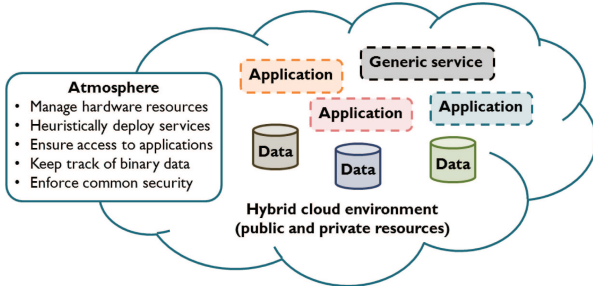


The VPH-Share project aims to provide members of the pan-European Virtual Physiological Human Network of Excellence with access to a set of scalable, computationally- and data-intensive applications, enabling further progress in the fields of neurobiology, virology and molecular dynamics.

- End user support**
Easy access to applications and binary data
- Developer support**
Tools for deploying applications and registering datasets
- Admin support**
Management of VPH-Share hardware resources



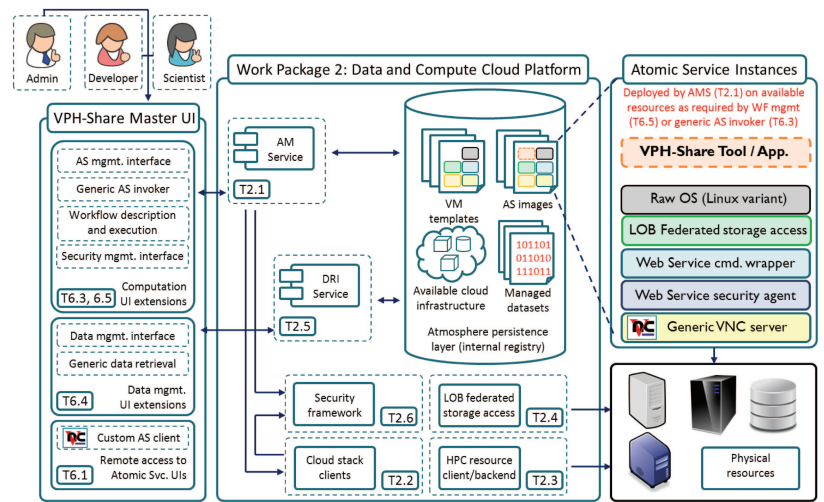
Features of the VPH-Share Cloud Platform

The goal of the Atmosphere Cloud Computing Platform is to manage cloud/HPC resources in support of VPH-Share applications by:

- Providing a mechanism for application developers to install their applications/tools/services on the available resources
- Providing a mechanism for end users (domain scientists) to execute workflows and/or standalone applications on the available resources with minimum fuss
- Providing a mechanism for end users (domain scientists) to securely manage their binary data in a hybrid cloud environment
- Providing administrative tools facilitating configuration and monitoring of the platform

The Architecture of Atmosphere

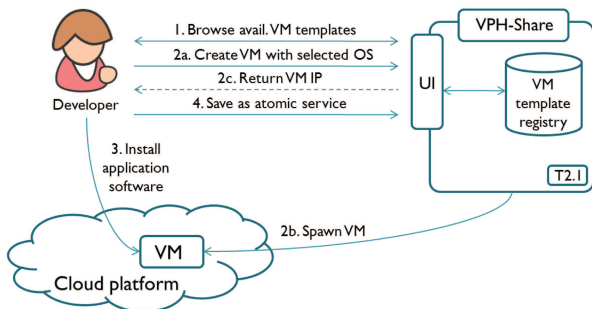
- Each VPH-Share application (or application component) is represented by an Atomic Service
- Each Atomic Service is installed on a Virtual Machine managed by Atmosphere
- Atomic Services can be instantiated and replicated on demand
- Access to application functionality is facilitated by a common set of Web Service interfaces
- Each Virtual Machine comes with a selection of tools enabling automatic integration with the VPH-Share architecture
- Virtual Machines (and by extension – the Atomic Services which operate on them) can access federated cloud storage to store and read sensitive data
- The platform is protected by a common security system where all calls to VPH-Share Atomic Services are authenticated and authorized by a role-based security component



Use case 1 - developer

Installing a VPH-Share application in the Cloud (developer action):

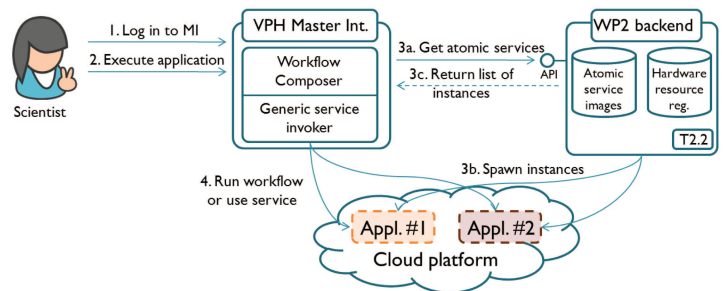
- Upon the application developers' request, the Atmosphere component spawns a fresh Virtual Machine, which resides in the cloud and contains all the features typically expected of an „out of the box” operating system (virtualized storage, standard libraries, root account, initial configuration etc.) If needed, many such VMs can be spawned, each encapsulating a single VPH-Share Atomic Service.
- It is the application developers' task to install components of their applications on these templates so that they can be wrapped as atomic services.
- Atmosphere can then further manage the atomic services and deploy their instances on cloud resources as requested by the user.



Use case 2 - end user

Preparing a VPH-Share application for execution (user action):

- The user requests execution of an application (using the VPH-Share Master Interface).
- An option is also provided to directly interface a specific atomic service (we call this the Generic Atomic Service Invoker)
- The Master UI (or Generic Invoker) informs Atmosphere which atomic services to deploy in the cloud so that the workflow may be executed.
- Atmosphere takes care of deploying the required assets and returns a list of service endpoints (typically IP addresses) whereupon workflow execution may commence.
- Atmosphere should „talk to” various computing stacks, both commercial and private – we are initially going to support OpenStack and Amazon EC2.



www.vph-share.eu

References

1. Tomasz Gubala, Marek Kasztelnik, Maciej Malawski, Marian Bubak, "Towards System-Level Science Support". In: Bubak, M., Albada, G.D.v., Dongarra, J., Sloot, P.M.A. (eds.), Proceedings ICCS 2008, Kraków, Poland, June 23-25, 2008, LNCS 5101, pp. 56-65, Springer 2008
2. Jeffery K. Neidecker-Lutz B, The Future of Cloud Computing, Expert Group Report published by the European Commission Information Society and Media Directorate General – available at <http://cordis.europa.eu/fp7/ict/ssai/docs/cloud-report-final.pdf> (accessed May 9, 2011)
3. M. Bubak et al., Virtual Laboratory for Collaborative Applications, In: M. Cannataro (Ed.) Handbook of Research on Computational Grid Technologies for Life Sciences, Biomedicine and Healthcare, Chapter 27, pp. 531-551, Information Science Reference, 2009, IGI Global
4. B. Balis, M. Kasztelnik, M. Bubak, T. Bartyński, T. Gubala, P. Nowakowski, and J. Broekhuijsen. The UrbanFlood Common Information Space for Early Warning Systems. Procedia Computer Science, 4:96-105, 2011. Proceedings of the International Conference on Computational Science, ICCS 2011