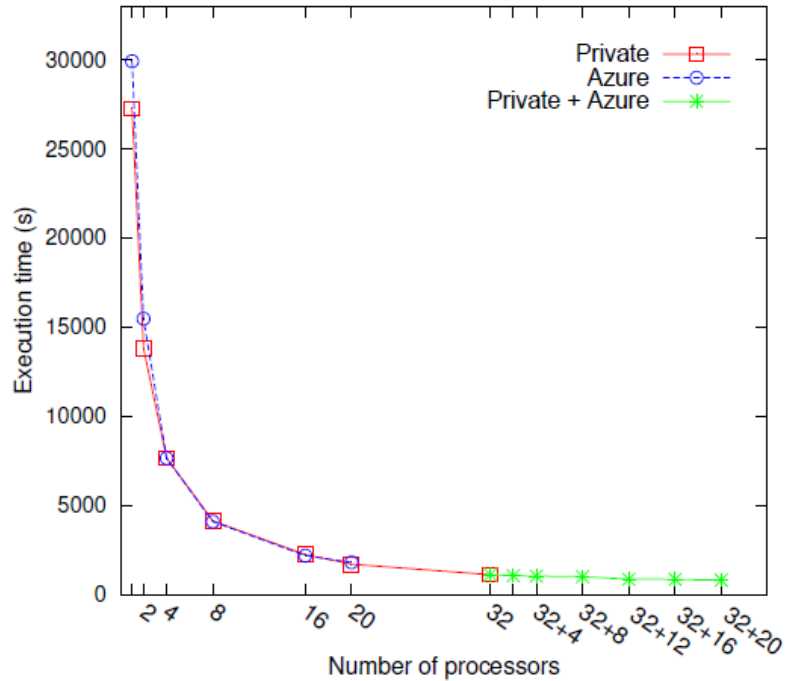


Evaluation: Goals

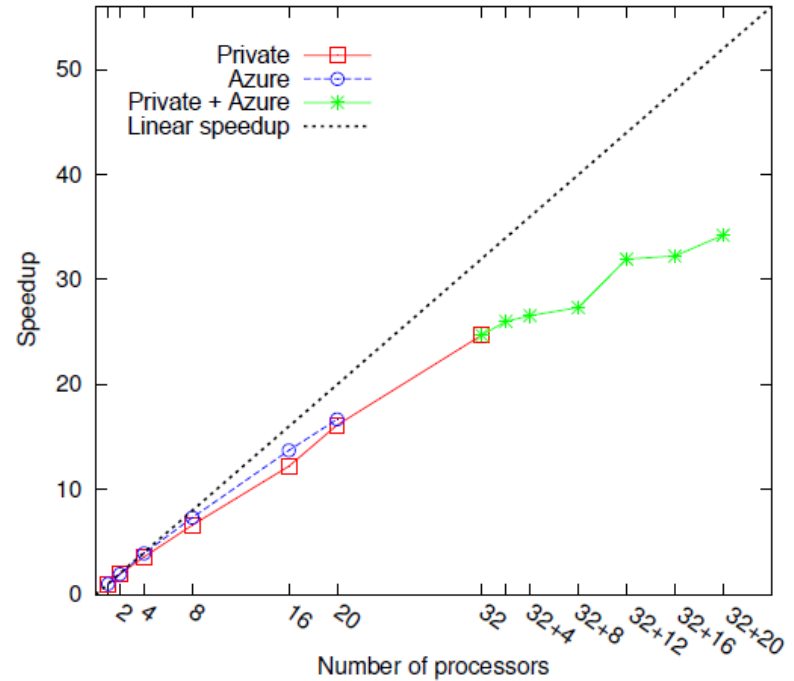
- ❑ Evaluating the performance of Azure GAT Adaptor through the execution of a classier-based workflow application on a pool of virtual servers hosted by Microsoft Cloud datacenters.
 - Scalability that can be achieved through the parallel execution on a pool of virtual servers.
- ❑ Evaluating the performance of the workflow application in a hybrid scenario
 - A private cloud environment managed by Emotive Cloud middleware
 - A public cloud testbed made of Azure instances
 - An hybrid configuration using both private and public clouds

Evaluation Workflow application





(a)



(b)

Fig. 4. Execution time and speedup values depending on the number of processors.

Conclusions

- ❑ COMPSs is platform unaware programming model that simplifies the development of applications in distributed environments
 - Transparent data managemet, task execution
 - Parallelization at task level
 - Independent of platform: clusters, grids, clouds
- ❑ The Azure adaptor extends COMPSs interoperability
- ❑ The proposed approach has been validated through the execution of a data mining workflow ported to COMPSs and executed on a hybrid testbed composed of a private cloud in VENUS-C
- ❑ Future work includes the support of the **dynamic resource provisioning** in Azure and enhancements to the Azure JavaGAT adaptor to optimize data transfers among different clouds, and the possibility to specify input files already available on the Azure storage.

Contacts & acknowledgments

Daniele Lezzi
Roger Rafanell
Francesc Lordan
Rosa M. Badia

@



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

www.bsc.es/compss

www.venus-c.eu

Fabrizio Marozzo
Domenico Talia

@



ICAR-CNR, University of Calabria