

Running BlobSeer inside a Nimbus Cloud

Eliana-Dina Tirsa

Dorneanu Daniela

Politehnica University of Bucharest
Computer Science Departament

!

!

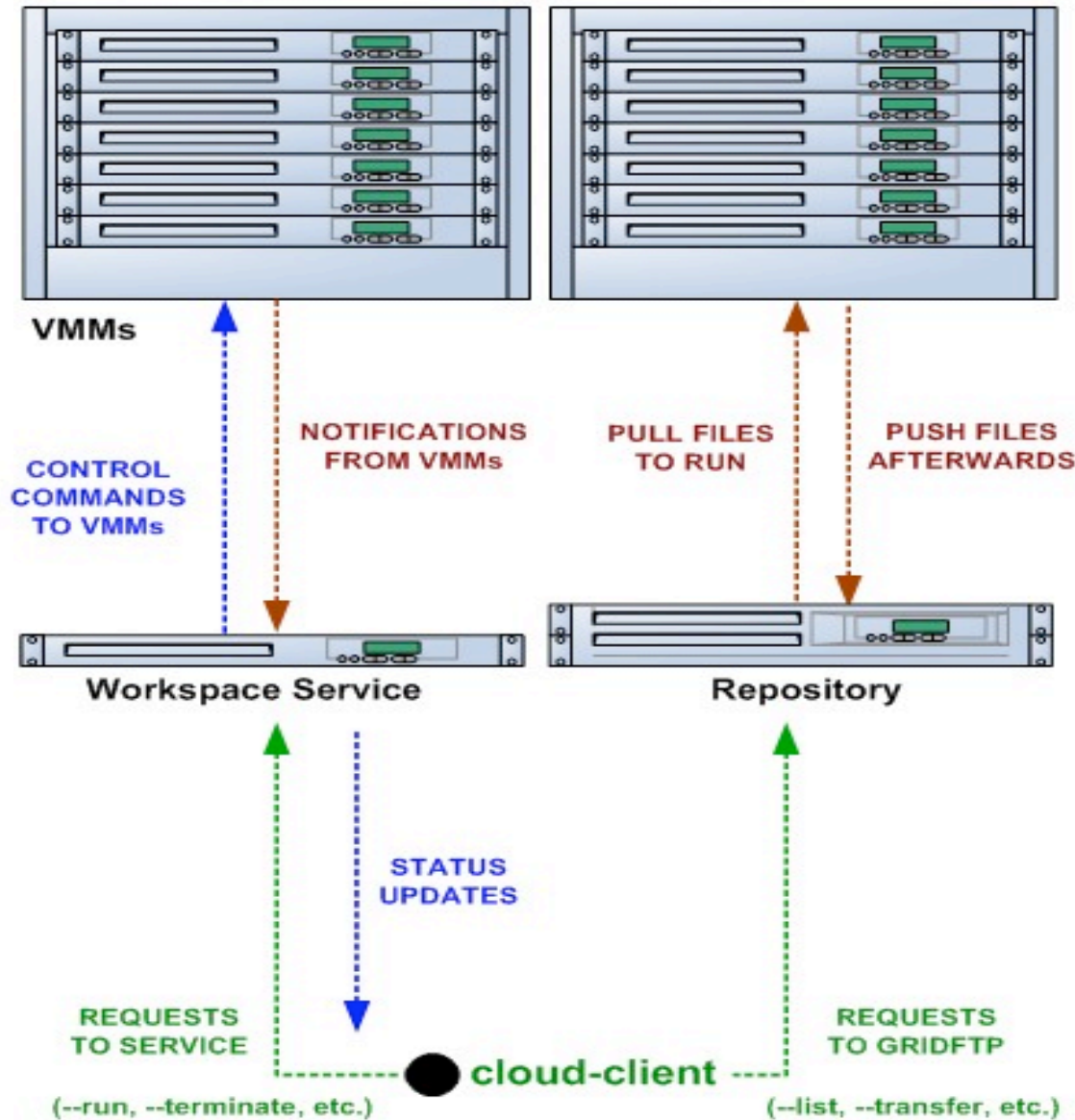
Agenda

- Nimbus overview
- Virtualization solution
- BlobSeer on IAAS – motivation
- BlobSeer on IAAS - objectives
- Laboratory setup
- BlobSeer checkpoint-restart approach
- Test scenarios
- Future work

Nimbus overview

- Developed at Argonne National Laboratory
- Open source IAAS Cloud
- Derived from early GT4 Workspace Management Module
- "Inherited" Globus certificates scheme

Nimbus overview (2)



Workspace Cloud Configuration



Virtualization solution – XEN

- Open source hypervisor based virtualization solution
- supports paravirtualization and full-virtualization
- doesn't require specialized hardware for virtualization, but takes advantage of it
- Weak points:
 - not a Linux kernel
 - Nimbus + libvirt + xen = trouble
- Next time: KVM

!

!

BlobSeer on IAAS – motivation

- 1. Run (map-reduce) applications **inside** the Cloud – use BlobSeer on IAAS for storing final/intermediate data
- 2. Run applications **outside** the Cloud – use BlobSeer as a Cloud service

BlobSeer on IAAS - objectives

- + Setup a lab Cloud environment using Nimbus
- + Deploy BlobSeer entities on various virtual machines in the Cloud
- + One click BlobSeer deployment inside Nimbus
- Add checkpoint-restart functionality to BlobSeer, in order to enable scheduled stops/restarts – partly implemented

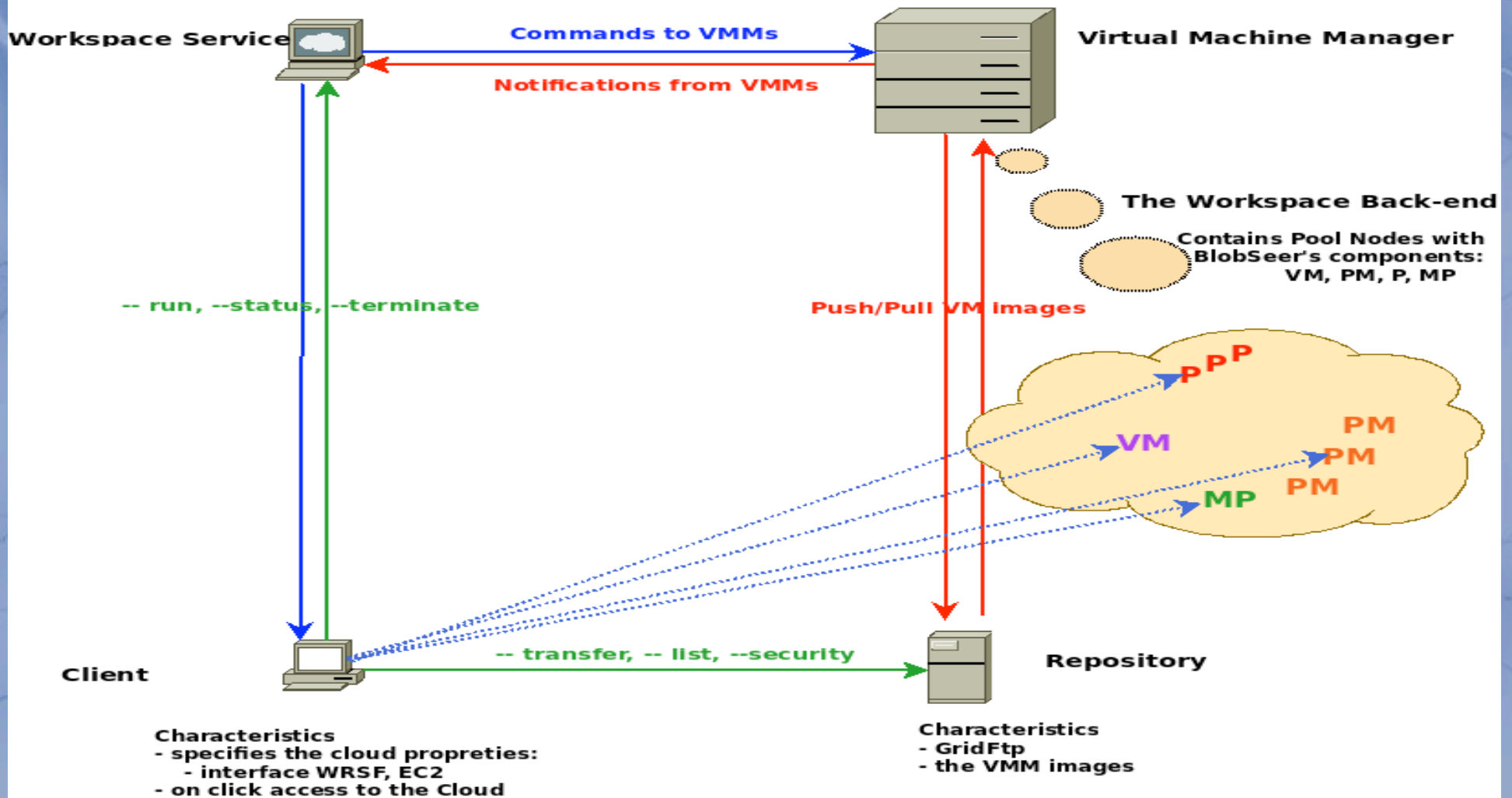
BlobSeer on IAAS -Lab Setup

Characteristics

- connects client and VMM
- protocol adapter -> messaging
- security: ssh, X.509 certificates
- networking
 - public IPs,
 - private IPs via VPN

Characteristics

- accessible through root user
- Xen Hypervisor, Libvirt, DHCP Server, Etables
- Enough RAM memory
- VM control: staging, stopping, pausing
- Integrating a VM into the network: MAC + IP



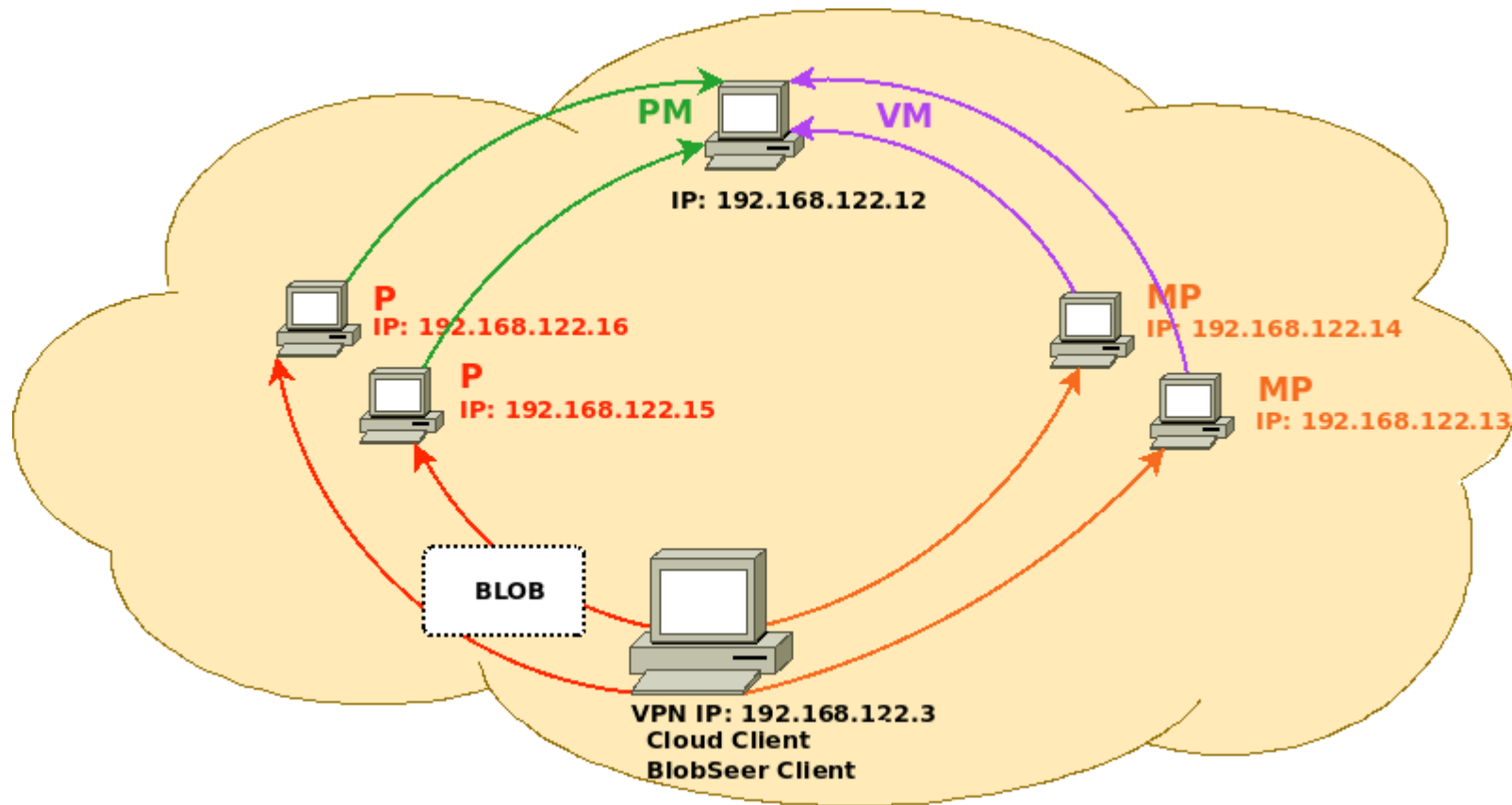
BlobSeer checkpoint-restart approach

- For now – uncoordinated checkpoints for each component
- Validation testing
 - Integration with the rest of BlobSeer components
 - Data persistence
- To do: Coordinated checkpoints

!

!

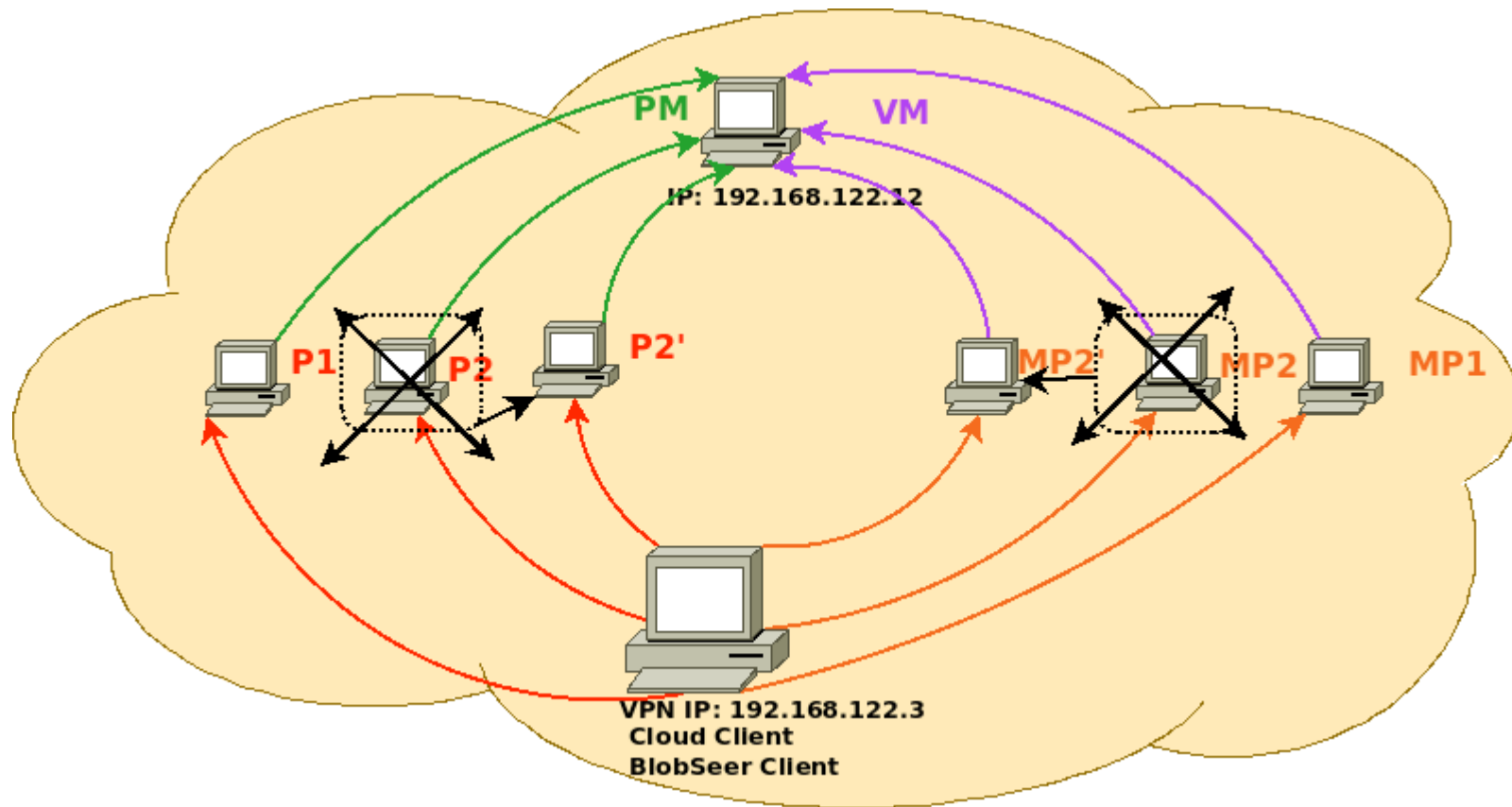
Test scenario 1 – one click Blobseer deployment



!

!

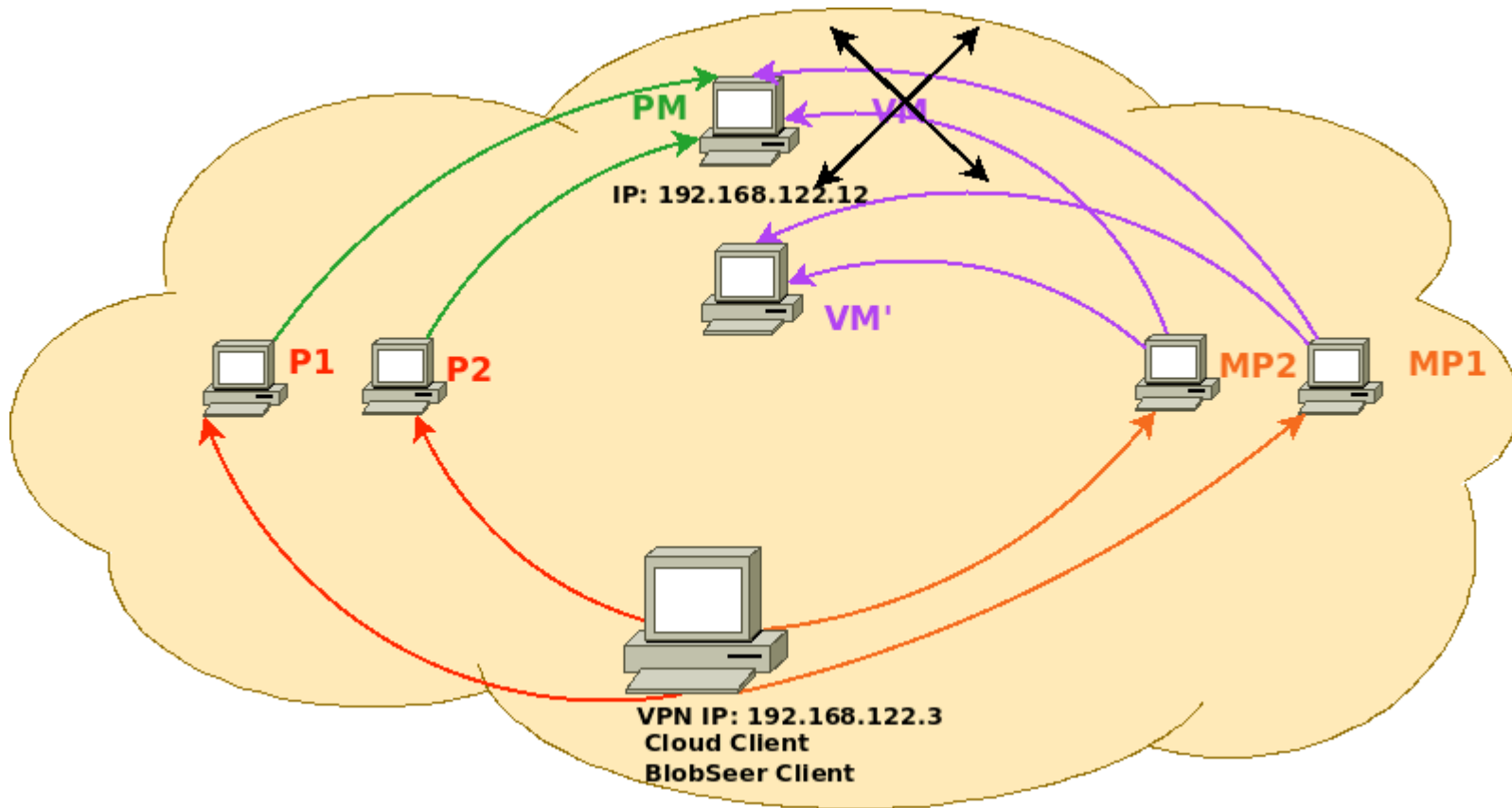
Test scenario 2 – checkpoint restart provider/version provider



!

!

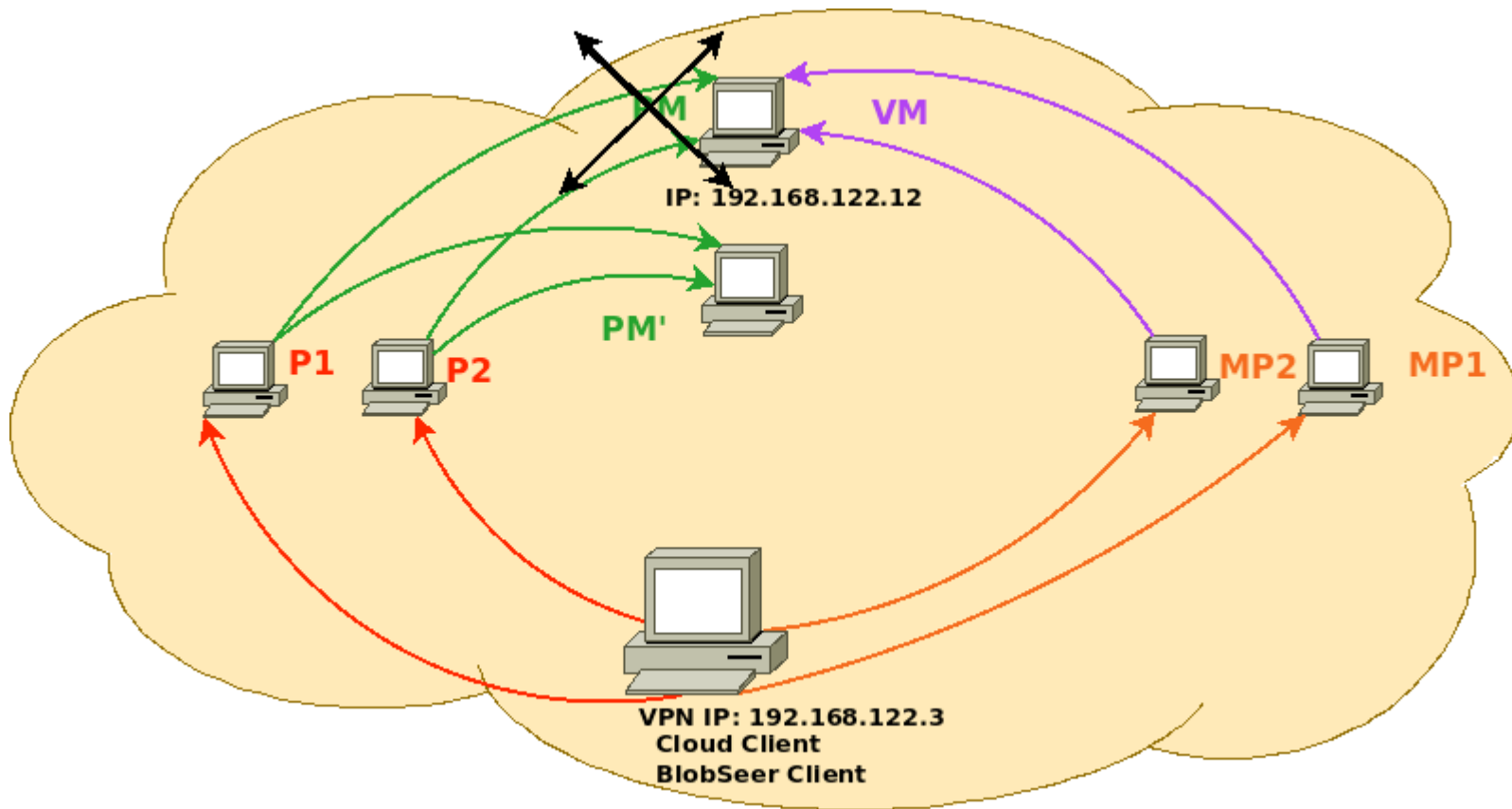
Test scenario 3 – checkpoint restart version manager



!

!

Test scenario 4 – checkpoint restart provider manager



!

!

Future work

- Short time objectives:
 - implement a BlobSeer full-system checkpoint restart mechanism
 - Further testing – inside bigger clouds, with real-life applications
- Related to Bogan's work: integrate BlobSeer as a VMM storage in Nimbus
- To discuss possibilities:
XtreemOS+BlobSeer(+Nimbus)
- BlobSeer as a data storage service – thin clients and beyond

Thank You!

!

!