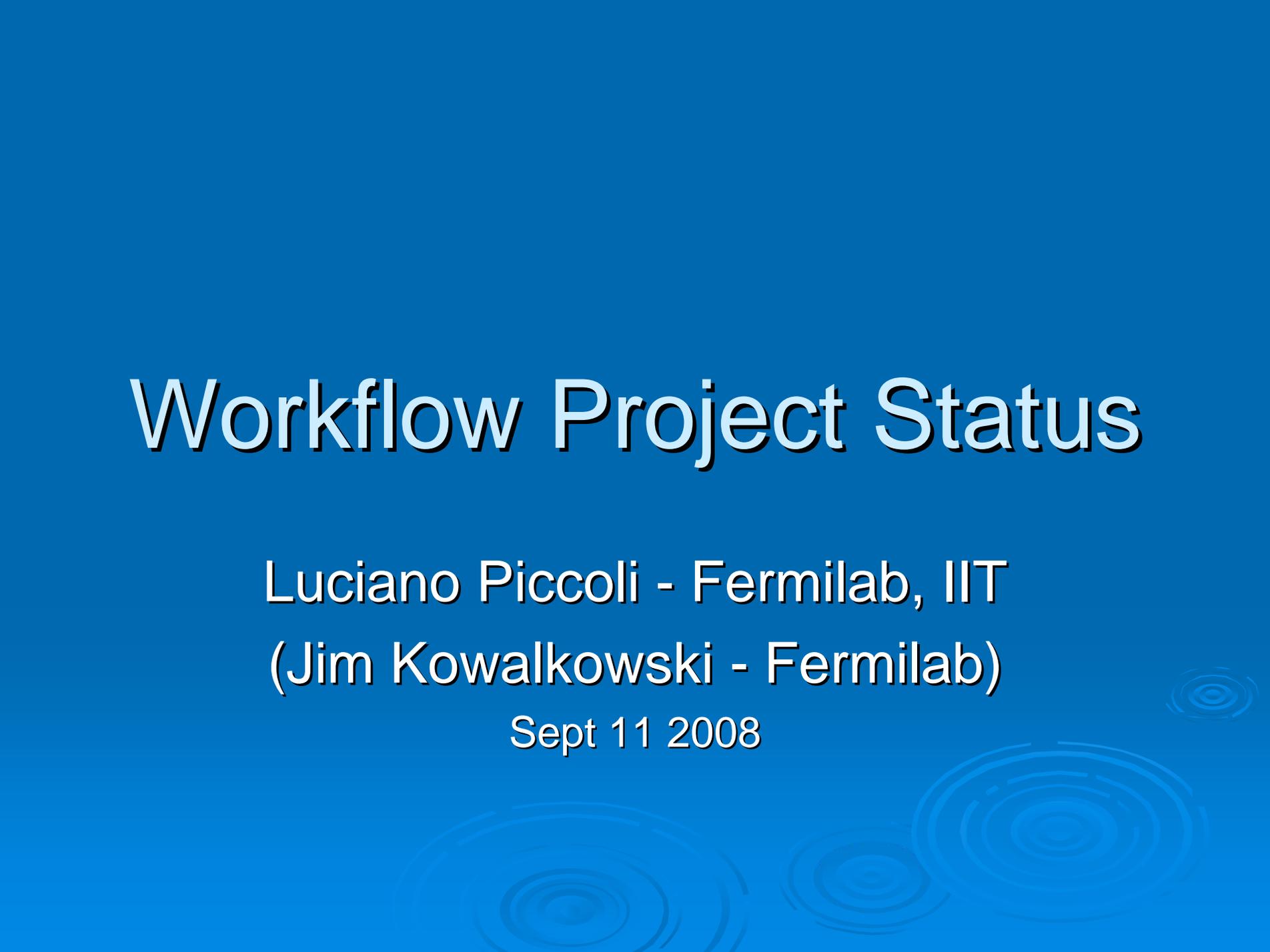


# Workflow Project Status

Luciano Piccoli - Fermilab, IIT  
(Jim Kowalkowski - Fermilab)

Sept 11 2008



# Workflow Reminder

- Workflow as a general concept:
  - From February meeting: A workflow is a reliably repeatable pattern of activity enabled by a systematic organization of resources, defined roles and **mass, energy** and information flows, into a work process that can be documented and learned. Workflows are always designed to achieve processing intents of some sort, such as physical transformation, service provision, or information processing.
  - Another try: The term *workflow* is used in computer programming to capture and develop human to machine interaction. Workflow software aims to provide end users with an easier way to orchestrate or describe complex processing of data in a visual form, much like flow charts, but without the need to understand computers or programming.
- A complex LQCD workflow example: an LQCD analysis campaign to compute hadronic 2- and 3-pt functions for each configuration of a gauge ensemble.

# Outline of Activities

- Attempted to collaborate with Swift team
- Changed focus to solving a specific LQCD processing need
- Development of database driven workflow and result tracking system
- Participation in Lattice 2008
- Participation in eScience 2008 and Super Computing 2008 workshops this fall

# Collaborate with Swift

- Ran into many obstacles: mainly hard to express our processing needs in their language, our earlier report highlights many of the issues
- Wanted changes to the core language
- Talked through really making configuration generation work
- Different priorities
  - Grid computing
  - Bioinformatics

# The Confgen Problem

- Analysis of the Configuration Generation workflow resulted in this document
  - <http://home.fnal.gov/~piccoli/doc/>
  - Test case: MILC asqtad
- Starting point for change of focus
- Workflow systems are only part of the solution (no longer part of the problem!)

# Change of Focus

- Independence of any particular workflow system
- Development of front and back-end systems
  - Parameterization
  - Run time history
  - Provenance
  - Secondary data storage

# Front and Back-End Goals

- Usable and understandable by scientists
  - Domain specific terminology and relationships
  - Capture the process of doing the science
  - Reduce learning curve for conducting complex science
  - Provides a documentation trail for all collaborators
- Workflow composition and management
  - Management of workflow templates
  - Management of participants and their mapping to instances of runnable code
  - Management of parameter sets (physics and system related)
- Scheduling and execution
  - Persistent workflow state
  - Fault tolerance (cluster reliability project)
  - Multiple workflows
  - Workflow and job (two-level) scheduling
- Workflow histories and data provenance
  - Management and user query facilities

# Tools

- Ruby on Rails for data model
  - <http://www.rubyonrails.org/>
  - Integration with relational database (ActiveRecord)
  - Extremely easy to manage web interfaces
  - Simple transformation between formats
    - e.g. XML to YAML to etc...
- OpenWFeru (Ruote) as workflow engine
  - Business Process Model (BPM) engine
  - Implemented in Ruby (<http://www.ruby-lang.org/en/>)
  - Modular and extensible
  - <http://openwferu.rubyforge.org/>

# Confgen Workflow (in Ruote)

```
class TuneProcessDefinition < OpenWFE::ProcessDefinition
  def make
    cursor do
      get_parameters

      _break :if => "${f:error} != none"

      prepare_tuning

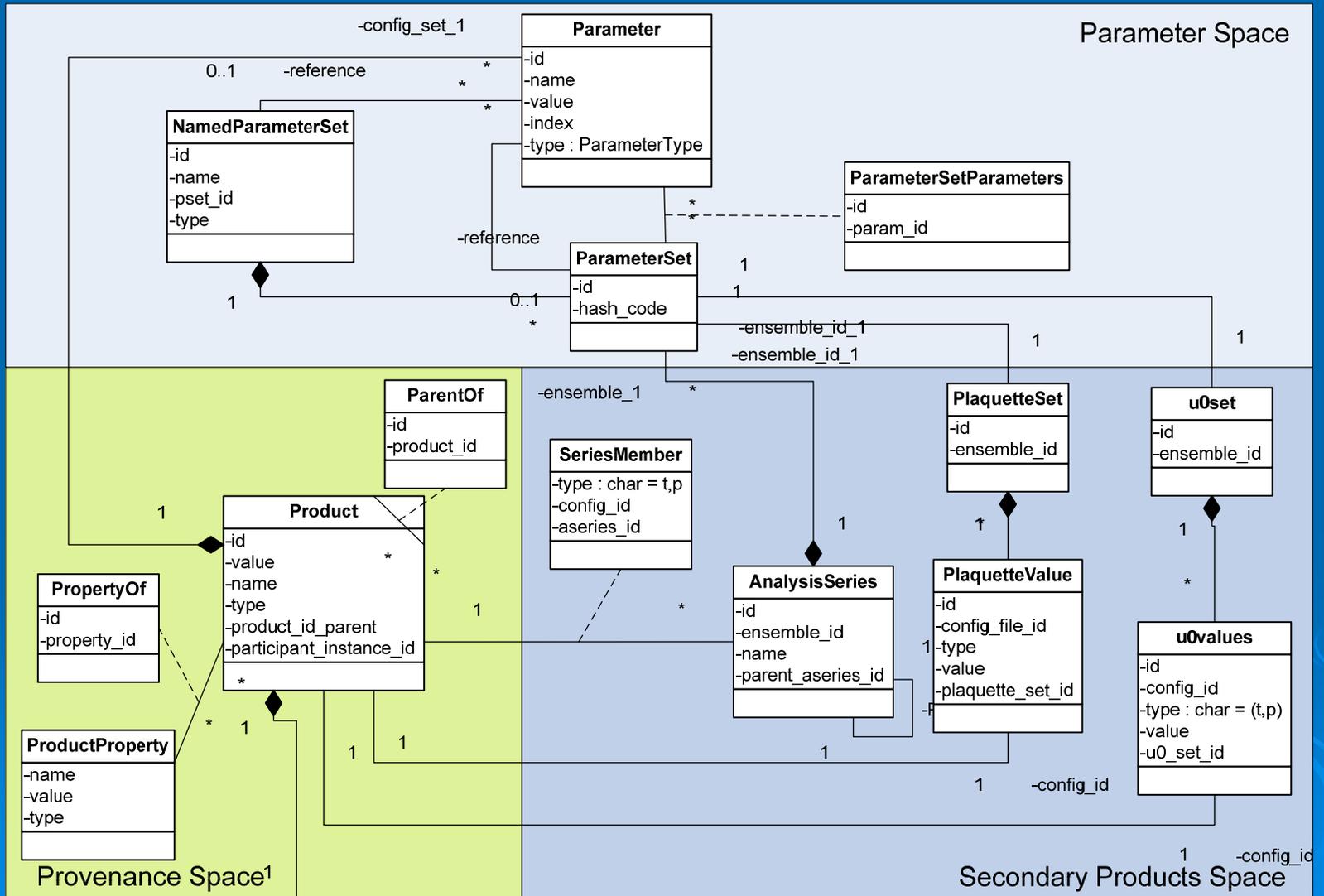
      _break :if => "${f:error} != none"

      _loop do
        tune
        check
        _break :if => "${f:done} == true"
        _break :if => "${f:error} != none"
      end
    end
  end
end
```

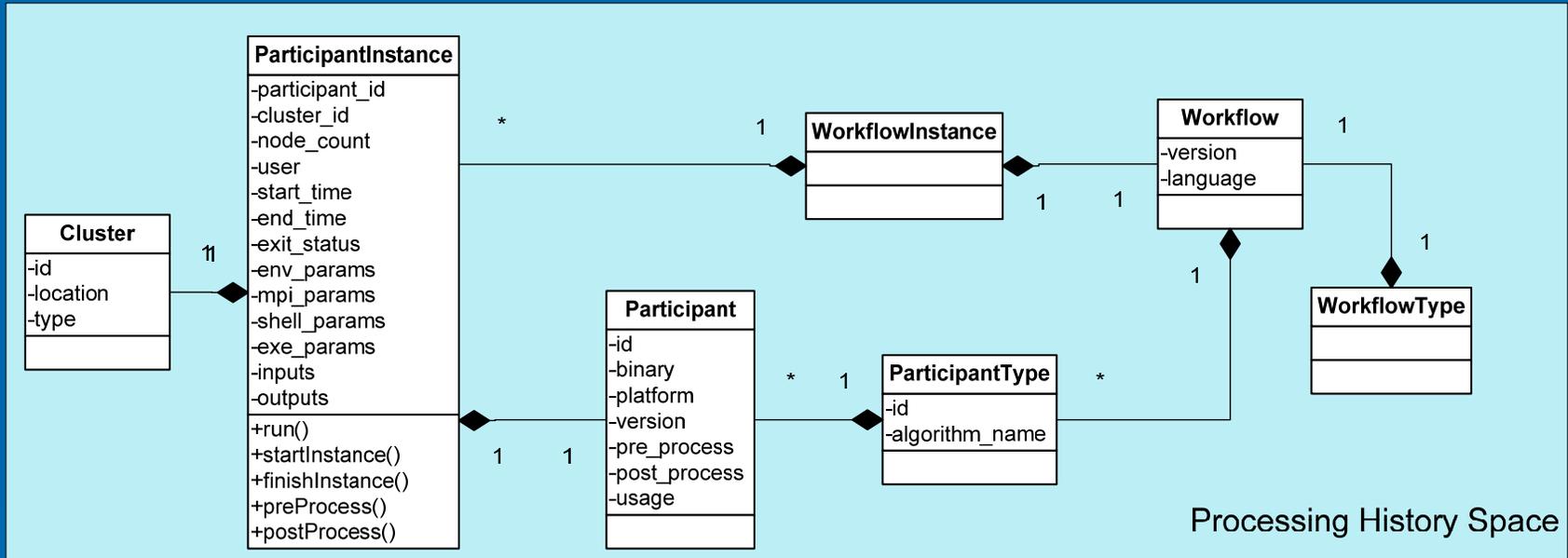
# Data Model

- Parameter Space: Archives all parameters used as input for a workflow, including physics parameters (e.g. quark masses), algorithmic parameters (e.g. convergence criteria) and execution parameters (e.g. number of nodes used).
- Provenance Space: Keeps information regarding inputs and outputs for each workflow participant.
- Secondary Data Space: Used for storing secondary information generated by workflow participants. (e.g. plaquette values from each iteration).
- Run History Space: Archives detailed information about execution, including algorithm versions used and outputs generated.

# Data Model



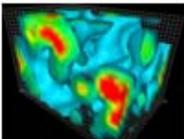
# Data Model



# Web Interface



The screenshot shows a web browser window with the following elements:

- Browser Tab:** LQCD Workflow System
- Address Bar:** http://kaon2:3000/
- Bookmarks Bar:** MyLogBook, Gabi & Lu, Run Coordinator, Telephone, ILC Controls, A0, NOvA DAQ, CRLW Index, Other bookmarks
- Page Content:**
  - LQCD WF System** (Main heading)
  - Based on OpenWFeru** (Sub-heading)
  - 
  - [Configuration Parameter Sets](#)
    - [Participant Instances](#)
    - [Participant Types](#)
    - [Participants](#)
    - [Clusters](#)

# Configuration Sets

## List of Config Parameter Sets

Id	Name(s)	U0 Values	Series	Files	Parameters	Parent Files	Parent Parameters	Workflows	Set Operations	Hash Code
4:	rmd_set	---	tuning	Show	# Show	---	---	<a href="#">Tune</a>	<a href="#">Fork</a> <a href="#">Add Name</a>	62759fa6583db28ad87d70b4fff64246

## Parameter Set [ Names: rmd\_set ; ID 4 ]

ID	Name	Value	Index	Type	Operations	Also Contained By
22	ensemble	1	0	p	<a href="#">Edit</a> , <a href="#">Show</a> , <a href="#">Clone</a>	6
23	parent		0	p	Unavailable	
24	parent_step	-1	0	i	<a href="#">Edit</a> , <a href="#">Replace</a>	
25	seed	5682304				
26	u0	identity				
27	algo_params	2				
28	config_params	3				
29	rationals					
30	file_name_rule	/home/pic				
72	status	ready				

## Parameter Set [ Names: rmd\_ensemble ; ID 1 ]

ID	Name	Value	Index	Type	Operations	Also Contained By
1	nx	8	0	s	<a href="#">Edit</a> , <a href="#">Replace</a>	
2	ny	8	0	s	<a href="#">Edit</a> , <a href="#">Replace</a>	
3	nz	8	0	s	<a href="#">Edit</a> , <a href="#">Replace</a>	
4	nt	8	0	s	<a href="#">Edit</a> , <a href="#">Replace</a>	
5	beta	7.0	0	f	<a href="#">Edit</a> , <a href="#">Replace</a>	
6	nflavors1	4	0	i	<a href="#">Edit</a> , <a href="#">Replace</a>	
7	nflavors2	4	0	i	<a href="#">Edit</a> , <a href="#">Replace</a>	
8	mass1	0.1	0	f	<a href="#">Edit</a> , <a href="#">Replace</a>	
9	mass2	0.1	0	f	<a href="#">Edit</a> , <a href="#">Replace</a>	
10	ensemble_name	l\$nx\$f\$nt\$f\$nflavors1\$f\$nflavors2\$b\$beta\$m\$mass1\$m\$mass2\$	0	s	<a href="#">Edit</a> , <a href="#">Replace</a>	

# Starting a Tuning Workflow

## Tuning workflow parameters (set 4)

Node Count:

Algo Name:

Algo Version:

Cluster Name:

# Running Workflow

## List of Config Parameter Sets

Id	Name(s)	U0 Values	Series	Files	Parameters	Parent Files	Parent Parameters	Workflows	Set Operations	Hash Code
4:	rmd_set	---	tuning	<a href="#">Show</a>	<a href="#"># Show</a>	---	---	<input type="button" value="Tune"/>	<input type="button" value="Fork"/> <input type="button" value="Add Name"/>	62759fa6583db28ad87d70b4fff64246

[Main Page](#)

# Participant Instances

## List of Participant Instances

<u>Id</u>	<u>Node count</u>	<u>User name</u>	<u>Start time</u>	<u>End time</u>	<u>Exit status</u>	<u>Participant</u>	<u>Hash code</u>	
1	2	piccoli	Wed Sep 10 16:28:44 -0500 2008		0	1	RUNNING	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>

[Main Page](#)

## List of Participant Instances

<u>Id</u>	<u>Node count</u>	<u>User name</u>	<u>Start time</u>	<u>End time</u>	<u>Exit status</u>	<u>Participant</u>	<u>Hash code</u>	
1	2	piccoli	Wed Sep 10 16:28:44 -0500 2008	Wed Sep 10 16:30:45 -0500 2008	0	1	68794bf10c3bf39b25ccfffaa3d8ad23	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>
2	2	piccoli	Wed Sep 10 16:30:45 -0500 2008		0	1	RUNNING	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>

[Main Page](#)

## List of Participant Instances

<u>Id</u>	<u>Node count</u>	<u>User name</u>	<u>Start time</u>	<u>End time</u>	<u>Exit status</u>	<u>Participant</u>	<u>Hash code</u>	
1	2	piccoli	Wed Sep 10 16:28:44 -0500 2008	Wed Sep 10 16:30:45 -0500 2008	0	1	68794bf10c3bf39b25ccfffaa3d8ad23	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>
2	2	piccoli	Wed Sep 10 16:30:45 -0500 2008	Wed Sep 10 16:32:46 -0500 2008	0	1	744febb2cf24721c751ae6f3024bf864	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>

[Main Page](#)

# Configuration Files

## List of Config Parameter Sets

Id	Name(s)	U0 Values	Series	Files	Parameters	Parent Files	Parent Parameters	Workflows	Set Operations	Hash Code
4:	rmd_set	Values	tuning	Show	# Show	---	---	<a href="#">Tune</a>	<a href="#">Fork</a> <a href="#">Add Name</a>	62759fa6583db28ad87d70b4fff64246
6:	a-set	Values	a	Show	# Show	4	# 4	<a href="#">Extend</a>	<a href="#">Fork</a> <a href="#">Add Name</a>	8484501258b9033f18cadd0ededb1c4c

## Configuration Files for set 4

ID	file name	series	step	u0 out	config set id
5	/home/piccoli/scratch/l88f44b70m01m01-tuning.1	tuning	1	0.940607694633679	4

## Configuration Files for set 6

ID	file name	series	step	u0 out	config set id
6	/home/piccoli/scratch/l88f44b70m01m01-a.3	a	3	0.8020045025445	6

## U0 Values

Value	Type	Config File
0.940608	T	5
0.802005	P	6

[Back](#)

# 2pt Analysis Workflow

```
class TwoPtHLProcessDefinition < OpenWFE::ProcessDefinition
  cursor do
    get_parameters
    concurrence do
      concurrent_iterator :on_field => "gauge_counter", :to_field => "gauge" do
        concurrent_iterator :on_field => "kappa_counter", :to_field => "kappa" do
          concurrent_iterator :on_field => "wsrc_counter", :to_field => "wsrc" do
            sequence do
              prepare_hq
              _break :if => "${f:error} != none"
              heavy_quark
              _break :if => "${f:error} != none"
              heavy_quark_convert
              _break :if => "${f:error} != none"
            end
          end
        end
      end
    end
    concurrent_iterator :on_field => "gauge_counter", :to_field => "gauge" do
      concurrent_iterator :on_field => "mass_counter", :to_field => "mass" do
        concurrent_iterator :on_field => "wsrc_counter", :to_field => "wsrc" do
          light_quark
          _break :if => "${f:error} != none"
        end
      end
    end
  end
end
...
```

# 2pt Analysis Workflow

```
...
concurrency do
  concurrent_iterator :on_field => "gauge_counter", :to_field => "gauge" do
    concurrent_iterator :on_field => "d1_counter", :to_field => "d1" do
      concurrent_iterator :on_field => "kappa_counter", :to_field => "kappa" do
        onia
        _break :if => "${f:error} != none"
      end
    end
  end
end
concurrent_iterator :on_field => "gauge_counter", :to_field => "gauge" do
  concurrent_iterator :on_field => "d1_counter", :to_field => "d1" do
    concurrent_iterator :on_field => "mass_counter", :to_field => "mass" do
      concurrent_iterator :on_field => "kappa_counter", :to_field => "kappa" do
        bstag
        _break :if => "${f:error} != none"
      end
    end
  end
end
end
end
end
end
end
```

# When will it be released?

- Needs Minimum Fault Tolerance
  - Restart workflow based on database information
  - Robust execution engine with more features (e.g. throttling)
- Account usage designed
- Participant wrapping specification and guidelines
- Resource management
- Improved running status
- Import data from previous runs for analysis

# Conference Participation

- Lattice08
  - <http://omega.cs.iit.edu/~lpiccoli/pub/LAT08-Poster.pdf>
- Paper submitted to SWBES08
  - On LQCD Requirements, Swift and Ascalon discussions
  - Workshop part of e-Science 2008 (December, Indiana)
- Paper will be submitted to WORKS08
  - Workshop at Supercomputing 2008 (November, Austin)

# Future work

## ➤ Workflow Integration

- Data movement and file management
  - Saving intermediate and final data from worker node – currently files generated on shared area.
- Participants independent of workflow system
  - Each workflow system requires participants to use different wrappers
- Integrate database with workflow
  - Import parameters into workflow
- Remote workflow execution
  - Update main database with remote execution results

## ➤ Fault Tolerance

- Resume from last milestone: resume workflow based on information from the provenance database.
  - Coordinate with workflow system

# Future work

## ➤ Provenance

- Data collection
- Reconstruct provenance from database information
  - Able to trace binaries, input and output files.
  - Provenance graph (e.g. DOT files), including participants and data products
- Add workflow to provenance schema
  - Dependent of workflow system
  - Database points to external repository (CVS, SVN)
  - Support for multiple workflow languages?
  - Description on how workflow is invoked
    - Automatic creation of web interface where users select versions of participants to use (workflow instantiation)

# Conclusion

- We started looking for an off-the-shelf workflow product for LQCD, but we need more than that
- Discovered that workflow component is just part of the system
  - Management of files, parameters and other data are part of the new focus
- Is “workflow system” still the right word to describe what we are doing?
  - Add features needed by LQCD
  - Build upon current workflow solutions