PODShell: Simplifying HPC in the Cloud Workflow

June 2011

Penguin provides Linux HPC Solutions

Linux Systems

Optimized for Linux

- Intel® & AMD® Rackmount Servers
- Storage
- Networking
- Infrastructure
- GPGPUs
- Professional Workstations

Cluster Management Software

Highly SCYLD

Ease of Management

- Scyld ClusterWare™
- Scyld TaskMaster™
- Enterprise Linux
- Compilers & Tools

HPC as a Service - Penguin on Demand

POD

Elastic Computing

- On-demand environment
- HPC 'optimized'
- Tiered 'Pay-as-you-go' pricing
- Premium set-up and support services

Professional Services and Engineering

Linux and Cluster Expertise

- Factory Integration
- Onsite Installation
- Training
- Product Support
- Software Support
- Customized Service Engagements

PENGUIN COMPUTING

Software Product Line

Remote administrators/users

On-premise systems

Insight

- *Complete physical or virtual cluster management GUI*
- *1:1 match with Scyld CW functionality*
- *Hardware and workflow monitoring*
- *Integrated log viewer for maintenance and diagnostics*

Beoweb

- *Exposes Scyld CW functionality through a secure Web interface (https://)*
- *Enables Web sites and applications to dynamically interact with a Scyld cluster*
- *Dynamically present cluster and job status to users*

PODTools

- *Convenient and secure method to submit jobs and data to POD*
- *Automatically returns results when complete*
- *Customize workflow through a scriptable interface*
- *Generate ad hoc reports to see core hour and GB usage*

Scyld ClusterWare

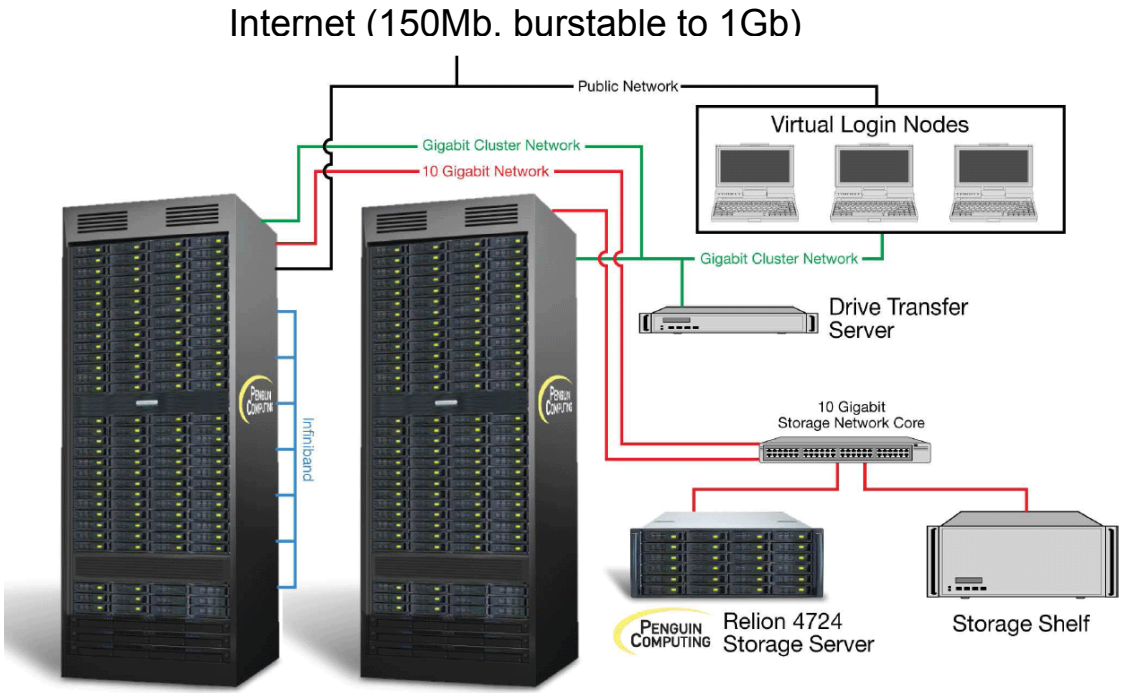
- *Cluster management interface*
- *Minimizes effort required to set-up, update and maintain a cluster*
- *Guarantees consistency across all servers*
- *Provides complete, tested and supported cluster software stack*

HPC in the Cloud

PENGUIN COMPUTING

What is POD

- On-demand HPC resource
- Virtualized login node
- Physical compute nodes
- 10 Gig direct attached NFS storage
- Panasas HPPFS Available
- GigE, 10Gig and IB interconnect
- Deployed in ‘scalable units’
- 1 to 480 cores available on-demand



The New EDU POD

- Owned and operated by the Penguin Computing on Demand staff
- Hosted at Indiana University data center
- AMD based compute infrastructure (shared environment)
  - Altus 1804 servers
  - 48 cores per server (Opteron 6174 12C, 2.2 GHz)
  - 128GB, DDR3-1333 RAM per server
- Lustre scratch file system (100TB)
- QDR (40Gbps) Infiniband
- Scyld Cloud Management System
- Internet, Internet2 connectivity

- Transferring data to/from POD

- SCP

Upload bandwidth	Time to transfer 1GB	Time to transfer 50GB	Time to transfer 200GB
1 Mbps	2 hours	5 days	Take a 2 week vacation…
10 Mbps	13 minutes	11 hours	2 days
50 Mbps	3 minutes	2 hours	9 hours
150 Mbps	53 seconds	45 minutes	3 hours

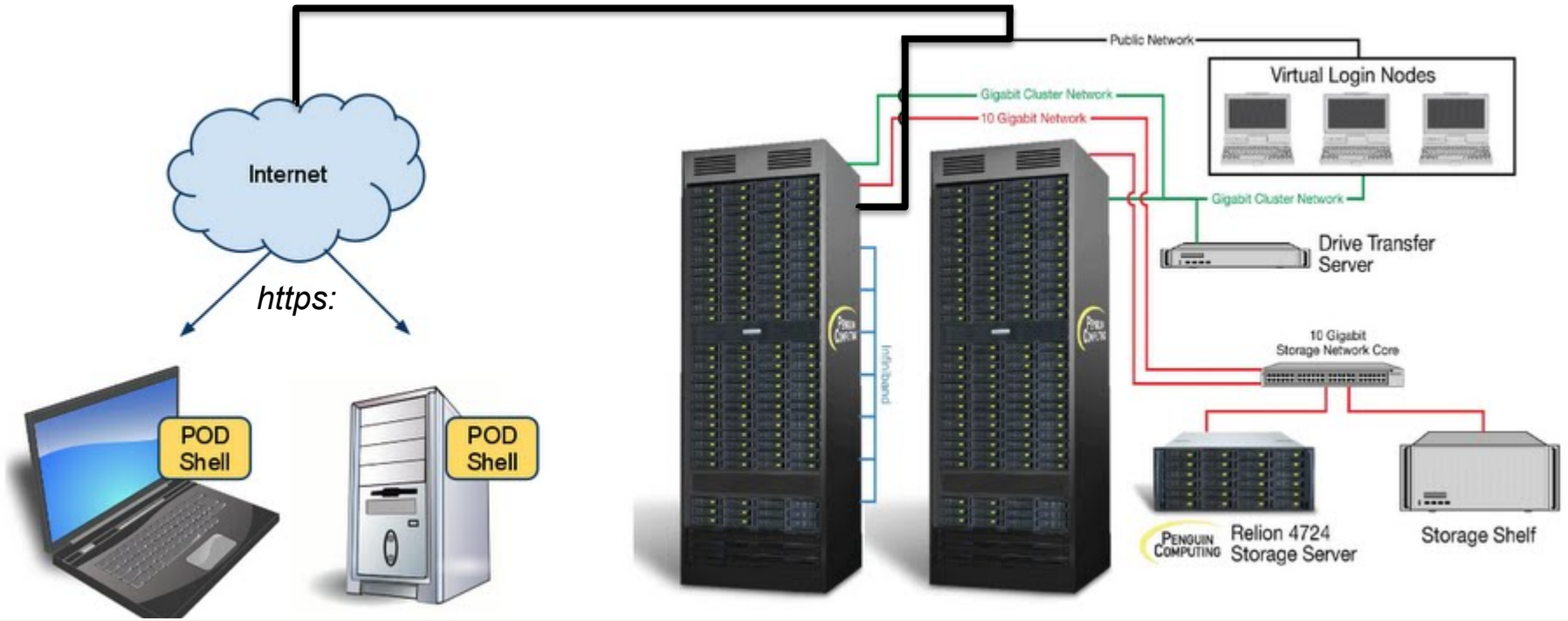
- POD Disk Caddy (free, but the user pays the FedEx fees)
- PODShell remote data staging (more on this later)

- Dedicated Storage Node serves as a physical Login Node
- User data transferred to POD on disks can be immediately available
- User can have complete control over data transferred (software RAID, encryption, etc.)
- Discounts apply to quarterly or yearly contracts

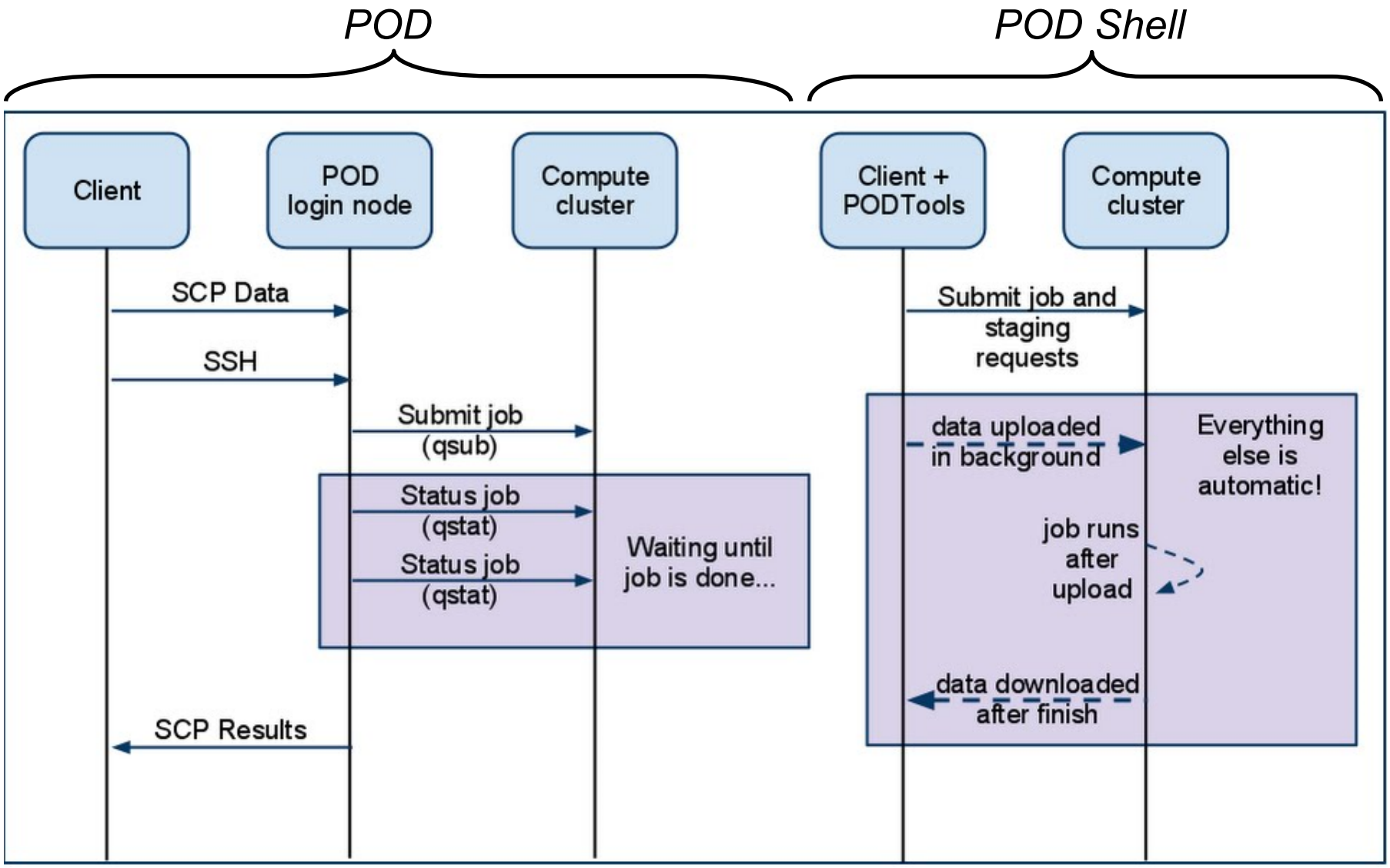
PODShell – overview

- Combine the power of “submit”, “stage-in” and “stage-out” to create a single command that executes your entire workflow (application, data and job submission)

```
[root@mojito ~]# podsh submit --stagein=mydata:~/data --stageout=
\~/myresults.log myjob.sub
Job Submitted successfully.  Job ID: 1590
```



Comparing POD and PODShell



How PODShell is set up at Caltech

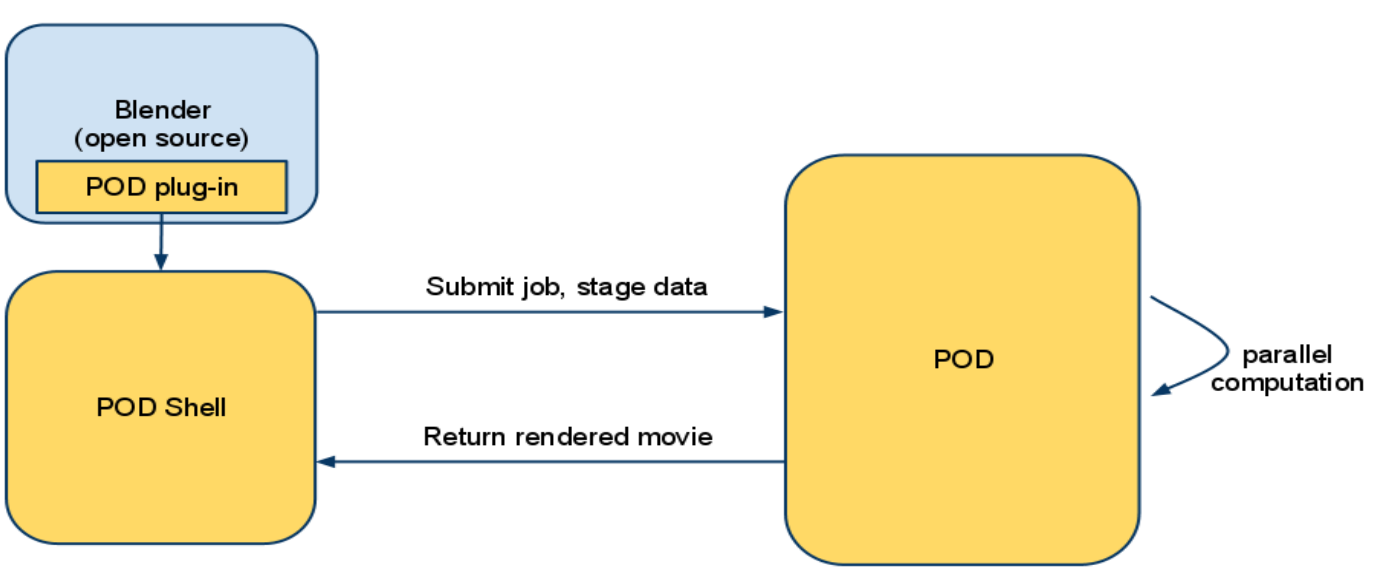
- Dedicated submission host with same libraries and compilation environment as POD
  - Enables upload of binary compatible executables
- Automated user creation based on LDAP attributes
- Caltech/Penguin jointly managed
- For help, contact help-hpc@caltech.edu

PODShell system requirements/compatibility

- PODShell is installable on RHEL4 or RHEL5 systems (or CentOS)
- If you don't need binary compatibility
  - Download RPMs for:
    - PODTools (includes PODShell and PODReport)
    - Python 2.6.5
    - Python libraries/utilities
- If you need binary compatibility
  - Ask for your own Virtual Submission Host (VM)
    - RPMs are already installed
- RHEL6 support in Q3 2011
- Internet access
- Firewall open to ports 5000, 10000-10100

PODShell – it's scriptable…

- Since PODShell is a command line interface, it's scriptable
- Allows automated workflows to POD without having to script SSH access and data transfers
- Can be called from other programs with a "plug-in" capability
- At SC10, we demoed a Blender plug-in built on PODShell that exported the scene to POD, rendered the frames, created the movie/ animation, then copied result back to client machine



PODShell – security

- PODShell communicates with POD using HTTPS and TCP
- HTTPS used for everything except data staging
- TCP sockets used for data staging
- BOTH encrypted with TLSv1 (successor to SSLv3)
- POD does not accept unencrypted connections with PODShell

PODShell – approximate Caltech prices

Service	Fee
Core hour rates	$0.20 (before Caltech discount)[1]
On-demand storage	$0.20 per GB-month (before Caltech discount)
Optional storage server, 72 TB raw	$1600/month (user supplies disks)
Persistent login node	Not required!
Bandwidth	Included
Support	Included
Disk transfers	TBD

1. Does not include 3rd party application fees, if required.

PODShell – submitting jobs

- POD supports two schedulers, TORQUE and SGE
- PODShell acts as a “remote qsub”
- Submit your job script with podsh submit

```
[root@mojito ~]# podsh submit test.sub
Job Submitted successfully.  Job ID: 1586
```

- PODShell can submit to different schedulers from the command like with --sched= option:

```
[root@mojito ~]# podsh submit --sched=SGE test.sub
Job Submitted successfully.  Job ID: 1587
```

PODShell – data staging

- PODShell can copy your data for you
  - As part of a job, or independently
- Use --stagein and/or --stageout options

```
[root@mojito ~]# podsh submit --stagein=mydata.bin
test.sub
Job Submitted successfully.  Job ID: 1588
```

- Your job will automatically be held until stagein completes!

```
[root@mojito ~]# podsh submit --stageout=\~/
myresults.log test.sub
Job Submitted successfully.  Job ID: 1589
```

- Stageout will not occur until after the job finishes

PODShell – checking job status

- podsh status gives you output similar to qstat
- Summary or detailed view, plus data staging information

```
[root@mojito ~]# podsh status

================================================================================
Id           Type     State        Job Name     S Stage-in     Stage-out
------------ -------- ------------ ------------ - ------------ ------------
361          COMPUTE  COMPLETE     N/A            FINISHED     NONE
1214         COMPUTE  COMPLETE     N/A            FINISHED     NONE
1373         COMPUTE  COMPLETE     N/A            FINISHED     FINISHED
1590         COMPUTE  RUNNING      test.sub     R NONE         NONE

[root@mojito ~]# podsh status 1590
ID: 1590
    Type: COMPUTE
    State: RUNNING
    Join_Path = oe
    fault_tolerant = False
    exec_host = n86/0
    Resource_List.neednodes = 1:ppn=1
    Resource_List.walltime = 00:05:00
```

PODReport

- PODReport is part of our PODTools “suite” of tools – which also include PODShell
- Allows for custom usage reports for core-hours and storage

```
[root@mojito ~]# podreport -t cpu
Please note that CPU reports can take a minute or more to generate
Password:
CPU Report generated for user trhoden
================================================================================
Date range: 2011-05-26 to 2011-06-14
================================================================================
User                                                                Core-Hours
================================================================================
trhoden                                                                   0.04
[root@mojito ~]# podreport -t storage
STORAGE Report generated for user trhoden
================================================================================
Date range: 2011-05-26 to 2011-06-14
================================================================================
User                 Path                                    Storage Used (GB)
================================================================================
trhoden              /home/trhoden                                        0.59
trhoden              /panasas/trhoden                                      0.0
                                                                     =========
                                                     Total (GB):          0.59
```