

# Lightweight Scheduling for the PRAGMA Cloud Testbed

**Shava Smallen,**  
Nadya Williams, Phil Papadopoulos

October 9, 2015  
PRAGMA29

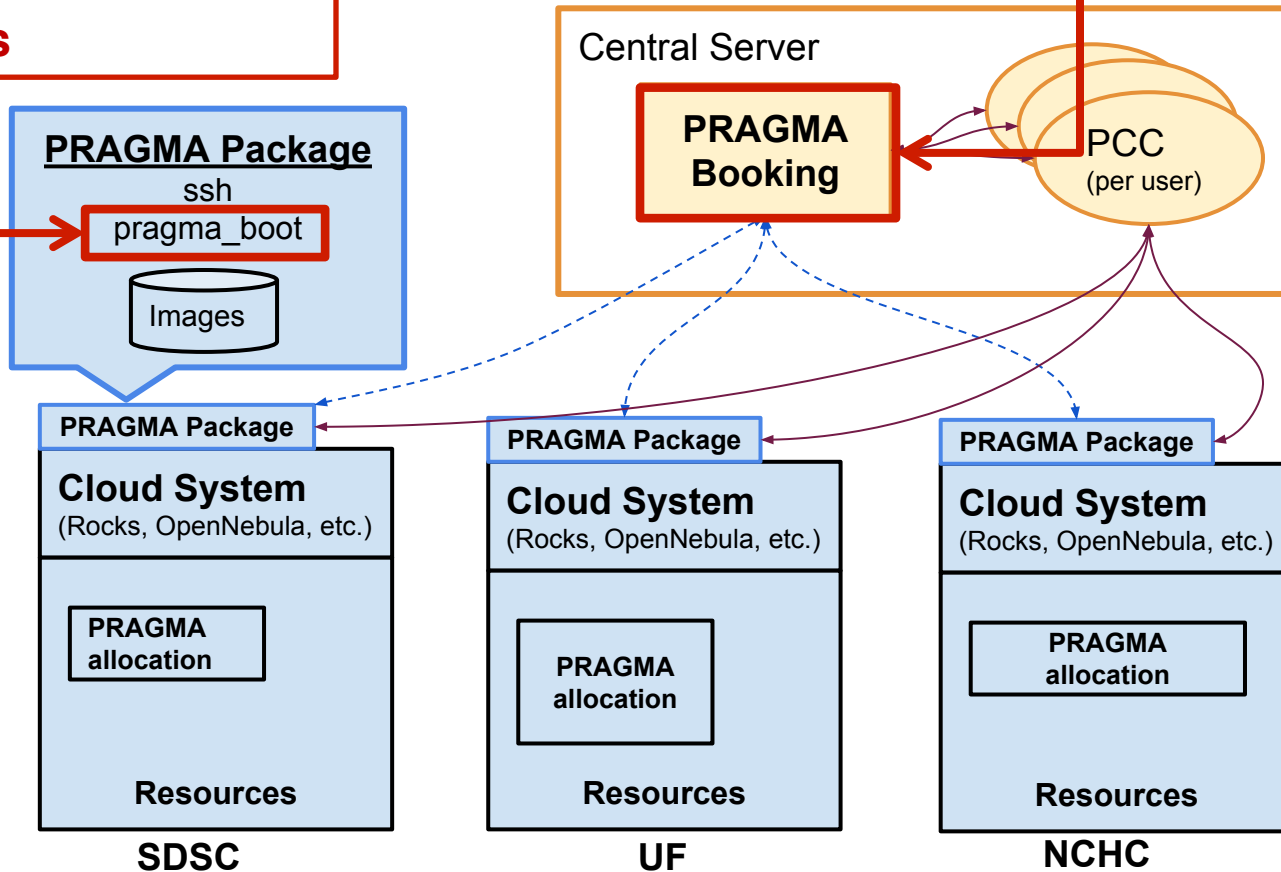
# PRAGMA Cloud Scheduler

- **Goal: Leverage a simple web-based calendar scheduler for sharing of PRAGMA resources.**
- **Leverages the following tools:**
  - **pragma\_boot**: Boots virtual clusters for users across PRAGMA institutions using local VM provisioner. Currently supports Rocks and OpenNebula.
  - **Personal Cloud Controller**: Manages startup, status monitoring, and shutdown of a virtual cluster. Built on top of pragma boot and HTCondor.

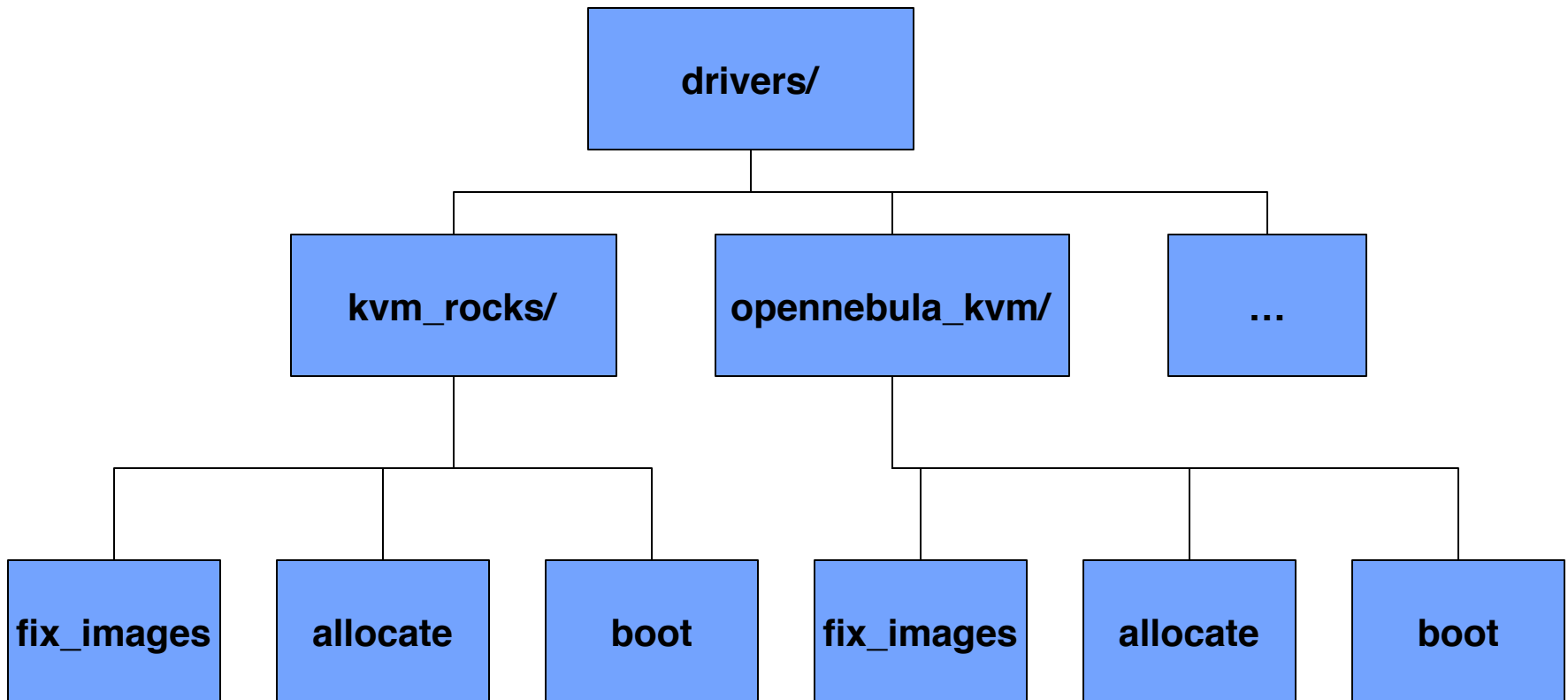
# Architecture Overview

**Demo: Boot of ZFS based virtual clusters**

**Demo: Make multiple reservations per resource**



# pragma\_boot



# pragma\_boot process (kvm\_rocks)

```
$ pragma_boot --vcname rocks-localdisk-basic --key id_rsa.pub --num_cpus 2
```

## Step 1: fix\_images

- Finds the locations of FE and compute images from vc-in.xml in the repository
- Uncompresses images if zipped; images must be **dynip** enabled
- **Copies (cp)** frontend and compute images to local temp dir

## Step 2: allocate

- Finds unused public IP and vlan and # of compute nodes
- Creates cluster (rocks add cluster”) and vc-out.xml

## Step 3: Boot FE

- Guestmounts image and installs FE vc-out.xml snippet
- **Copies (cp)** image to local KVM directory and boots VM

## Step 4: Boots compute(s)

- Guestmounts image and copies over compute vc-out.xml
- **Transfers (scp)** image to VM container and boots VM

**Takes about ~20 minutes to boot basic 2-node rocks cluster**


# ZFS based virtual clusters


- Leverages recent Rocks virtual cluster image management development work for SDSC's Comet resource.
- Based on ZFS with two modes for VM containers:
  - `img_sync=false`: Mounts ZFS volume from NAS storage
  - `img_sync=true`: Snapshots ZFS volume from NAS storage and sends it local VM container; Snapshots regularly to sync the two volumes.
- Phil described in more detail during resources-wg update




# Start ZFS Reservation

Create a new reservation

Shava Smallen (ssmallen@sdsc.edu) [Change](#) 

Resources to be reserved [Accessories](#) [Add](#) 

[SDSC cloud3](#) [More Resources](#) 

Begin

End

Reservation Length **1 days, 0 hours**

Repeat

Title of reservation

Description of reservation

Additional Attributes

CPU:  Memory (GB):  ENT-enabled:  VC Name:


# ZFS Reservation Confirmed

Create

Cancel

Create a new reservation

Shava Smallen (ss)  
Resources to be re  
[SDSC cloud3](#) More F  
Begin 2015-10-04  
End 2015-10-08  
Reservation Length  
Repeat Does Not  
Title of reservation  
Demo rocks-zfs-ba  
Description of rese  
  
Additional Attrib  
CPUs: 32



Your reservation was successfully created!  
Your reference number is 5611dc46c46f4966296376  
2015-10-04  
Resources: SDSC cloud3  

Close

Create

Cancel

**SDSC** SAN DIEGO  
SUPERCOMPUTER CENTER

UNIVERSITY  
OF  
CALIFORNIA



# Confirm ZFS Reservation

Past

Pending

Reservable

Reserved

My Reservation

Starting VMs

Running VMs

Stopping VMs

Resource Filter

Clear Filter

- All -

All

Advanced Filter

Minimum Capacity

Resource Type

- All -

Available CPUs:

Available GB Memory:

Deployment type:

--

ENT-enabled:

--

Site hostname:

Filter

Sunday, 2015-10-04	00:00	01:00	02:00	03:00	04:00	05:00	06:00	07:00	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
SDSC cloud1																								
SDSC cloud2																								
SDSC cloud3	Unavailable																							
UF cloud																								
Monday, 2015-10-05																								
SDSC cloud1																								
SDSC cloud2																								
SDSC cloud3																								
UF cloud																								
Tuesday, 2015-10-06																								
SDSC cloud1																								
SDSC cloud2																								
SDSC cloud3																								
UF cloud																								
Wednesday, 2015-10-07	00:00	01:00	02:00	03:00	04:00	05:00	06:00	07:00	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
SDSC cloud1																								
SDSC cloud2																								
SDSC cloud3																								
UF cloud																								

SDSC

Description

Rocks 6.2 KVM. Hosting Virtual clusters and virtual machines

Notes

(no notes)

Contact

(no contact information)

Location

UCSD/SDSC

Resource Type

(no resource type set)

2 Reservations

Demo rocks-nfs-basic reservation (Shava Smallen)

2015-10-04 00:00:00 - 2015-10-09 00:00:00

Status: **Running**

CPU:32

Memory (GB):8

ENT-enabled:no

VC Name:rocks-nfs-basic

Edit

Demo rocks-zfs-basic reservation (Shava Smallen)

2015-10-04 00:00:00 - 2015-10-08 00:00:00

Status: **Starting**

CPU:32

Memory (GB):8

ENT-enabled:no

VC Name:rocks-zfs-basic

Edit

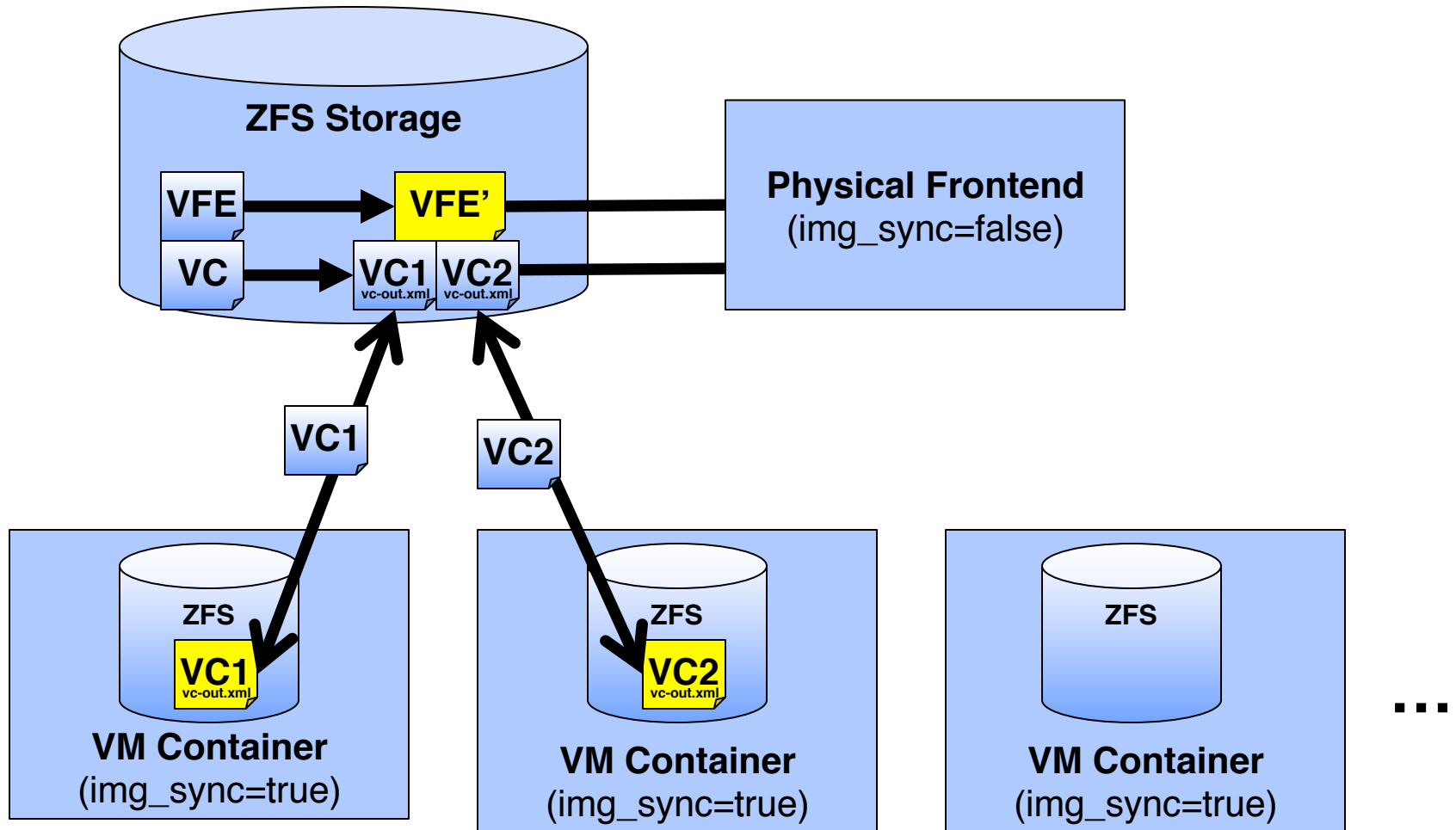
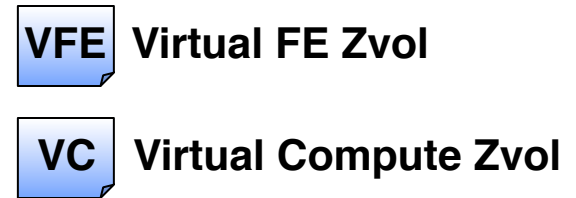
• Available CPUs: 64

• Available GB Memory: 256

• Deployment type: Rocks KVM

• ENT-enabled: no

• Site hostname: lima-7.sdsc.optiputer.net



# pragma\_boot process (rocks) for ZFS images

## Step 1: fix\_images

- No action

## Step 2: allocate

- Finds unused public IP and vlan and # of compute nodes
- Creates cluster (rocks add cluster”) and vc-out.xml
- **Clones** frontend zvol and makes N clones of compute zvol on NAS

## Step 3: Boot FE

- Temporarily mounts zvol to physical FE and installs FE vc-out.xml snippet
- Boots VM from NAS (img\_sync=false)

## Step 4: Boots compute(s)

- Temporarily mounts zvol to physical FE and installs compute vc-out.xml snippet
- Zvol is **syncned** to remote ZFS pool on VM Container and booted (img\_sync=true)
- Zvol is regularly **snapshot** and **syncned** between pools

**Takes about ~8 minutes to boot basic 2-node rocks cluster**

# Describing ZFS volumes

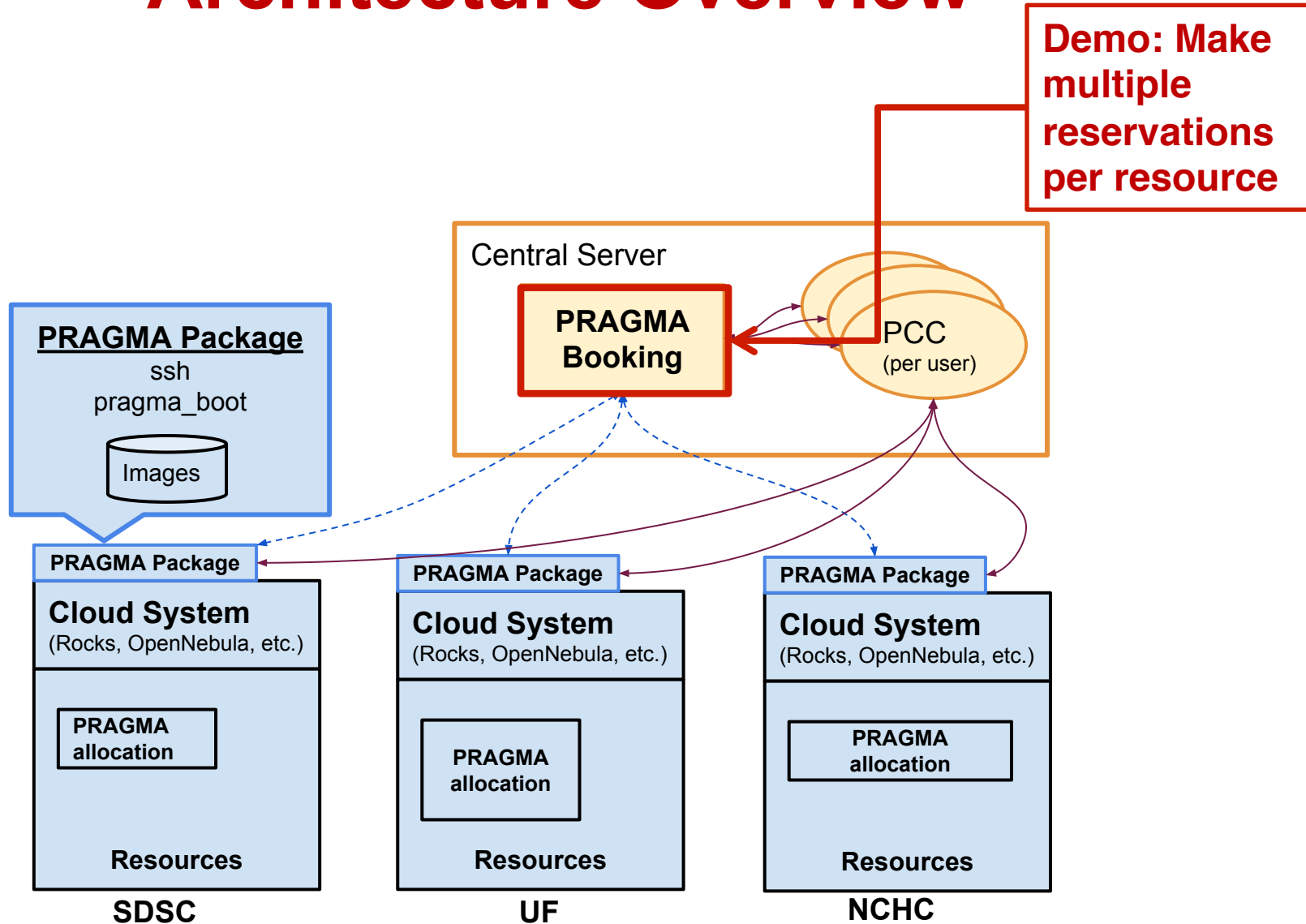
X

- **Repository**

- vcdb.txt
- <vcdir>
  - vc-in.xml  
(based on libvirt XML syntax)
  - <FE image file>
  - <compute image file>
- <vcdir>
  - **vc-in.xml**

```
<vc type='Local Beowulf'>
  ...
  <frontend> ...
    <devices> ...
      <disk type='volume' device='disk'>
        <source volume='pragma/lima-vc-3-vol'
          pool='pragma' host='nas-0-0' />
        <target dev='vda' bus='virtio' />
      </disk> ...
    </devices>
  </domain>
</frontend>
<compute> ...
  <devices> ...
    <disk type='volume' device='disk'>
      <source volume='pragma/vm-lima-vc-3-1-vol'
        pool='pragma' host='nas-0-0' />
      <target dev='vda' bus='virtio' />
    </disk> ...
  </devices>
</domain>
</compute>
  ...
</vc>
```

# Architecture Overview



# PRAGMA Booking

## Pros:

- ✓ Open source
- ✓ Easy to setup
- ✓ Nice GUI interface
- ✓ Usage reporting
- ✓ REST API
- ✓ Customizable
- ✓ LDAP and Active Directory support.
- ✓ Fine tuned roles and permissions.
- ✓ User and group quotas.

<http://www.bookedscheduler.com>

The screenshot shows the homepage of the Booked Scheduler website. At the top, there's a navigation bar with links: Features, What's New, Help, Live Demo, Free Trial, and Download. The main header area has a blue sky background with clouds. On the left, it says "booked" with a logo. In the center, the text reads "A Simply Powerful, Reserve Anything Scheduler". Below this is a "Try It For Free" button with the subtext "GET STARTED IN SECONDS". On the right, there are five feature icons with text: "Book Anything" (calendar icon), "Track Everything" (bar chart icon), "Securely" (lock icon), "In the Cloud" (cloud icon), and "With a Ton of Options" (gear icon). Below the header, there's a section titled "Why Host?" with a paragraph about turn-key hosting. It lists "Unlimited users, schedules, resources for just \$10 per month." and then compares benefits for "Schedule Administrators" (on-site installation, no technical background needed, configure how you want, up and running in 5 minutes) versus "System Administrators" (nothing to install, zero maintenance, professional support, keep your domain name).

## Cons:

- ❖ Can only handle one reservation per resource at a time
- ❖ PHP changes can be painful (heavy OO makes it hard to find right files)
- ❖ Doc is sparse



# Cloud Scheduler

[Dashboard](#)[My Account](#)[Schedule](#)[App](#)

Signed in as Shava  
[Sign Out](#)

[Help](#)

Filter resources by attributes

View reservation summary by scrolling over reservation (in blue).

Mauricio Tsugawa  
2015-04-03 04:00 - 2015-08-01 06:00  
WA-DOCK Demo  
Resources (1): SDSC cloud1  
----- PRAGMA Cloud Scheduler Update @ 2015-04-02 12:25:02.907085 -----

Your resource reservation is being started. You will receive an email when the resources are ready for you to login.

----- PRAGMA Cloud Scheduler Update @ 2015-04-02 13:50:04.634613 -----

Your resource...

CPU: 12  
Memory (Gb/host): 8  
ENT-enabled: no  
VC Name: wa-dock

2015-05-09 →

My Reservation Starting VMs Running VMs Stopping VMs

	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Sunday, 2015-05-03																
SDSC cloud1																
SDSC cloud2																
UF cloud																
Monday, 2015-05-04																
SDSC cloud1																
SDSC cloud2																
UF cloud																
Tuesday, 2015-05-05																

## SDSC cloud2

SDSC

Description Rocks 6.2 KVM. Hosting Virtual clusters and virtual machines  
Notes (no notes)  
Contact (no contact information)  
Location UCSD/SDSC  
Resource Type (no resource type set)

- Available CPUs (total): 12
- Available Memory (Gb): 32
- Deployment type: Rocks KVM
- ENT-enabled: no
- Site hostname: calit2-119-121.ucsd.edu

View resource details by scrolling over resource name

Make a new reservation by clicking on an available time slot.

## Create a new reservation

Shava Smullen (ssmullen@sdsc.edu)

Resources to be reserved

Accessories Add

[SDSC cloud2](#) [More Resources](#)

Begin 2015-04-01 11:00

End 2015-04-01 12:00

Reservation Length 0 days, 1 hours

Repeat Does Not Repeat

Title of reservation

PRAGMA 28 Demo

Description of reservation

Additional Attributes

CPU: 0

Memory (Gb/host): 8

Deployment type: ROCKS KVM

ENT-enabled: no VC Name: wa-dock

Create

Cancel

# Changes to Booked

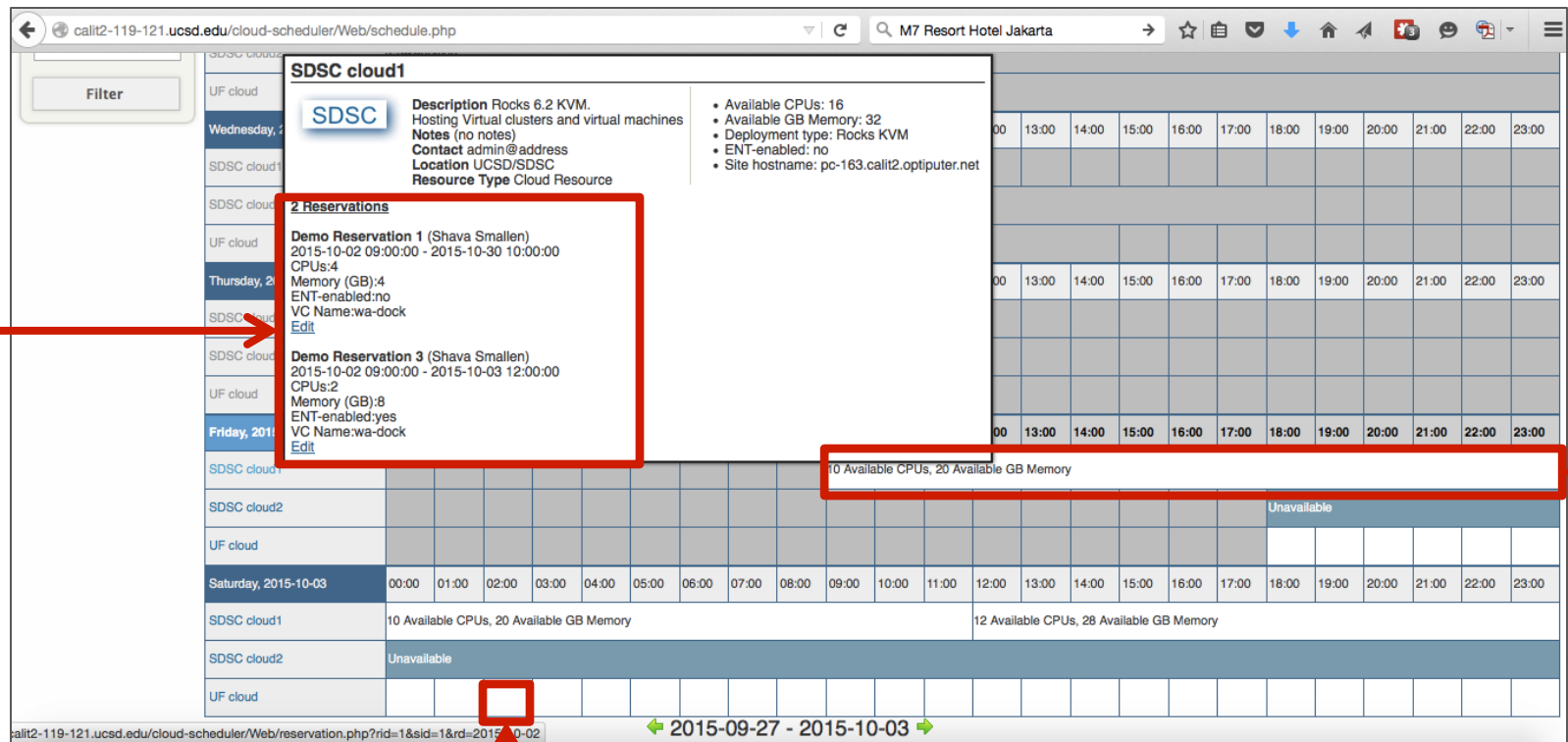
X

- **Reservations stored in MySQL database**
  - No backend changes needed
- **PHP object changes**
  - **Domain/Values/CustomAttributes.php**: Get capacity left based on existing reservations in hour time slot
  - **Domain/Values/CustomAttributes.php**: Find overlapping reservations and calculate capacity for every hour time slot
  - **lib/Application/Schedule/ScheduleReservationList.php**: Read data from db and create slots
  - **lib/Application/Schedule/ReservationSlot.php**: Modified to take array of reservations instead of single value
- **Schedule display**
  - **Pages/SchedulePage.php**: Get capacity left info for a time slot
  - **tpl/Schedule/schedule.tpl**: Display of Schedule pages
- **Add reservation info to resource detail popup**
  - **Pages/Ajax/ResourceDetailsPage.php**: Resource detail popup
  - **tpl/Ajax/resourcedetails.tpl**: Display active reservations in popup
- **Error checking**
  - **lib/Application/Reservation/Validation/CustomAttributeValidationRule.php**: Verifies capacity of new reservation with overlapping reservations



View reservations  
by mousing over  
resource name

2 reservations already  
10 CPUs, 20GB memory still available



Available (empty)

Not available (full)

Create a new reservation

Shava Smallen (ss...)

Resources to be reserved

[SDSC cloud1](#) [More...](#)

Begin 2015-10-20

End 2015-10-20

Reservation Length

Repeat Does Not Repeat

Title of reservation

Reservation 4

Description of reservation

Additional Attributes

CPU: 8


Memory (GB): 4

ENT-enabled: no

VC Name: wa-dock


Create

Cancel



### Your reservation could not be made

- Reservation exceeds available capacity on resource SDSC cloud1 (6 CPUs, 24 memory) from 2015-10-20 00:00:00 America/Los\_Angeles - 2015-10-20 01:00:00 America/Los\_Angeles



Create

Cancel

# Future Work

- Integrate IPOP and PRAGMA-ENT
- Rework PCC and integrate HTCondor so it's used in personal mode
- Leverage CloudFront option in pragma\_boot to manage application virtual cluster images and staging them to each of the sites.
- Package and document for easy installation at sites