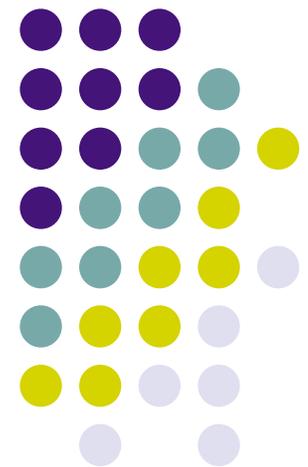
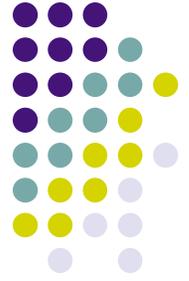


Workflow Developer Training

Getting Started with LWWE & NED

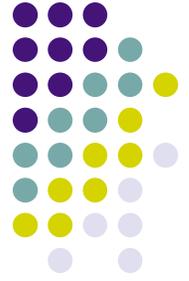
NASA SIVO Code 610.3
October 2009





Outline

- Review of Workflow System Concepts
- Example Workflow
- GEOS-5 Workflow
- Hands-on Example (simple)
- Troubleshooting
- Resources

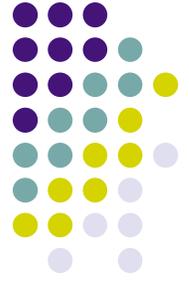


Basic LWWE Terms

LWWE - the workflow engine that controls the execution of workflow tasks

LWWE Components

- **workflowArchitecture.xml** – defines the workflow structure
 - Contains the workflow name, description, modification information, status and workflow task structure and information in XML format.
- **ChildTask** – an element in workflowArchitecture.xml
 - Can be a parent of other tasks or a just a single task.
 - Tasks have a type and can also contain system executable statements.
 - Tasks can define dependencies on other tasks.



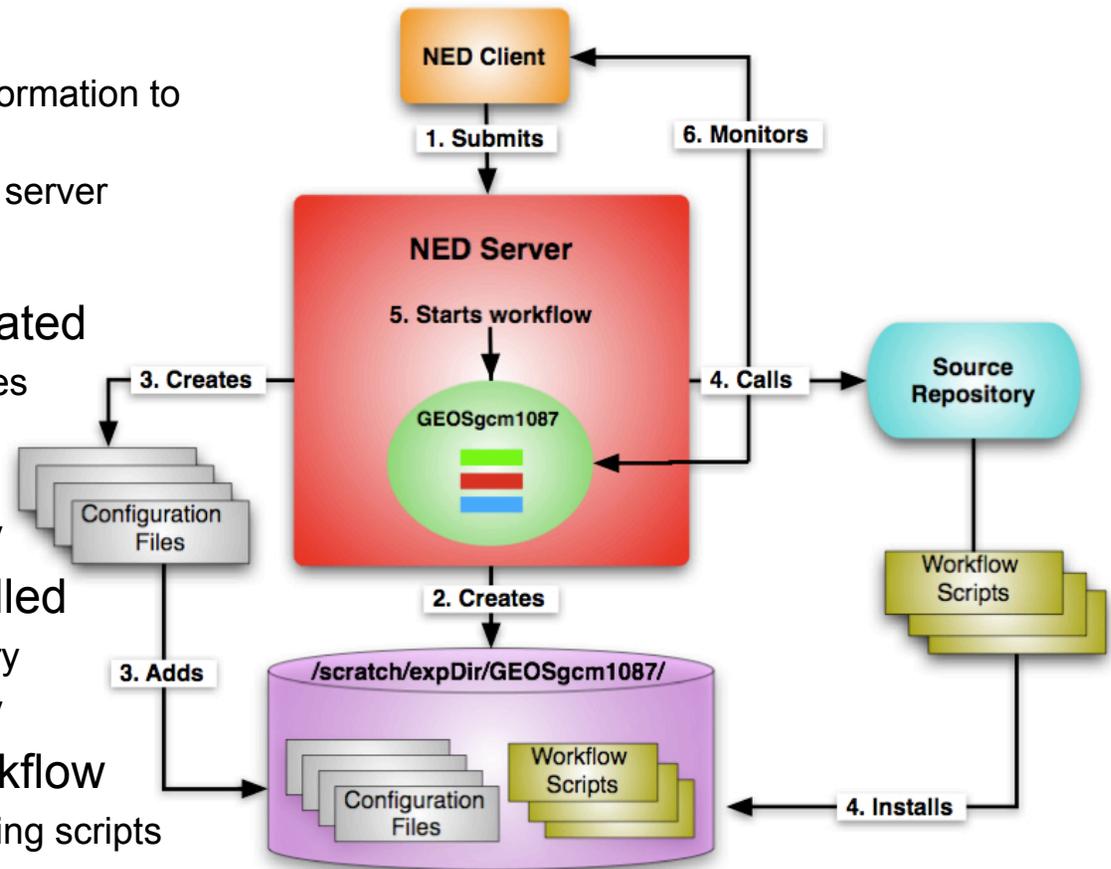
Files Making up a Workflow

- **Architecture definition** –
an XML file that contains the overall structure of workflow tasks and their dependencies
- **Workflow Scripts** –
scripts that perform actions for workflow tasks
- **Configuration files (.nar/.ned)** –
NED configuration files that contain the experiment metadata for a workflow (.nar is a compressed .ned file)

Running a LWWE Workflow in NED



1. User submits workflow
 - NED client sends workflow information to NED server
 - Validation is performed by the server
2. Install directory is created
3. Configuration files are created
 - Define the experiment variables
 - Generated according to the configuration settings
 - Added to the **install directory**
4. Workflow scripts are installed
 - Pulled from a source repository
 - Added to the **install directory**
5. NED server starts the workflow
 - Tasks execute, potentially calling scripts
6. Monitoring occurs
 - Users monitors and controls their workflow



Example Workflow



NED 1.6 User: mrdamon Mode: USER

File Search Tools Visualization & Analysis Run Views Help

Configuration Tree

- END_DATE
- JOB_SGMT
- NUM_SUBMIT
- GCS Components
 - HISTORY templat
 - AGCM
 - COUPLED
 - GCMTRAJ
 - MERRA_CFIO
 - Seasonal
 - tavg1_2d_s
 - Resource Bun
 - MERRA_GRAD
 - POWER
 - Visualization setti
 - EMAIL_OPTION
 - Advanced settings

Configuration

Name	Description	Values
format	File format used for output files.	'CFIO'
filename	The output file name.	'tavg1_2d_slv_Nx'
template	Defines the time stamping template used for output files.	'%y4%m2%d2_%h2%n2z.nc4'
mode	Controls whether values are either instantaneous or time averaged.	instantaneous
frequency	Frequency of output in HHHMMSS.	010000
duration	Duration for each file produced (YYYYMMDD).	010000
fields	A table of output fields, components, and aliases used for generating	Click To Edit...

Status

Status	Affects	Description

Current File: /home/workflow/sharedExperiments/geos5gcm_Fortuna_lwwe.nar Logged Into: localhost:8675[no description]

Configuration in NED

LWWE Workflow monitoring in NED

NED 1.6 User: mrdamon Mode: USER

Visualization & Analysis Run Views Help

Configuration geosgcs_2181

- geosgcs_2181
 - Prepare Environment Prepare the Set-up Environment
 - Prepare Workflow Setup Rename the top level of experiment directory and
 - Setup Workflow Environment Set up the workflow environment
 - Setup Model Environment Set up the model environment
 - Prepare Installation Prepare installation including CVS
 - Main Handles the main loop tasks
 - Create Experiment Directories Create the Experiment Directories
 - Create RC Templates Creates the RC Templates
 - Create Live RCS Creates the Live RCS
 - Link BCS Creates the Link BCS
 - Get GCM Restarts Gets the GCM Restarts
 - Run Loop Handles the run loop tasks
 - Submit Job Submits the segment ID
 - Wait In Queue Waits for the segment ID in queue.
 - Run Model Runs the model after getting the segment ID
 - Rename Checkpoint to Restart Renames Checkpoint to Restart
 - Archive Restarts Archives the restarts
 - Move History Moves the History directory or files
 - Post Processing Performs the post process
 - Prepare Next Segment Prepares the next segment
 - Pull Back Data Pulls back data

COMPLETED UPDATED QUEUED UNKNOWN SUSPENDED RUNNING RESUB

Status geosgcs_2181-Run Model.log

```
INFO - of privacy. Unauthorized access or use may subject you to disciplinary
INFO - action and criminal prosecution.
INFO - This U.S. Government resource is for authorized users only. By accessing
INFO - this system you are consenting to complete monitoring with no expectation
INFO - of privacy. Unauthorized access or use may subject you to disciplinary
INFO - action and criminal prosecution.
INFO - This U.S. Government resource is for authorized users only. By accessing
INFO - this system you are consenting to complete monitoring with no expectation
INFO - of privacy. Unauthorized access or use may subject you to disciplinary
INFO - action and criminal prosecution.
```

Current File: /home/workflow/sharedExperiments/geos5gcm_Fortuna_lwwe.nar Logged Into: localhost:8675[no desc]



Example Workflow

Configuration in NED

Name	Description	Values
numSubmits		3
modelResolution	The spatial resolution of the model.	4x5
speciesNames	Names of species concentrations	Click To Edit...
archiveSystem	Name of system to archive data on.	discover01

```
<Value>3</Value>
<DataType>INTEGER</DataType>
<Widget xsi:type="TextBoxWidget" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Type>TEXTBOX</Type>
</Widget>
</Variables>
<Variables>
  <Name>modelResolution</Name>
  <Description>The spatial resolution of the model.</Description>
  <DesignOnly>>false</DesignOnly>
  <Enabled>>false</Enabled>
  <Required>>false</Required>
  <Value>4x5</Value>
  <Value>2x2.5</Value>
  <Value>1x1.25</Value>
  <DataType>STRING</DataType>
  <Widget xsi:type="ComboBoxWidget" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Type>COMBOBOX</Type>
    <selected>0</selected>
    <editable>>false</editable>
  </Widget>
</Variables>
<Variables>
  <Name>speciesNames</Name>
  <Description>Names of species concentrations</Description>
  <DesignOnly>>false</DesignOnly>
  <Enabled>>false</Enabled>
  <Required>>false</Required>
  <Value>C0</Value>
  <Value>true</Value>
  <Value>N03</Value>
  <Value>true</Value>
  <Value>O3</Value>
  <Value>false</Value>
  <DataType>STRING</DataType>
  <Widget xsi:type="CheckBoxListWidget" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Type>CHECKBOX_LIST</Type>
  </Widget>
</Variables>
<Variables>
  <Name>archiveSystem</Name>
  <Description>Name of system to archive data on.</Description>
  <DesignOnly>>false</DesignOnly>
  <Enabled>>false</Enabled>
  <Required>>false</Required>
  <Value>discover01</Value>
  <Value>dali</Value>
  <DataType>STRING</DataType>
  <Widget xsi:type="TextListWidget" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Type>TEXT_LIST</Type>
  </Widget>
</Variables>
```

Configuration in XML

workflowArchitecture.xml



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<WorkflowArchitecture xmlns="LightweightWorkflowEngine">
  <Name>BaseWorkflow</Name>
  <Description>A simple workflow.</Description>
  <Creator></Creator>
  <CreationDate>Sept. 5, 2009</CreationDate>
  <ModifiedBy>mrdaon</ModifiedBy>
  <ModificationDate></ModificationDate>
  <SubmittedBy>mrdaon</SubmittedBy>
  <OverallStatus>Completed</OverallStatus>
  <Version></Version>
  <MD5Checksum></MD5Checksum>
  <ChildTasks>
    <Name>baseWorkflow</Name>
    <Description>Basic workflow.</Description>
    <Information></Information>
    <TaskType>Parent</TaskType>
    <ChildTasks>
      <Name>Parent task</Name>
      <Description>Parent task for children tasks.</Description>
      <Information></Information>
      <TaskType>System</TaskType>
      <ExecutableObject>echo "Running parent task"</ExecutableObject>
      <ChildTasks>
        <Name>Print Environment</Name>
        <Description>Prints out the environment information</Description>
        <Information></Information>
        <TaskType>System</TaskType>
        <ExecutableObject>$NED_WORKING_DIR/bin/ChildTasks.py printOutEnv $NED_WORKING_DIR</ExecutableObject>
      </ChildTasks>
      <ChildTasks>
        <Name>Empty task</Name>
        <Description>A task that doesn't really do anything</Description>
        <Information></Information>
        <TaskType>System</TaskType>
        <ExecutableObject>$NED_WORKING_DIR/bin/ChildTasks.py emptyTask</ExecutableObject>
      </ChildTasks>
    </ChildTasks>
  </ChildTasks>
  <ChildTasks>
    <Name>Looping tasks</Name>
    <Description>Task that loops</Description>
    <Information>Will loop over number of submissions</Information>
    <IterationLimit>BaseWorkflow.Group.numSubmits</IterationLimit>
    <TaskType>Loop</TaskType>
    <ChildTasks>
      <Name>Run model</Name>
      <Description>Loads the ENV into workflow</Description>
      <TaskType>System</TaskType>
      <ExecutableObject>echo "Running a model"</ExecutableObject>
    </ChildTasks>
  </ChildTasks>
</ChildTasks>
</WorkflowArchitecture>
```




GEOS-5 Workflow

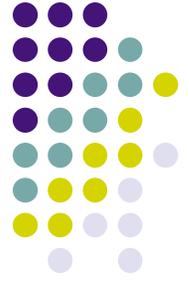
- Access the GEOS-5 workflow scripts from mapme by typing at the prompt:

```
>> svn checkout file:///home/workflow/WorkflowRepository/GEOSGCMdemo/tags/release-1.15
```

- release-1.15/GEOSgcm/bin contains the core GEOS-5 Python scripts:

UserUtilities.py
PullBackData.py
SubmitJob.py
WaitInQueue.py
RunLoopRemoteJob.py
RunModel.py
CreateLinkBCs.py
CreateLiveRCS.py
GetGCMRestarts.py
RunLoop.py
CreateExpDirs.py
CreateRCTemplates.py
CleanEnvironment.py
SetupGroupList.py
RemoteSystemTools.py

XMLFileTools.py
UtilSystem.py
TemplatedInstantiation.py
SubmitPbsJobAndRecord.py
RunScriptOnRemoteSystem.py
SetUpWorkFlowEnv.py
SetUpMain.py
SetUpEnv.py
PostProcess.py
MainTasks.py
SetUpModelEnv.py
PrepareInstallation.py
RemoteJob.py
GEOS5Main.py



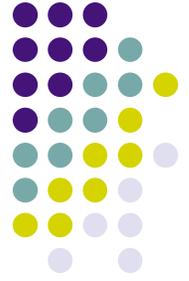
Hands-On Demo

Simple Demo to Modify

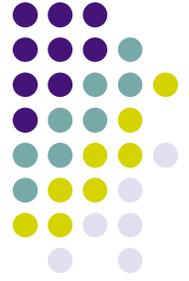
- Check out a demo workflow
- Make a simple modification
- Check it back in
- Run it



Requirements for Running Workflows



- Required for this demo:
 - Some experience with CVS or Subversion
 - Account on mapme
 - Member of wf_dev group on mapme

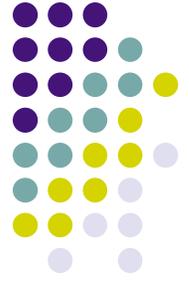


Setup a New Workflow

- Log-in to **mapme @ NCCS**
- First time developers must switch to the “wf_dev” group:
`newgrp wf_dev`
- Create a working directory and *cd* to it.
- Run developer startup script to add command-line shortcuts:
`./home/workflow/bin/script/wfdev.bash`
- Execute the following scripts to create a new “demo” workflow in your working directory

```
branchWorkflow <username>
```

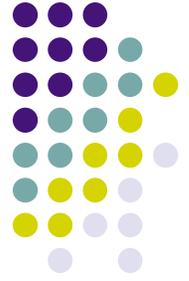
```
checkoutWorkflow <username>
```



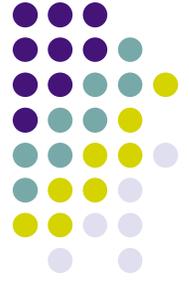
Enhance the Workflow

- *cd to working directory/<username>/bin*
- Bring up NED (type “ned”)
 - Enter in your NCCS User Name, Password, and Passcode and click OK.
 - Select *File -> Open from File*. Click OK.
 - Navigate to *working directory/<username>/config*
 - open *BaseWorkflow.nar* .
 - Go to *File* and check *Design Mode*
 - Under *Configuration Tree* select *Group*. Execute <ctl>right click and navigate to *Add a Variable*.
 - Enter *expName* in the *Information* field of the *Name* property
 - Fill in the other *Information* fields as needed. Click OK.
 - Expand the *Variable Group* and note the variable named *expName* listed with the existing variables.
 - Under *Configuration Tree* select *BaseWorkflow*.
 - Execute <ctl>right click then navigate to *Edit Workflow Properties*.
 - In the *Repository Branch* workflow property enter:
/baseWorkflow/branches/<username>

Enhance the Workflow (cont.)



- Save and run NED workflow
 - Select *File* -> *Save*.
 - Select *Run* -> *Submit Workflow*
 - Use the tab labeled *baseWorkflow_<Unique ID>* and watch the workflow execute.
 - All tasks should turn yellow (completed) within seconds.
 - Right click in the white space where the completed LWWE task tree is shown. Select *Expand Tree*.
 - Right click the task called *Print Environment* under *baseWorkflow_<Unique ID>->baseWorkflow->Parent task* and select *View Task Log*
 - Check the log information in the lower pane for the *expName* variable output.

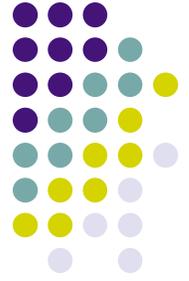


Tips: Testing Workflows

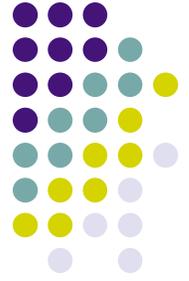
- Write your workflow with testing in mind:
 - Be generous with logging
 - Check for error codes
 - Use Python's exception and error handling
- Run the workflow to test it
 - New workflows get installed and run under:
`/scratch/expWorkDir`
 - Check the install directory when problems occur
 - Use the log files which are viewable from NED
- Test the workflow
 - As you development make sure to test frequently
 - Use multiple user accounts to avoid hard-wired solutions

Tips: Troubleshooting

When a workflow fails...



- Determine where the problem is coming from
 - Files may be missing - did you commit?
 - User script errors
 - Check the log file
 - Edit and re-run the task to see if it passes
 - Remote script errors - must have a log file
- Visit Modeling Guru for questions and answers
 - NASA website:
<http://modelingguru.nasa.gov/clearspace/community/mapmewkflow>
 - SIVO staff regularly monitors Modeling Guru
 - Knowledge base for Earth System modeling
- Contact SIVO workflow staff for more assistance



Other Resources

- **Modeling Guru community**

This points to a number of resources for workflow developers, such as discussion forum that you can use to post questions to us and others.

User tutorials

Workflow group documentation

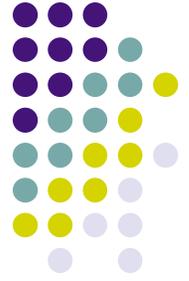
- <https://modelingguru.nasa.gov/clearspace/community/mapmewkflow>

- **LWWE Documentation**

The LWWE software was built by SIVO.

<https://modelingguru.nasa.gov/clearspace/docs/DOC-1683>

BACKUP SLIDES



GEOS-5 workflow

Run.script flowchart

Submit workflow from NED

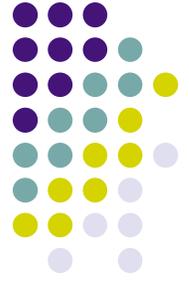
mapme



Steps to integrating the GEOS run scripts



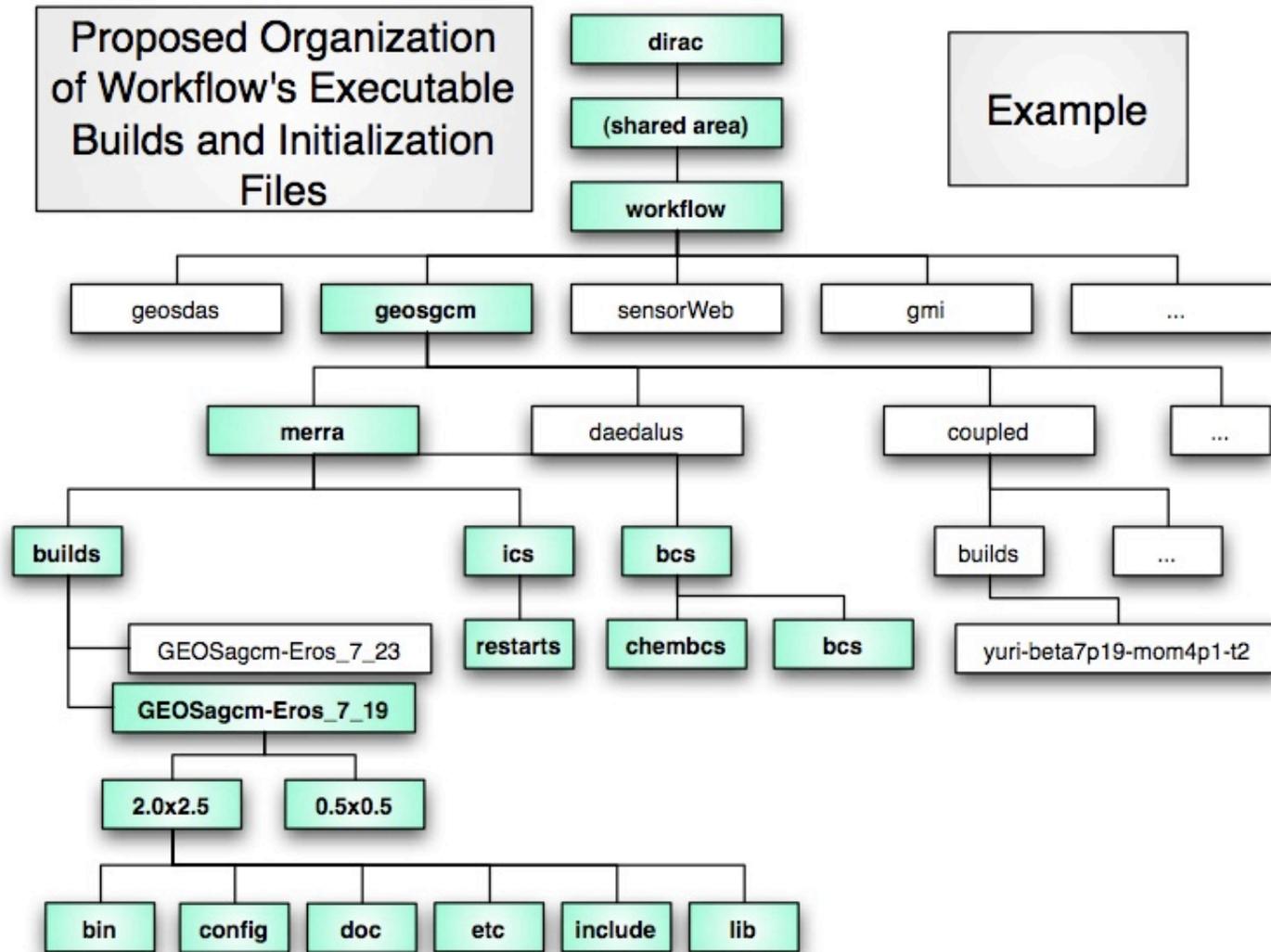
1. GEOS run.script is componentized or decomposed
 - Identify portions of the script that are generally
 - Independent,
 - Perform a basic task
 - Pull out that portion into a new script
2. Associate with SMS
 - New script becomes an SMS task
 - Task is defined in the definition file
 - A .sms file is created that invokes the shell script
3. Finally add variables from the workflow configuration
 - Most scripts will require user-configurable variables
 - Simply source the file generated by NED at the top of the script



Special cases

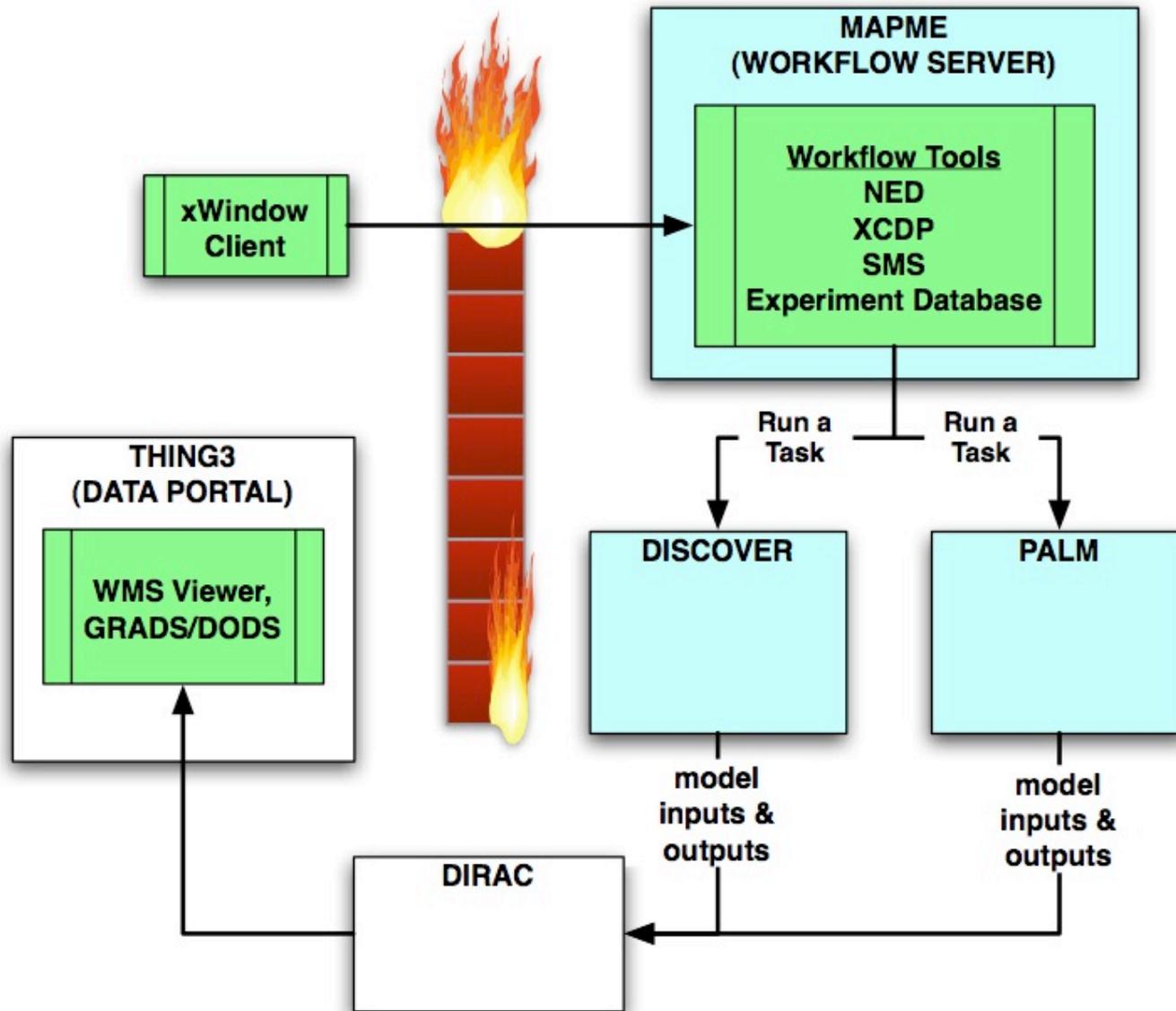
- Scripts that finish running are considered “complete” by SMS
 - So submitting a job to a queue requires you to wait for the job to complete before exiting
- Scripts that get extracted may have variable dependencies
 - Shared variables between scripts can be included
- Breaking down tasks changes how jobs enter the queue
 - Lots of small jobs vs. single large job

Workflow Config Management



Ideally GEOS source and workflow's run scripts should be kept in sync in the same repository

Workflow System in NCCS



- Run from single location within NCCS: “mapme”
- Connects to HPC resources to perform tasks
 - palm
 - discover
 - dirac
 - *Others*

Workflow System in NCCS

