Empowering Cloud with latest VMDIRAC improvements

DIRAC User Workshop, Ferrara, 2015

Víctor Méndez Muñoz
vmendez@uab.es

Agenda

- A single context to drive them all
- Cloud flexibility: Hybrid and multi-layered
- Use Case: S3 1K genome processing
- Conclusions

A single context to drive them all

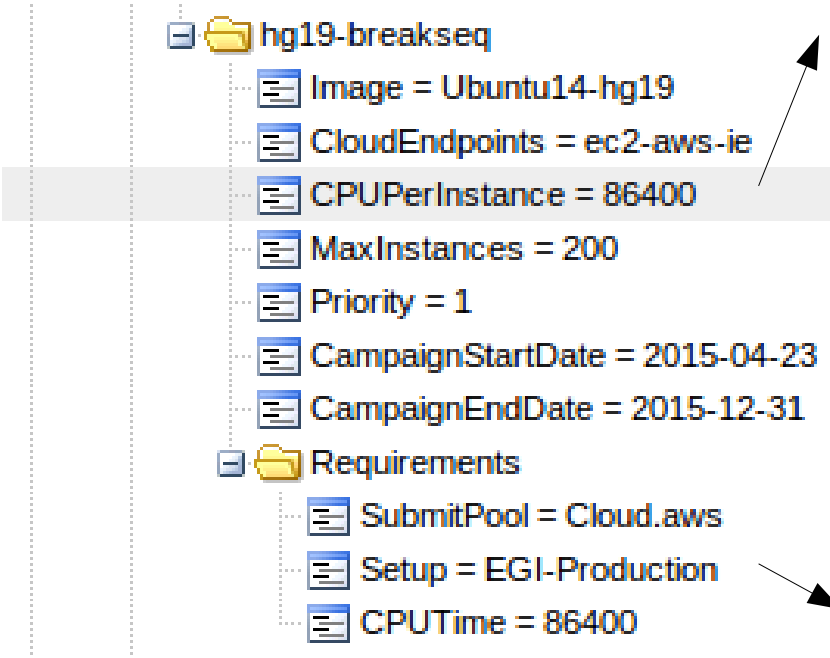
VMDIRAC v1r3 -April 2015- includes a single cloudinit method for

- OpenStack
- OpenNebula
- AWS

- A RunningPod will bind Job Requirements to be allocated in the Vms + A DIRAC Image + List of cloud endpoints
- DIRAC Image in the Configuration System, includes the Image contextualization
- Cloud endpoint should be setup in the DIRAC Configuration

A single context to drive them all

RunningPod setup:
Resources | VirtualMachines | RunningPods



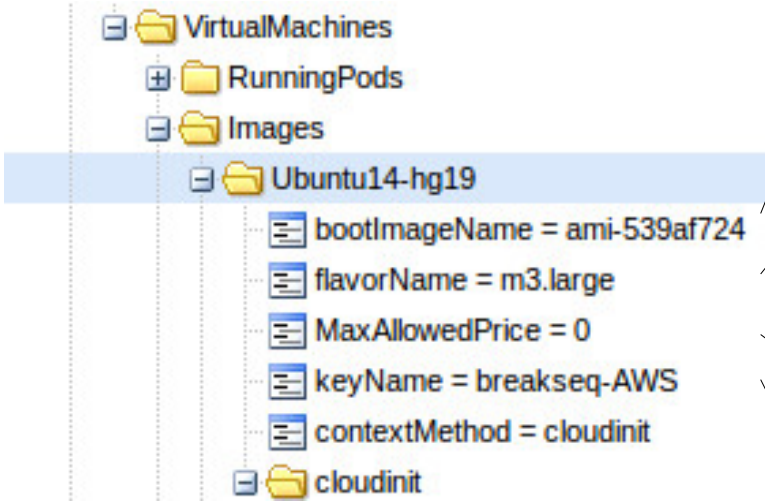
minimal overall CPU of the DIRAC jobs waiting in the task queued to submit a new VM, elasticity:

- 0 → no minimal CPU in tasks queue, best total wall time
- Average required CPU → compromise between VM efficiency and total wall time
- A very large value to maximize the efficiency in terms of VM creation overhead

Requirements you want to put in /opt/dirac/etc/dirac.cfg in the Vms
Will be matched with the jobs

A single context to drive them all

DIRAC Image setup
Resources | VirtualMachines | Images



Your golden image name in the IaaS provider

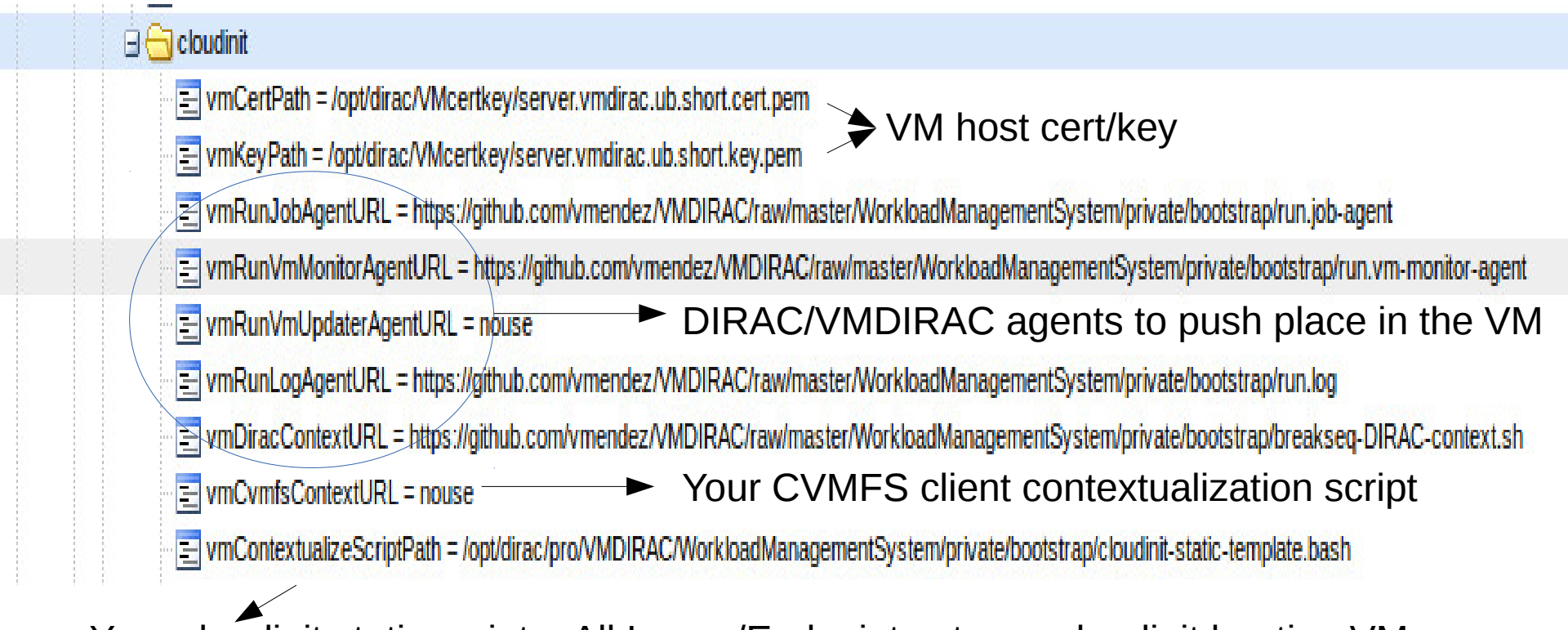
Flavor = Instance Type, depending on your IaaS provider

If MaxAllowedPrice = 0 or not definen then not spot instances are used

KeyName in your IaaS to push public key in the Vms (deploying / debuggin matters)

A single context to drive them all

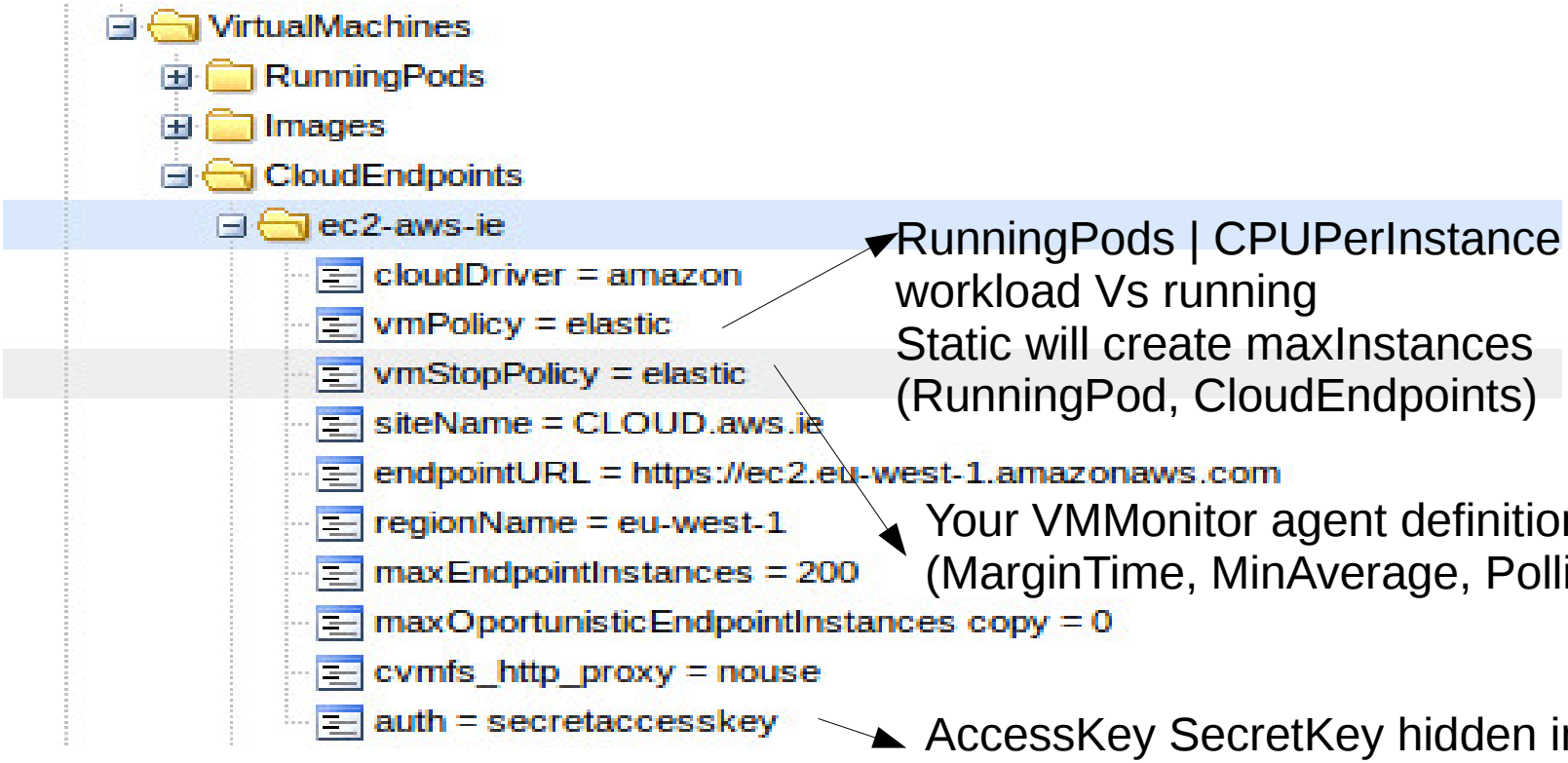
DIRAC Image setup: Single cloudinit contextualization
Resources | VirtualMachines | Images



Your cloudinit static script + All Image/Endpoint setup → cloudinit booting VM

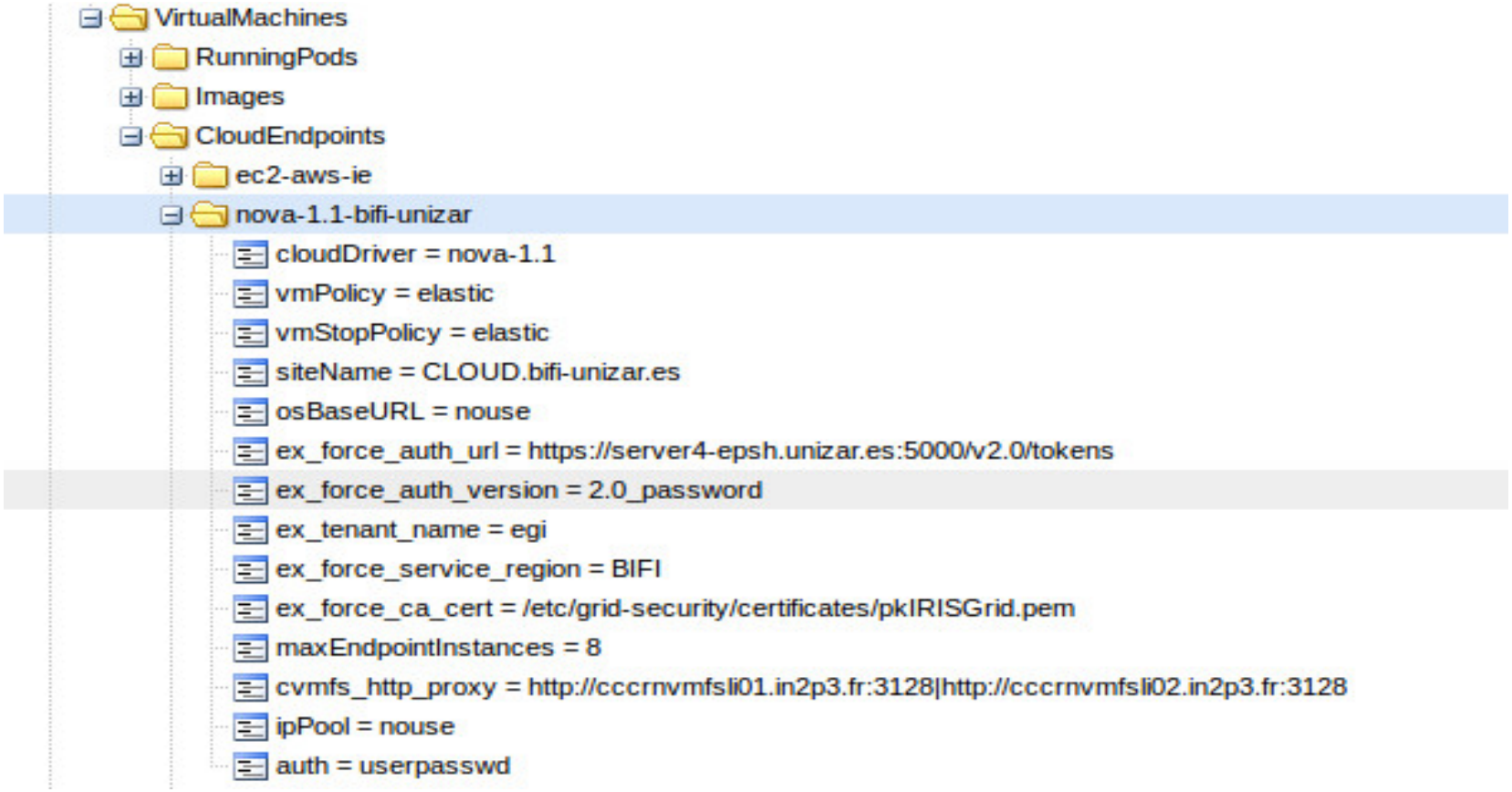
A single context to drive them all

Cloud endpoint setup: AWS
Resources | VirtualMachines | CloudEndpoints



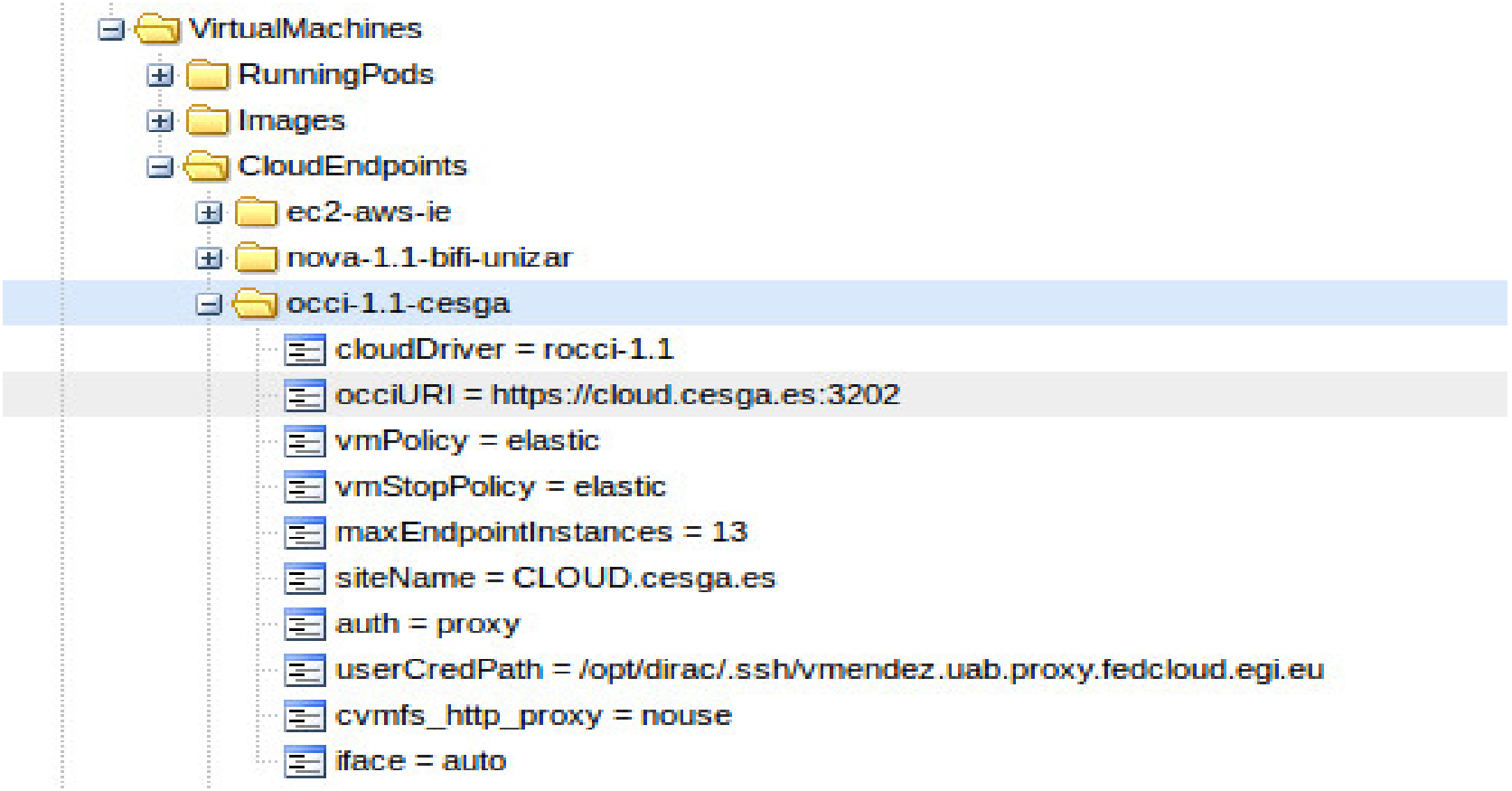
A single context to drive them all

Cloud endpoint setup: OpenStack
Resources | VirtualMachines | CloudEndpoints



A single context to drive them all

Cloud endpoint setup: OpenNebula
Resources | VirtualMachines | CloudEndpoints



Cloud flexibility: Hybrid and multi-layered

Hybrid Cloud

- Usual Categorization:
  - Public and Private nature of IaaS resources
  - Public is any Cloud owned by third party
  - Private Cloud is a user infrastructure

- Other Categorizations:
  - Community Cloud: Is owned and / or managed by a particular user comunity.
    - Public: like some of EGI Fedcloud resources owned by providers which are EGI members
    - Private
    - Hybrid: f.e. public AWS + private OpenStack

Cloud flexibility: Hybrid and multi-layered

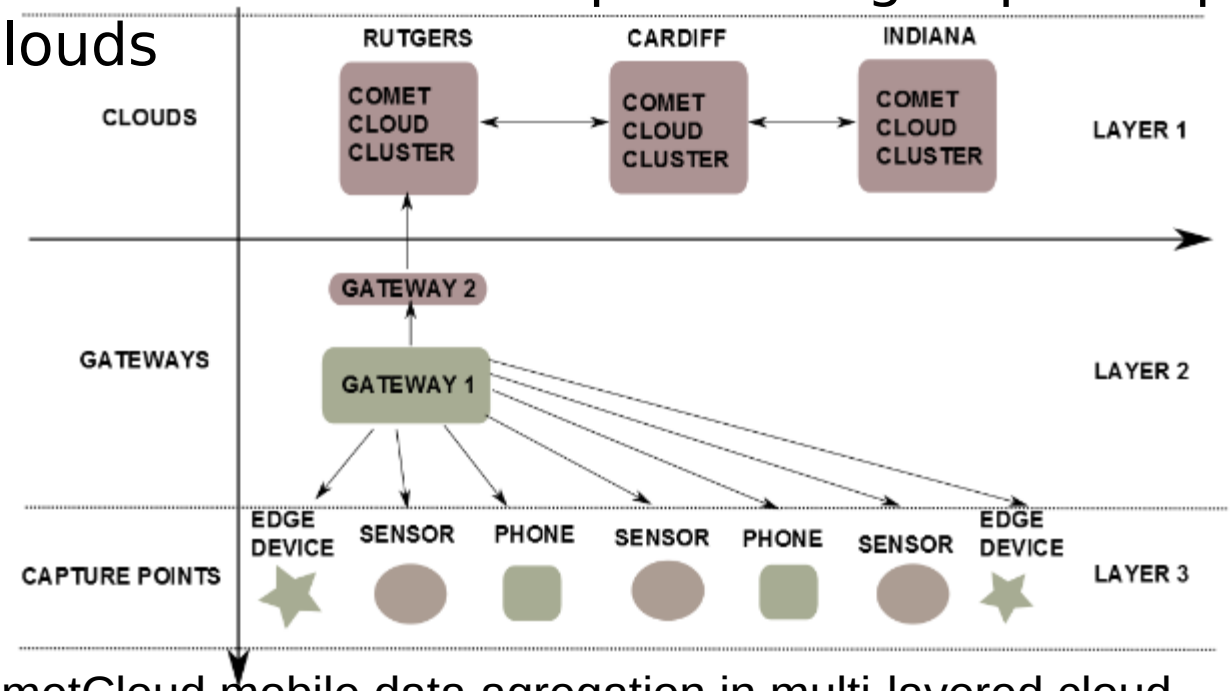
Multi-layered Cloud

- Taking advantage of pre-procesing / post-processing to filter data out of the main processing in public pay-per-use Clouds

Example:

Input data Filtered

Sensors: Input data



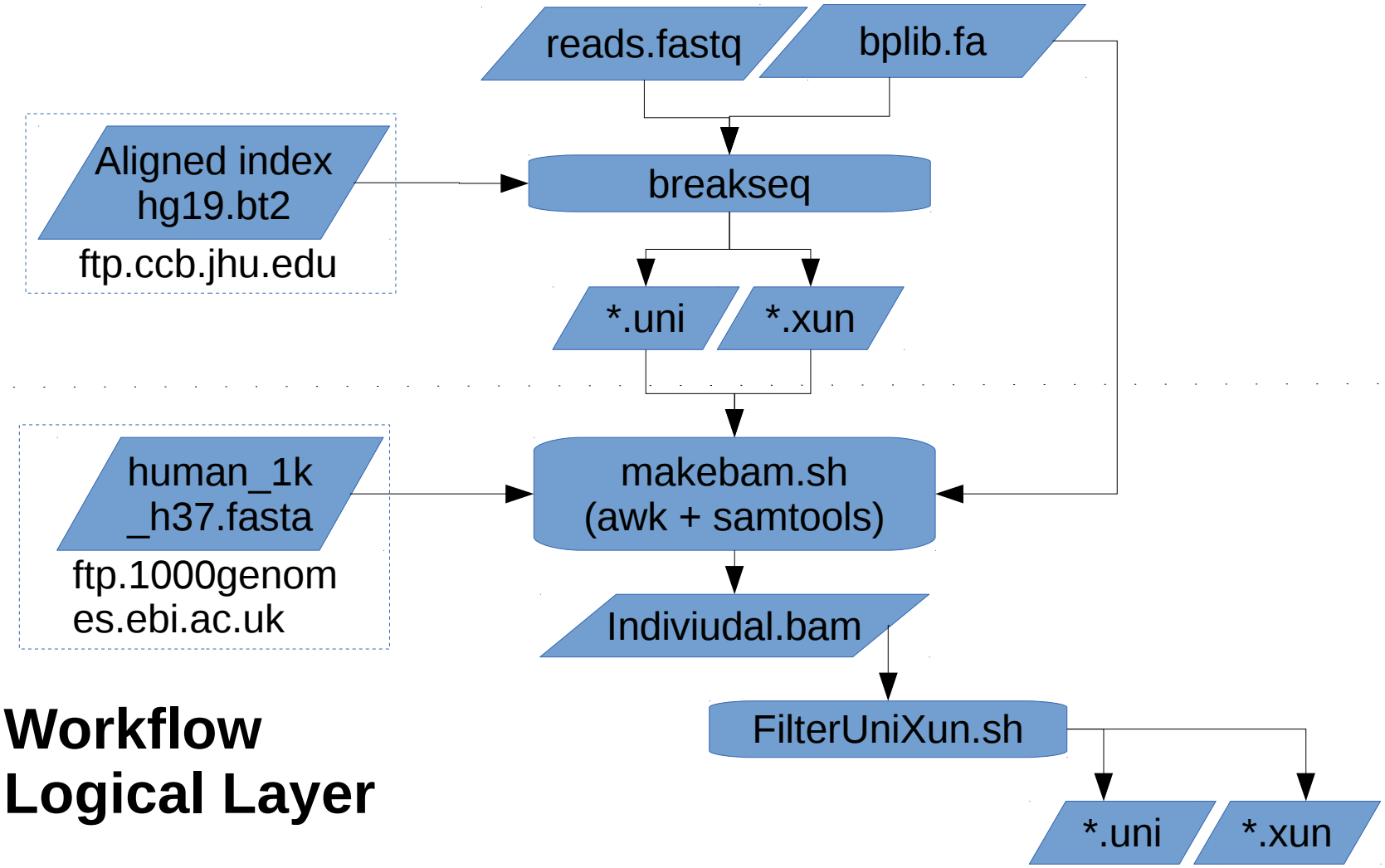
CometCloud mobile data agregation in multi-layered cloud
http://nsfcac.rutgers.edu/CometCloud/sites/nsfcac.rutgers.edu.CometCloud/files/petri-diaz-FiCloud14-cm.pdf

Use Case: S3 1k genome processing

Indivual mapping of breakpoint junctions

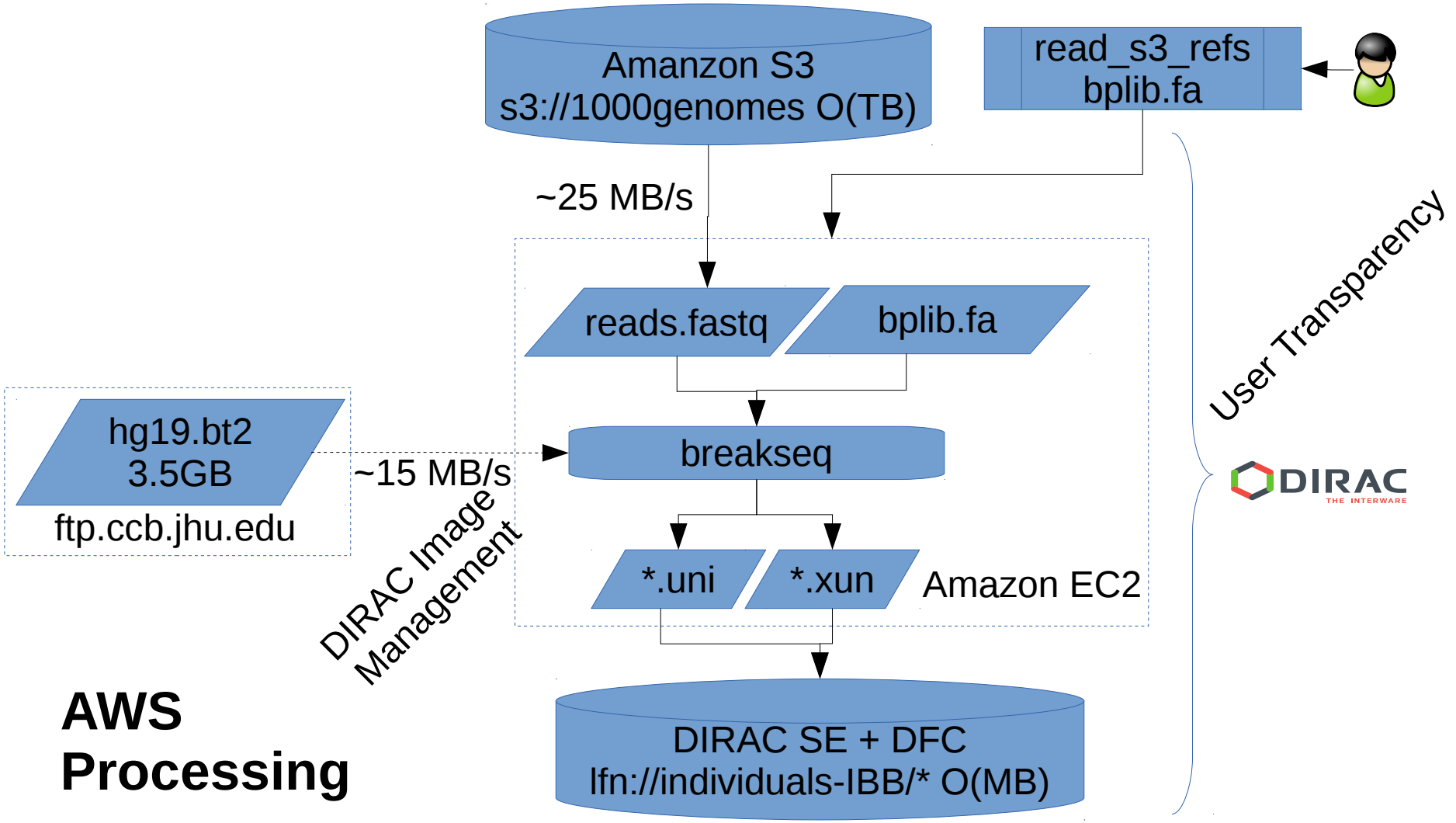
- Breakseq provides genome processing, allowing mapping of reads with breakpoint junctions
- Human individuals specific mapping:
  - Data Patterns:
    - Homo sapiens raw genome: human_g1k_v37.fasta
    - Homo sapines aligned genome sequence: .bt2
    - Breakpoint junctions of interest: bplib.fa
  - Input Data
    - Indivuals to be mapped: reads.fastq
  - Output Data
    - Filtered allele inverted : .uni
    - Filtered reference allele: .xun

Use Case: S3 1k genome processing



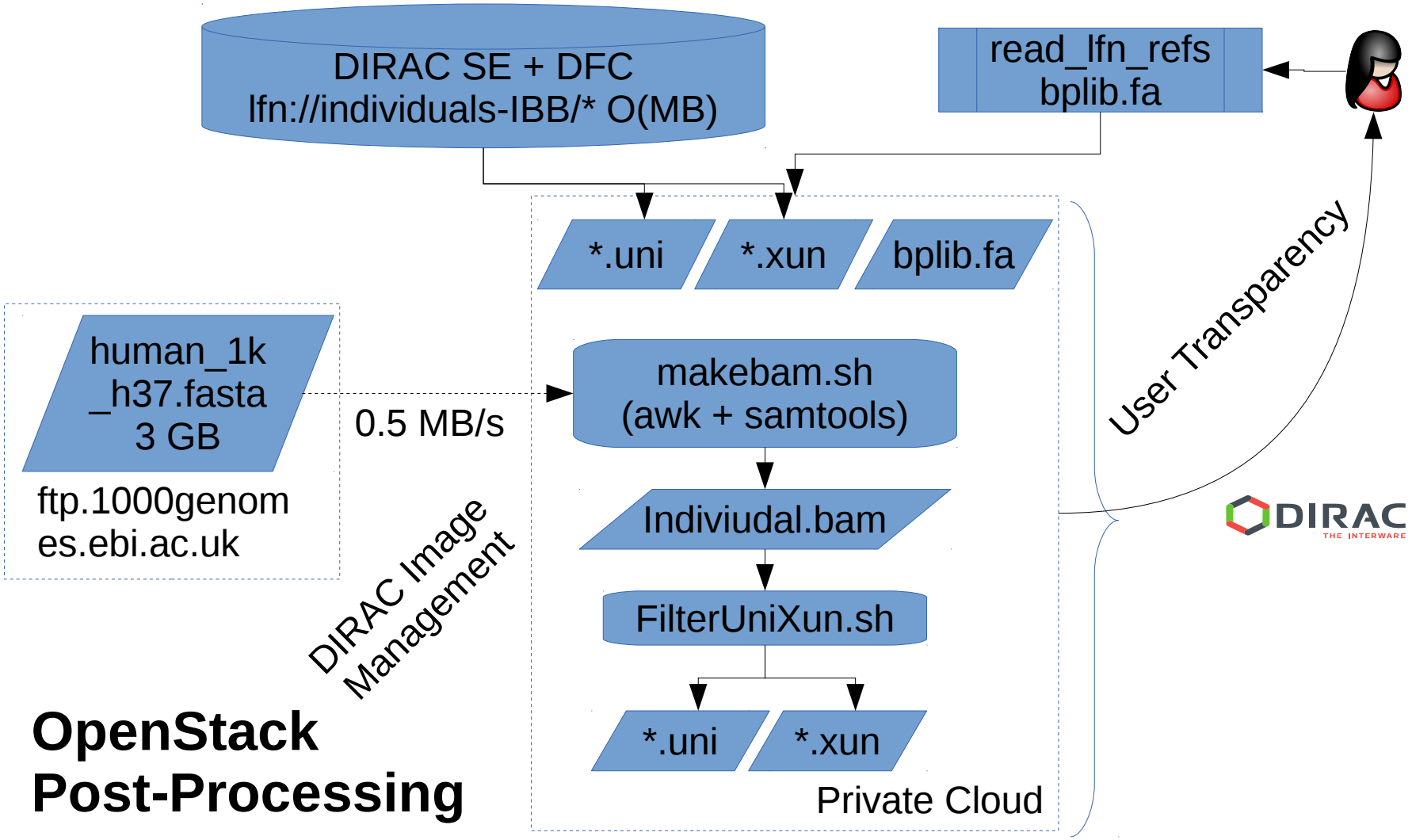
Workflow Logical Layer

Use Case: S3 1k genome processing



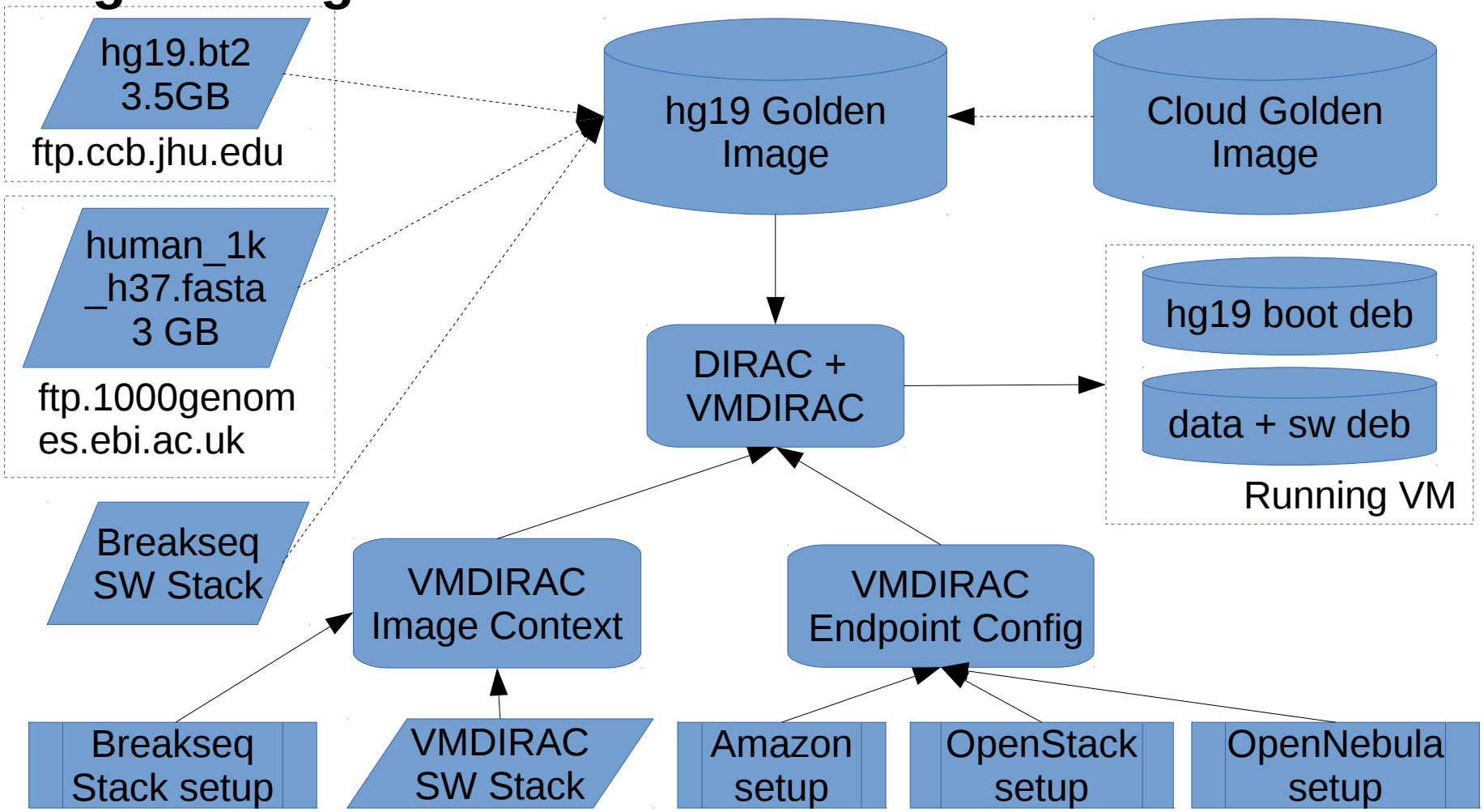
AWS Processing

Use Case: S3 1k genome processing



Use Case: S3 1k genome processing

Image management



Coclusions

- Cloudinit allows a single Image contextualization script for all your IaaS providers
- Cloudinit enables straight forward integration of the pilot 2.0, cloudinit script = pilot 2.0 script
- VMDIRAC setup allows hybrid cloud and multi-layer cloud setup to optimize computational costs and comercial cloud prices
- Everithing is setup and managered by dirac administrator, enabling user transparency in the use of cloud resources

References

- Please, cite related papers in the link:

  https://scholar.google.com/citations?user=2RVFZrwAAAAJ&hl=es

Thanks