

POD

PROOF on Demand

PROOF on Demand

Development Status Report

Propaganda (5 slides)

What is PoD?

PROOF on Demand (PoD) is a tool-set, which sets up a PROOF cluster on any resource management system.

PoD is NOT a substitution of PROOF!
It is rather a helper tool for PROOF.

“static”/pre-installed PROOF cluster

- One user can disturb other users.
- From time to time admin. interventions are needed.
- There is only one ROOT/xrootd version for PROOF services.
- There is a master node limitation.

PROOF via PoD

User

- can entirely control his/her dynamic cluster,
- can setup and use it on demand,
- can dynamically change an amount of workers,
- can select a preferable master host,
- doesn't need admins to take an action,
- doesn't disturb other users,
- is free to choose a ROOT version for services.

Key features

- Easy to use
- GUI & Command-line
- Different job managers (SSH, gLite, LSF, PBS, GridEngine, Condor)
- Multiuser/-core environment
- Native PROOF connections
- Packet-forwarding
- User defaults - configuration

PoD v3.x

PoDWorker

PoD Agent

PoD GUI/CLI

PoD server management

PROOF workers monitor

Job Manager plug-in system

SSH

LSF

PBS

Grid Engine

Condor

gLite

...

Current status

v3.1 was released (2011-01-07)

v3.4 was released (2011-02-23)

Ongoing development:

v3.5 - development version

v3.6 - will be the next stable tag

Condor plug-in

The Condor plug-in joined a long list of supported plug-ins.

PoD is now officially shipped with:

SSH, gLite, LSF, PBSPro/OpenPBS/Torque,
Sun/Oracle Grid Engine and Condor
plug-ins.

PoD supports shared
installations

pod-info

- was 100% revamped,
- is the first step to UI and Server disentangling,
- can now also be used to request information from remote PoD servers.

PoD Server

- utilizes now the new pod-info,
- got significant speed up improvements: it takes now less than a second to start,
- got the idle time-out handler improved,
- got a smarter PoD processes handling: cleans precisely only own processes and their children.

PoD Worker

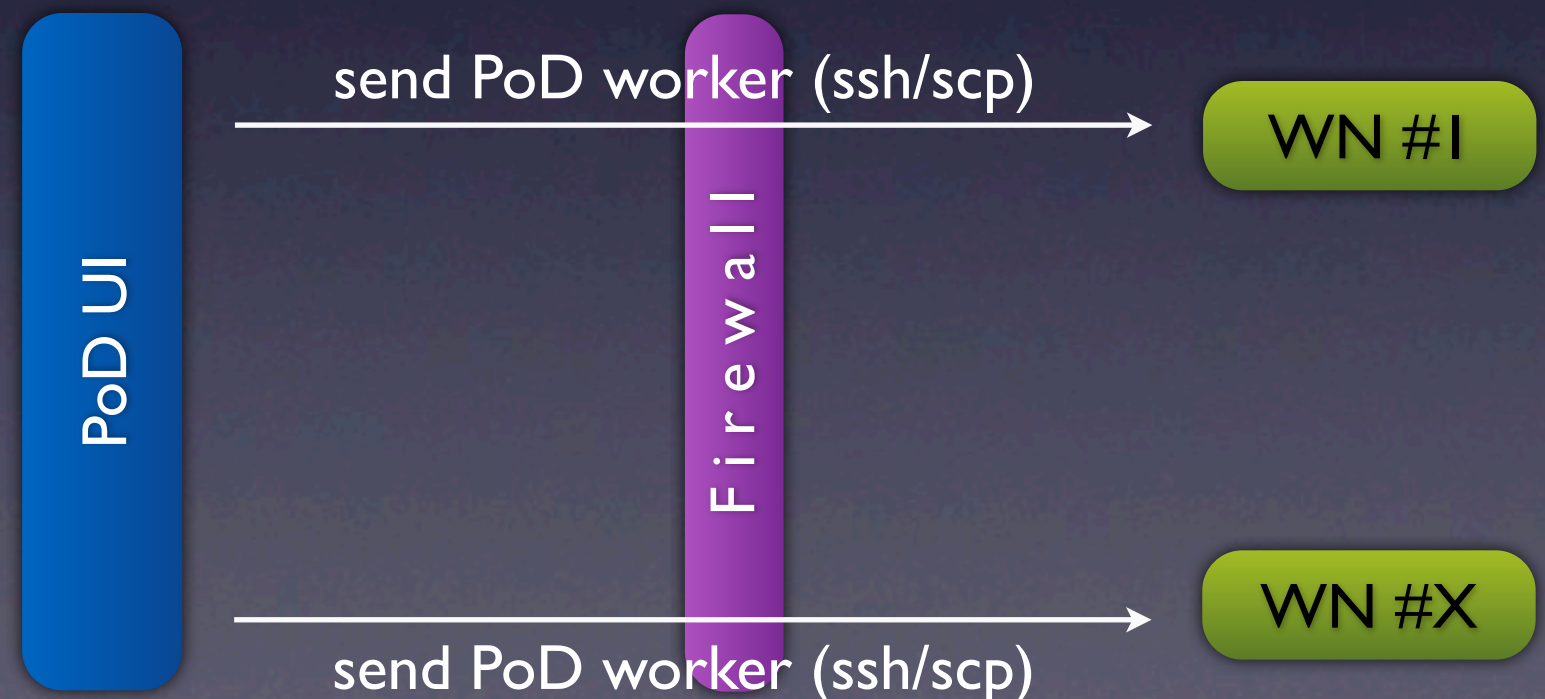
- Worker's script cleans precisely only own processes and their children.
- Worker's package is now updated automatically if any of its component is updated.

SSH plug-in: auto spawn

a simple CSV file as an input to the plug-in:

```
id , login@host , ssh_params , wn_dir , num_of_workers
r1 , anar@lxcg27.gsi.de , -p24 , /tmp/test , 10
a2 , user@lxi001 , , ~/pod_wn ,
125 , doom@host.my , -p22 , /opt/pod ,
```

the plug-in is in PoD
CLI only, so-far

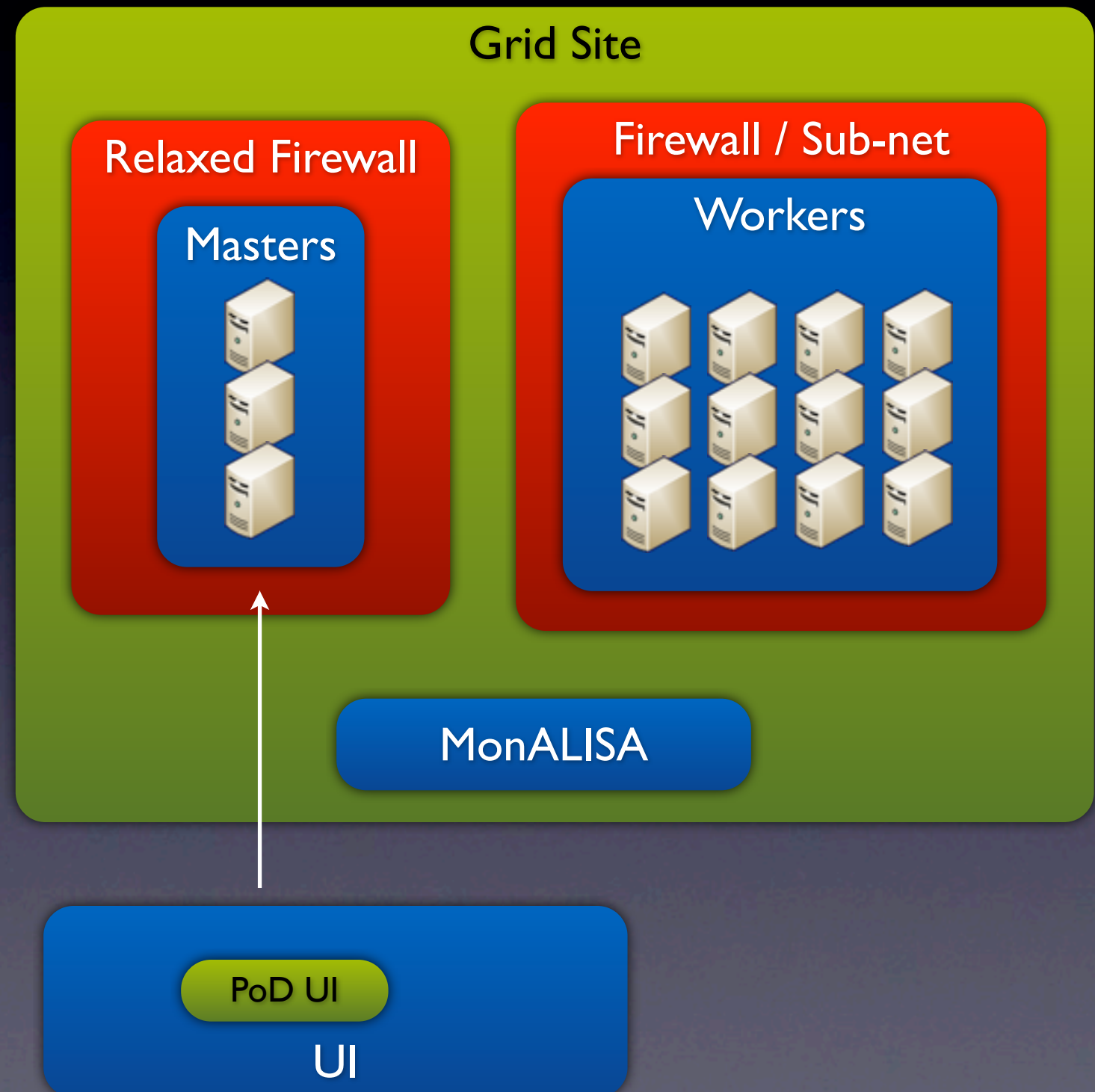


Live Demo

AliEn integration

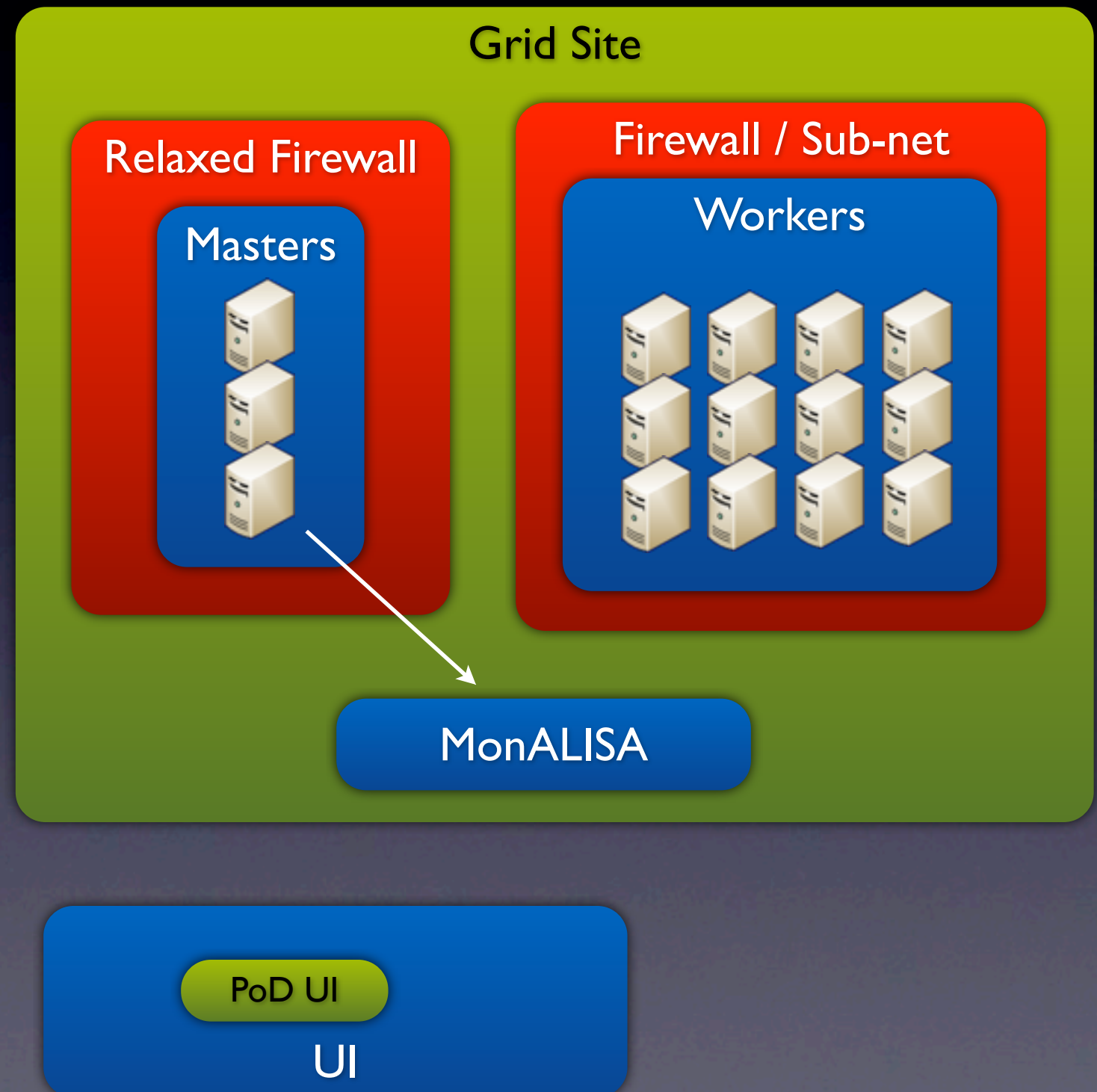
Step #1 User request a PoD server

- User's request is only a configuration file for PoD server. No binaries are transferred.
- AliEn using master queue starts a PoD server on the PoD-aware master machines, using user's config file.
- User could specify, a minimum and a maximum number of workers and for how long (s)he can wait for the max. number of workers.



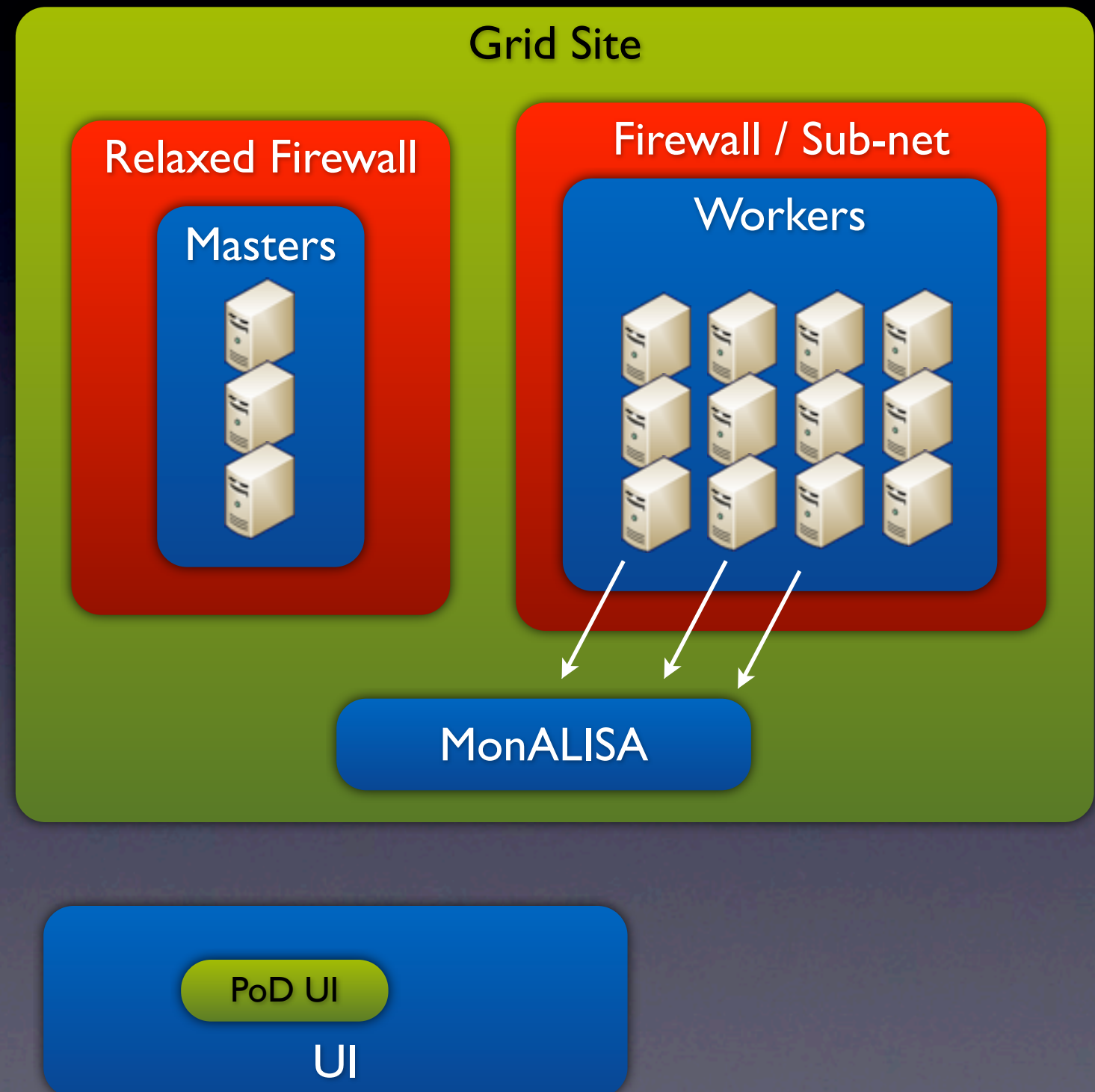
Step #2 PoD server registers in MonALISA

- Server sends info to MonALISA every X sec.
- Server stops sending info, if it:
 - got enough workers or
 - can't wait longer (as user requested) or
 - shut down.



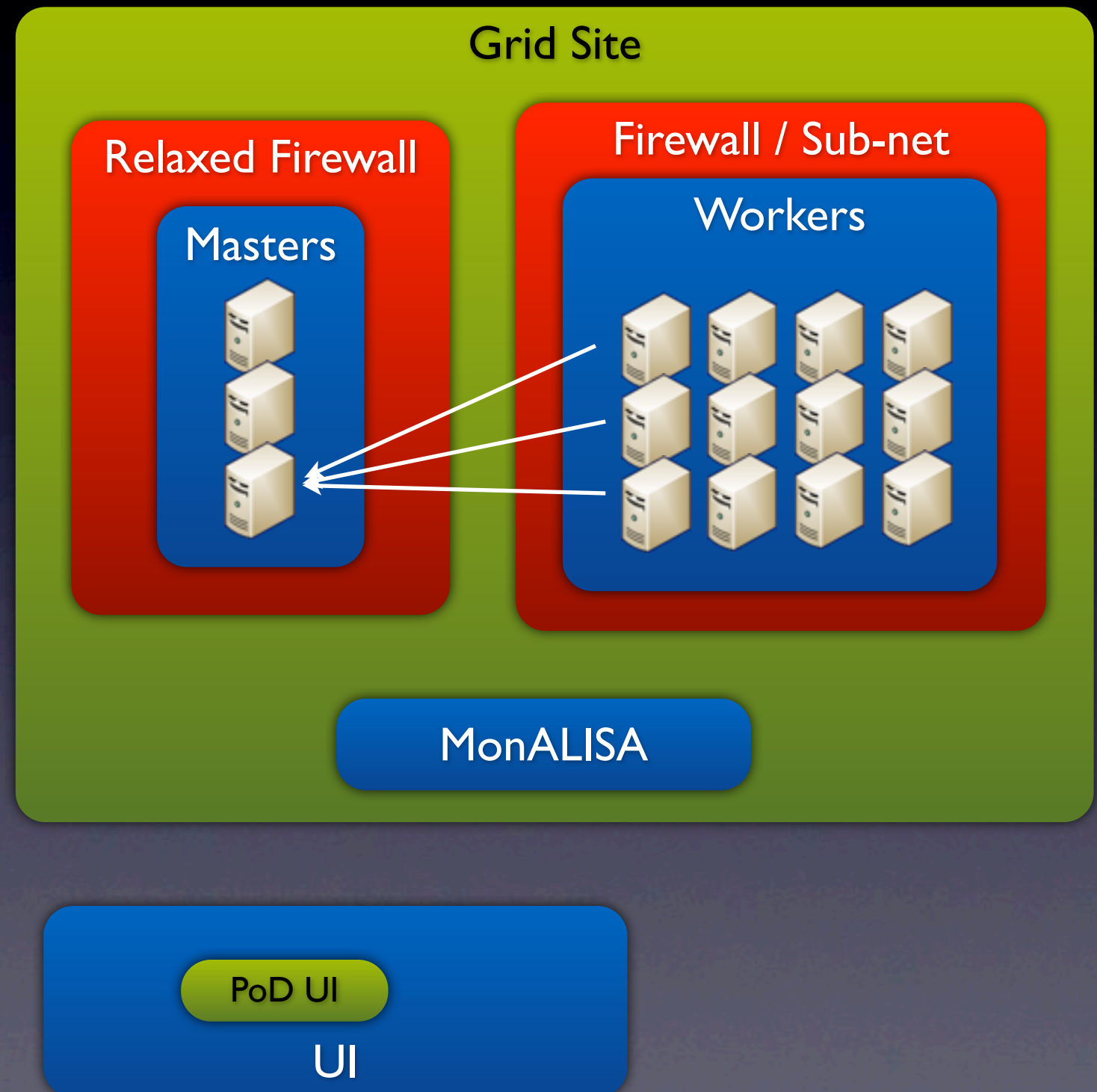
Step #3 AliEn constantly starts PoD workers whenever there are free job slots

- This is a constant process.
- Before starting a regular job on a free slot, AliEn starts a PoD worker.
- PoD worker checks MonALISA for available PoD servers.
- If there is no servers, PoD worker exits after a define amount of time, so that AliEn job slot could take a normal job.



Step #4 PoD workers connect to available PoD servers

- As soon as a PoD server got enough workers, it will stop publishing to MonALISA and will stop to except anymore workers.



Step #5 PoD will process its usual routines in order to setup PROOF

Site Requirements

TCP Ports (small ranges):

1. WNs: incoming from Masters (required only for native PROOF connections, otherwise PoD will use packet-forwarding),
2. WNs: outgoing to Masters,
3. Ms: incoming from WNs,
4. Ms: outgoing to UI,
5. Ms: incoming from UI on PROOF ports.

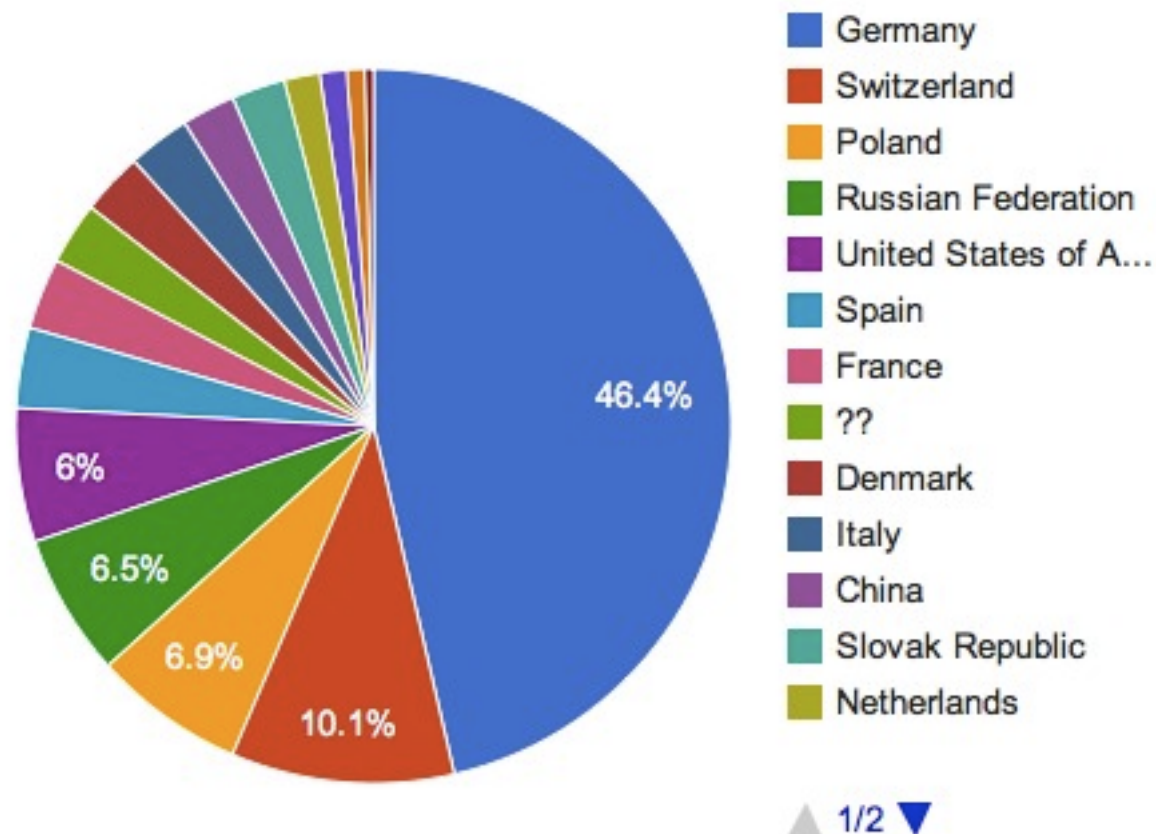
AliEn integration: steps for PoD to accomplish

- implement a shared installation feature, (100 % Done)
- disentangle PoD UI and PoD Server, (60% Done)
- teach PoD WN and Server to understand PoD's user defaults configuration via PoD protocol, (10% Done)
- teach PoD Server and WNs to talk to MonALISA.

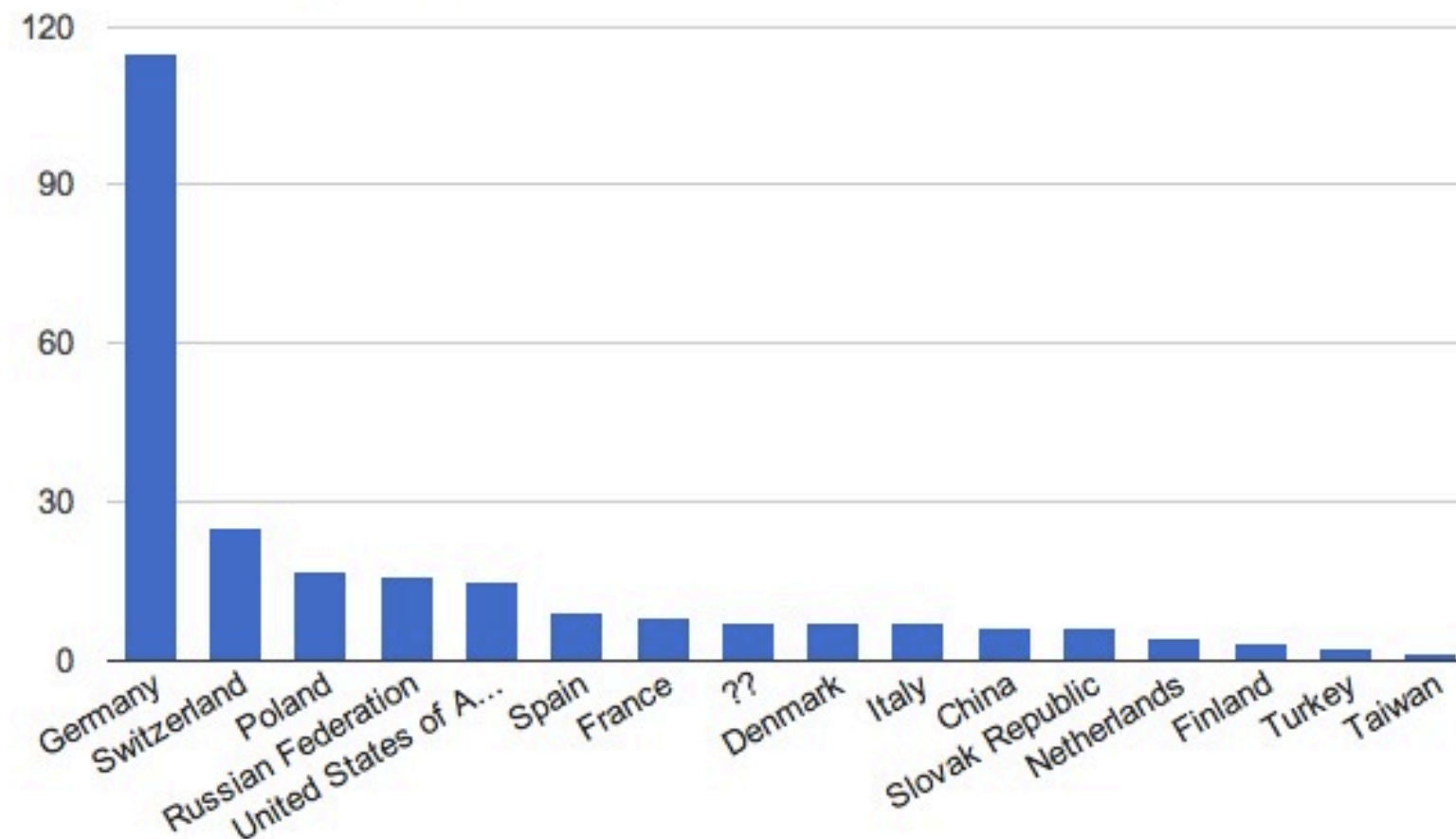
Download Statistics

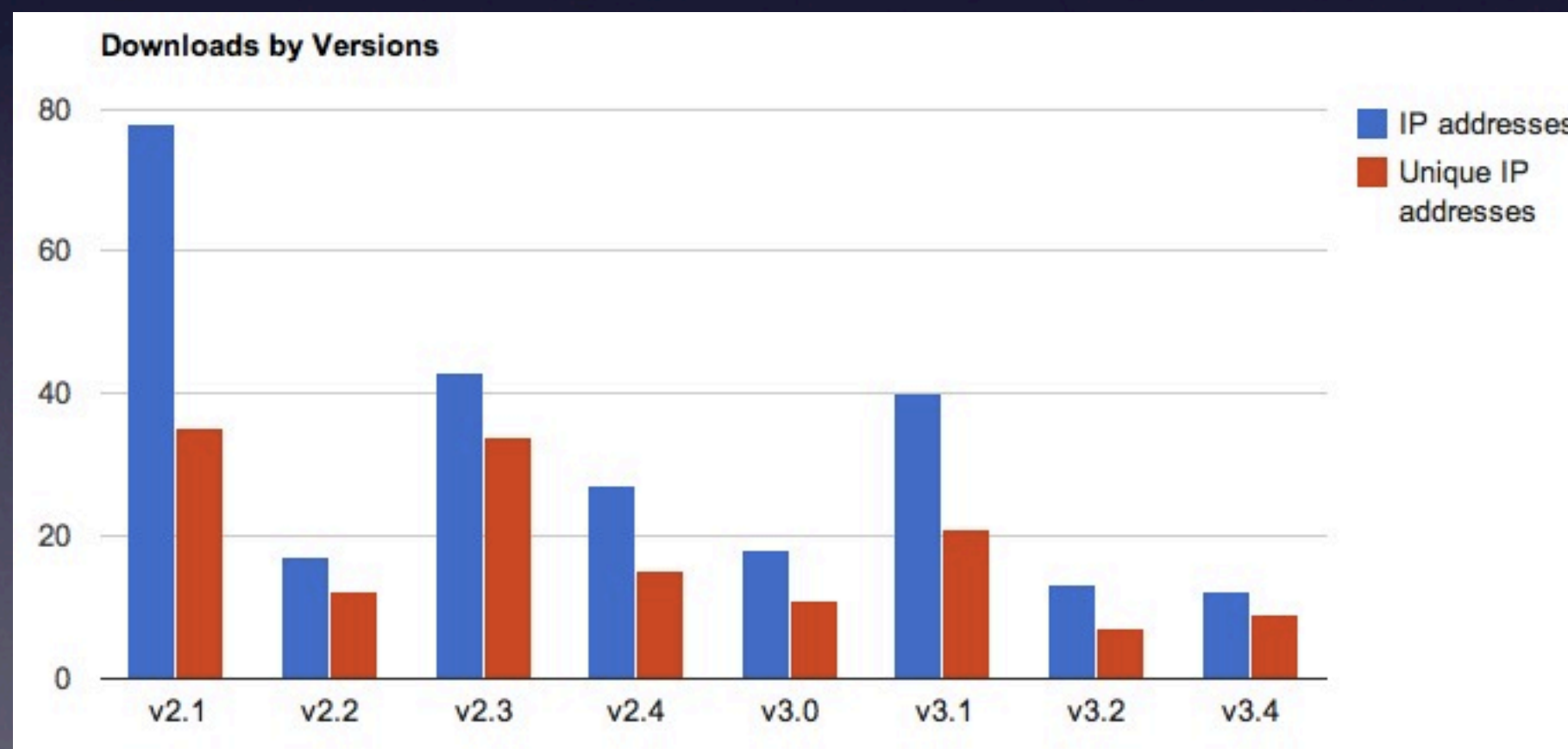
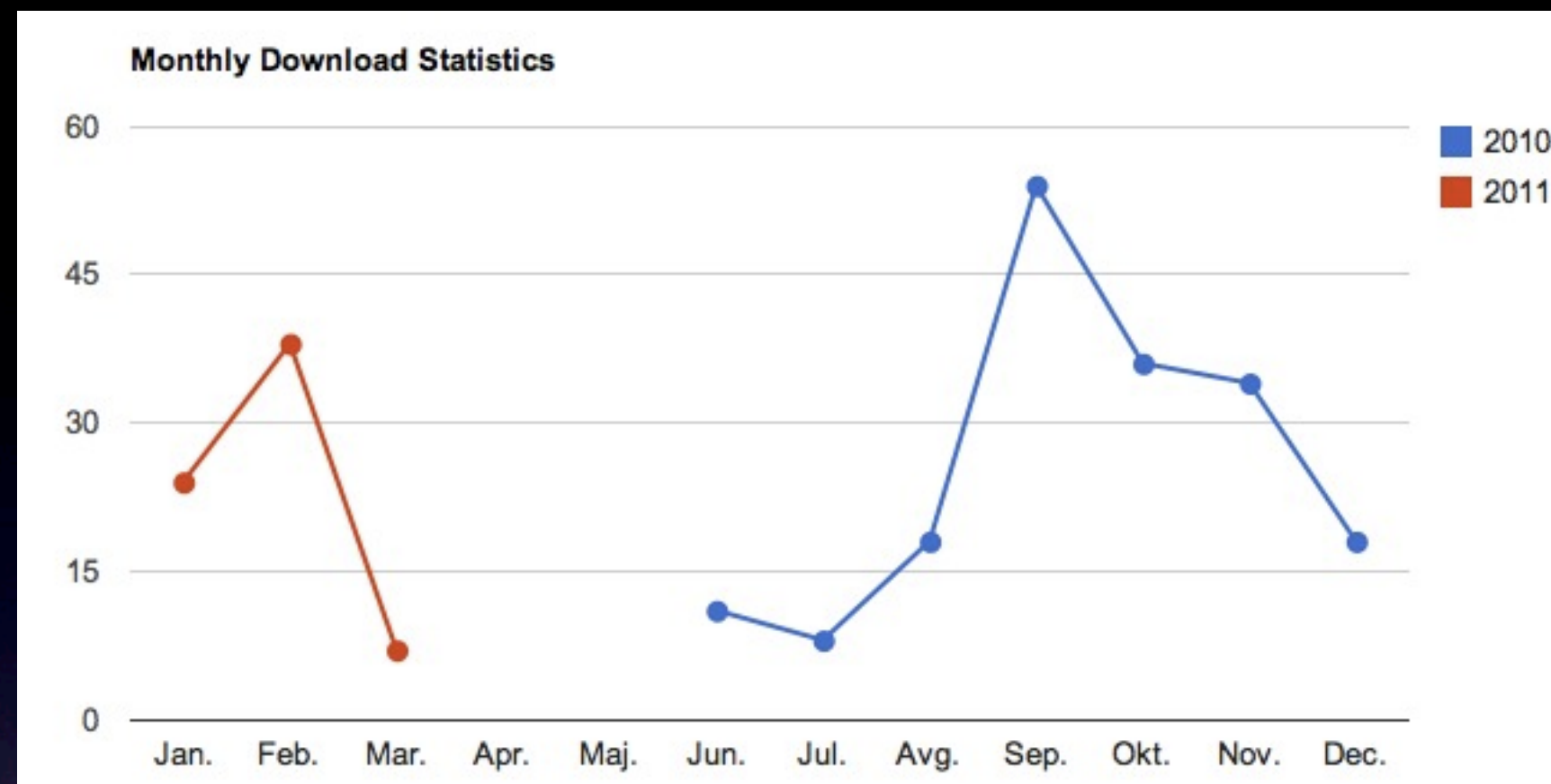
248 Downloads from 15 countries
(since June, 2010)

Downloads by Country



Downloads by Country





Short-term ToDo

- finish the PoD UI and Server disentangling,
- continue with the AliEn integration,
- implement new PoD GUI.

<http://pod.gsi.de>