

Extending Functionality with Initializers

Huston Rogers
Computer Specialist II
High Performance Computing Collaboratory
Mississippi State University



Overview

- Setup
- Definitions
- Examples



Setup

<https://github.com/ubccr/hpc-toolset-tutorial/>

```
$ git clone https://github.com/ubccr/hpc-toolset-tutorial.git  
$ cd hpc-toolset-tutorial  
$ ./hpcts start
```

Definition of initializers

`/etc/ood/config/apps/dashboard/initializers/ood.rb`

- customize fields and behavior on a per-user basis
- built-in mechanism for adding or changing base functions
- querying associated cluster information
 - reduce user typo rate

Storage Locations

- Files App has an associated ruby variable:
OodFilesApp.candidate_favorite_paths
- By default only contains \${HOME}
- Requires mounting other filesystems on the OOD host

Storage Locations:_INITIALIZER Example

```
#Function for adding other fs to list
Rails.application.config.after_initialize do
  OodFilesApp.candidate_favorite_paths.tap do |paths|
    # Hash of base paths to check for additional directories with titles
    base_paths = ['/work','/scratch','/work2']

    base_paths.each do |base_path|
      # Check if the base path exists and is a directory, to avoid error
      next unless Dir.exist?(base_path)
      if File.readable?(base_path) && File.executable?(base_path)
        paths << FavoritePath.new(base_path)
      end
    end
  end
end
end
```

List of New Directories

Error Checking: Is a Directory
And our User has access

Getting User Info

- can run/get information for a user
 - slurm/cluster association
 - quota information*
 - resource balance information*
- stored in a variable/data structure available in other apps

*(if your system doesn't use the json format built into OOD)

Getting User Information: Slurm Example

- Ruby module Open3 for making system calls

```
sinfo_cmd = "/opt/slurm/bin/sacctmgr show User $USER --associations format=account,qos --parsable2 --noheader"  
o, e, s = Open3.capture3(sinfo_cmd)
```

- Parse for OOD use

```
o.each_line do |v|  
  full = v.split("|")  
  @qos=full[1].split(",")  
  @qos_all.unshift(*@qos)  
  @@accounts.append(full[0].strip())  
end
```

- Pass into form.yml.erb

```
custom_partition:  
  label: "Partition (--partition=value)"  
  required: true  
  widget: select  
  options:  
    <%- CustomPartitions.partitions.each do |a| %>  
    - [ "<%= a.strip %>", "<%= a %>" ]  
    <%- end -%>  
  cacheable: false
```


Submission Form Result

Partition (--partition=value)

ptolemy

ptolemy

gpu-a100

development

Questions? Ready for Hands on?

<https://github.com/huston-rogers/ondemand-initializers>