



Argonne
NATIONAL
LABORATORY

... for a brighter future

An Automated Component-Based Performance Experiment Environment

*Van Bui, Boyana Norris, and Lois Curfman
McInnes*

Argonne National Laboratory, Argonne, IL.



U.S. Department
of Energy

UChicago ►
Argonne_{LLC}

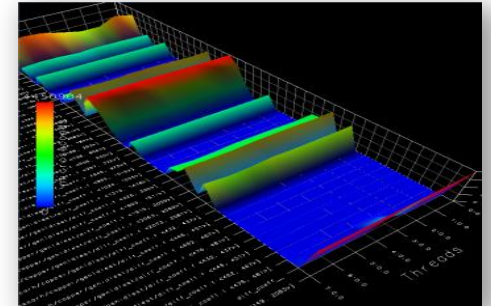
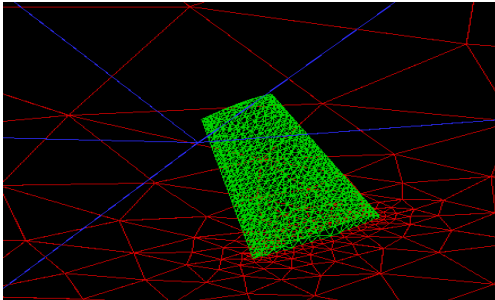


A U.S. Department of Energy laboratory
managed by UChicago Argonne, LLC

CBHPC'09 Nov. 16 2009

Computational Quality of Service (CQoS) Infrastructure

- Uses metadata for describing non-functional properties and requirements, e.g., quality “metrics”
- Supports automated performance instrumentation and monitoring
- Enables offline performance data analysis through machine learning, statistics, etc.



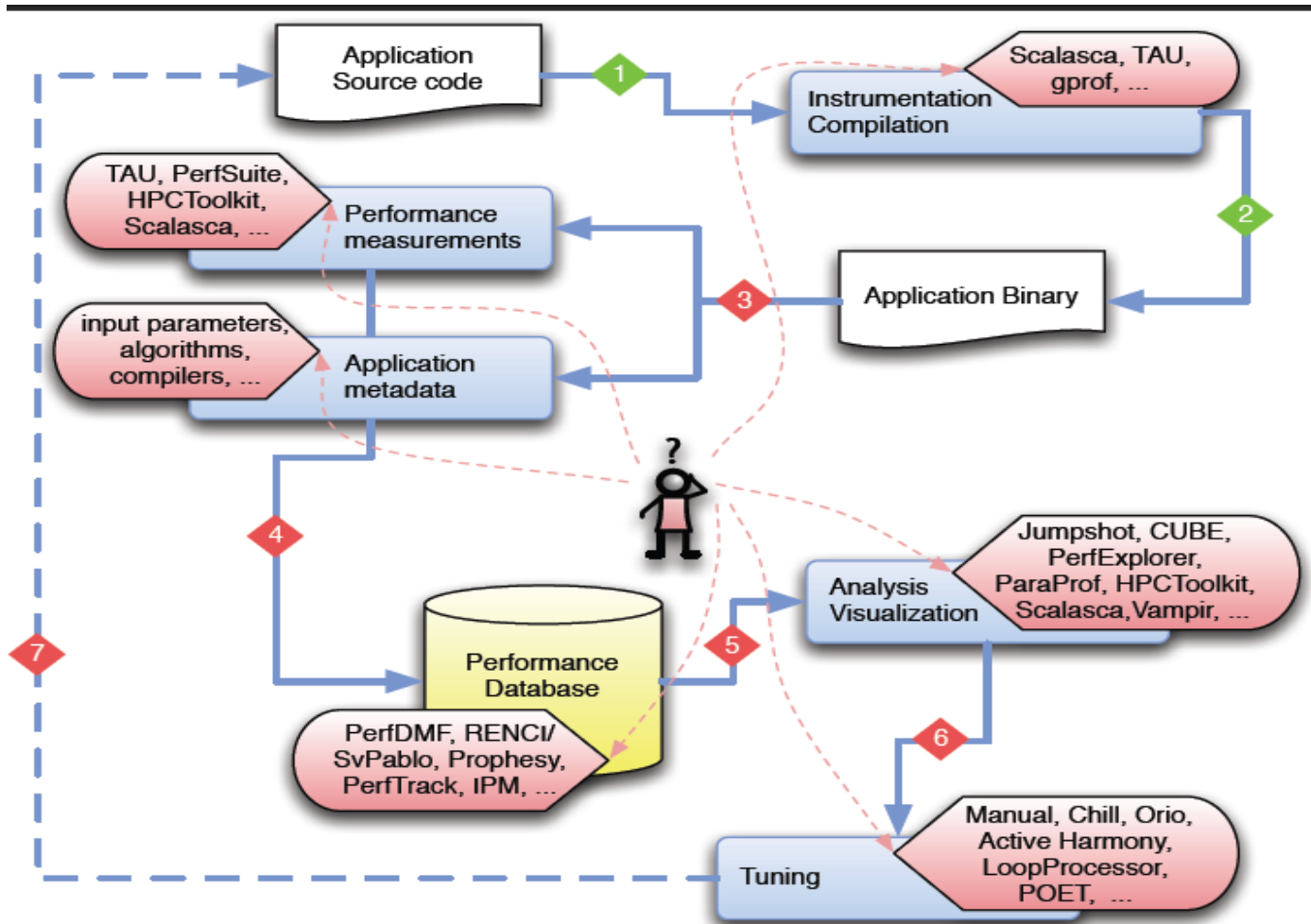
Motivation

- Computational Quality of Service (CQoS) requires support for
 - Performance measurement
 - Performance databases
 - Performance analysis
- Performance analysis can involve running thousands of experiments varying different parameters

Project Goals

- Automate performance experiments as much as possible using a component approach
- Design a uniform interface across platforms, tools, etc...
- Design a portable and extensible tool infrastructure to streamline performance experiments

Performance Experiment Workflow



Performance Components

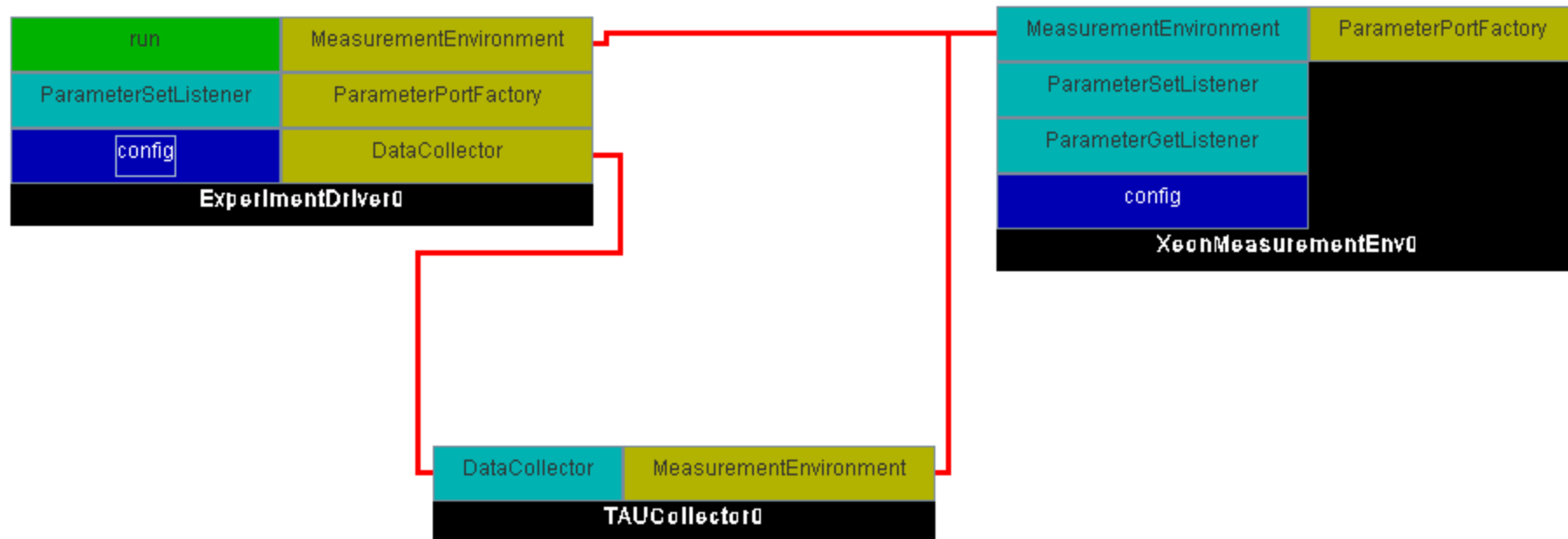
Experiment Setup and Collection

Data Management

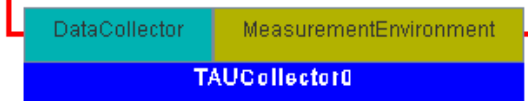
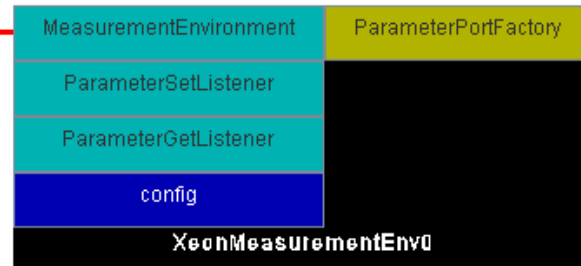
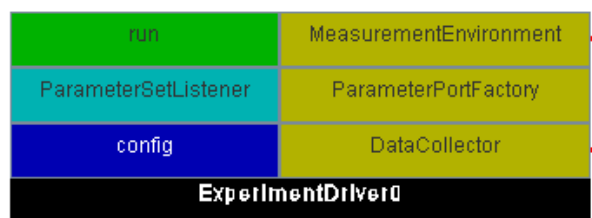
Analysis Phase

Experiment Set-up and Collection

- Configure application, tools, and platform
- Select measurement approach
- Run the application and collect data



Experiment Driver Configuration



Experiment Environment Configuration

Enter the command to execute the program ?

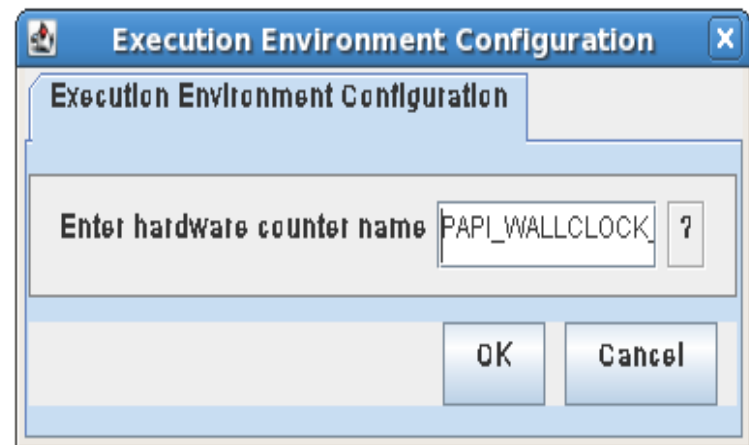
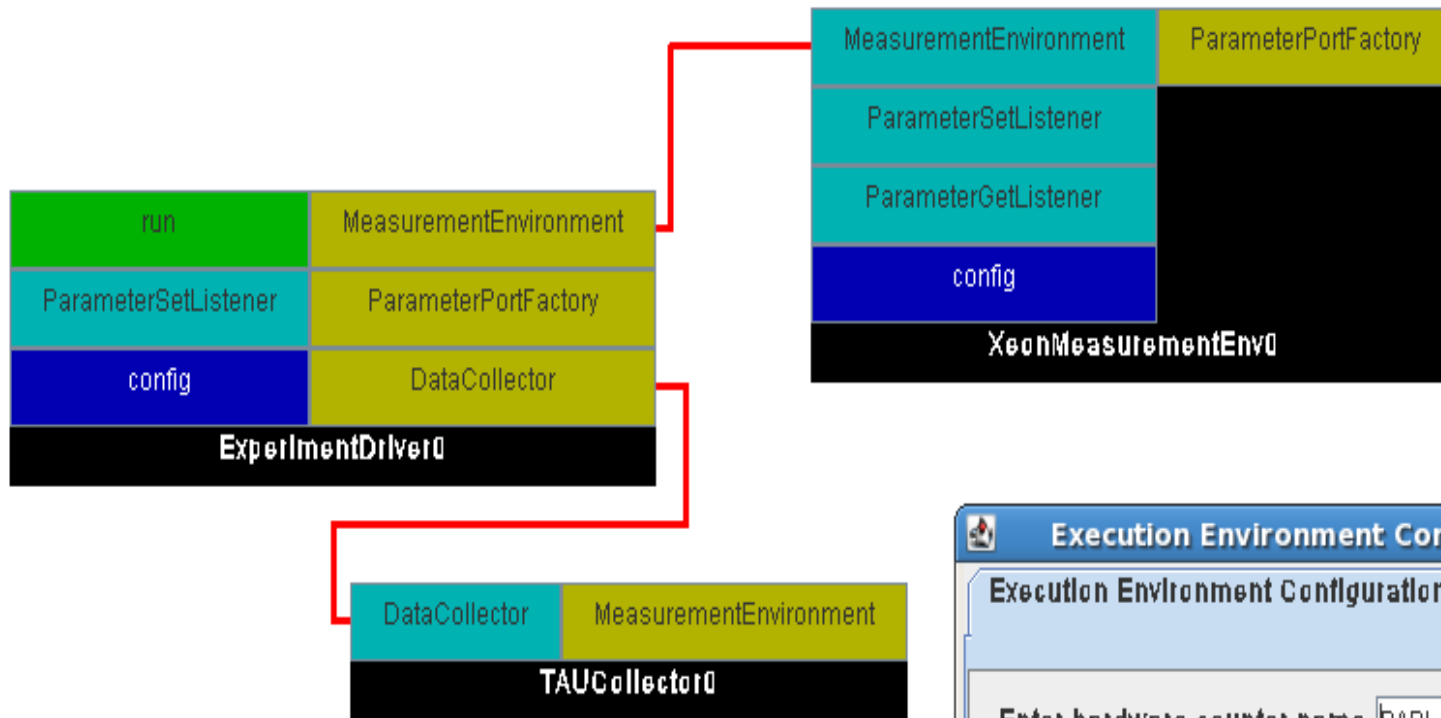
Enter the MPI directory ?

Enter the number of MPI processes ?

Enter the number of OpenMP threads ?

OK Cancel

Measurement Env. Configuration



Performance Components

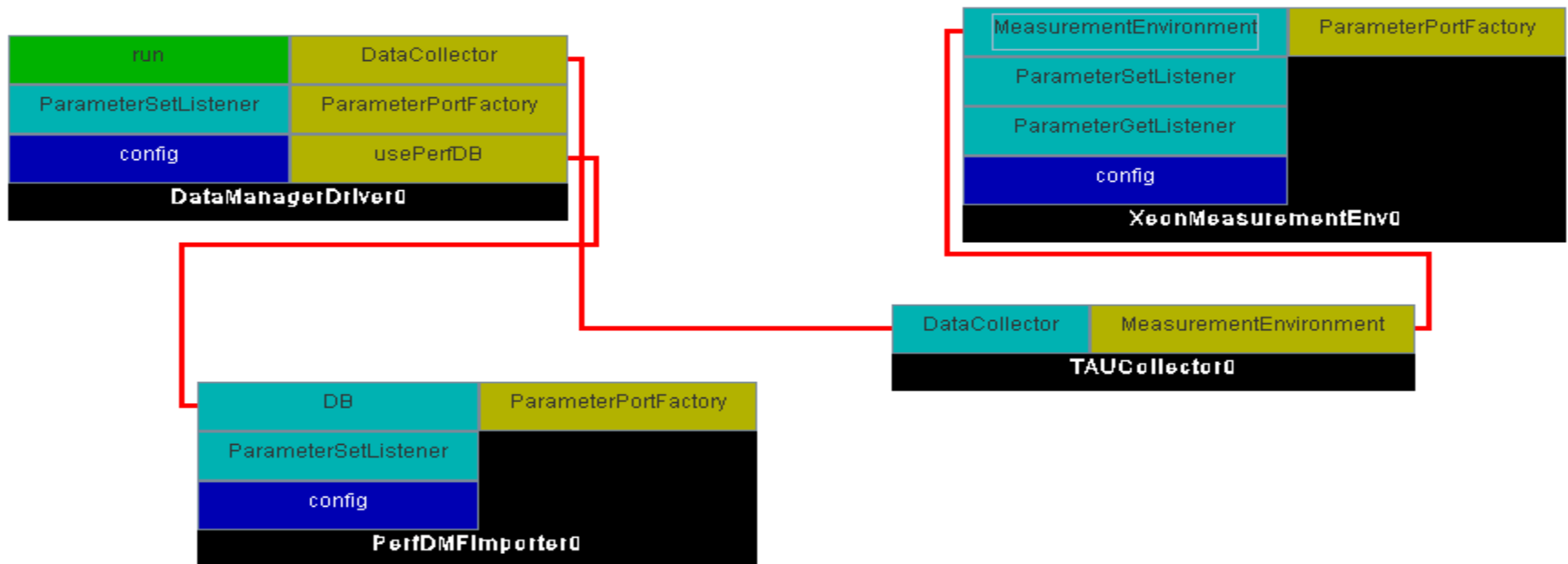
Experiment Setup and Collection

Data Management

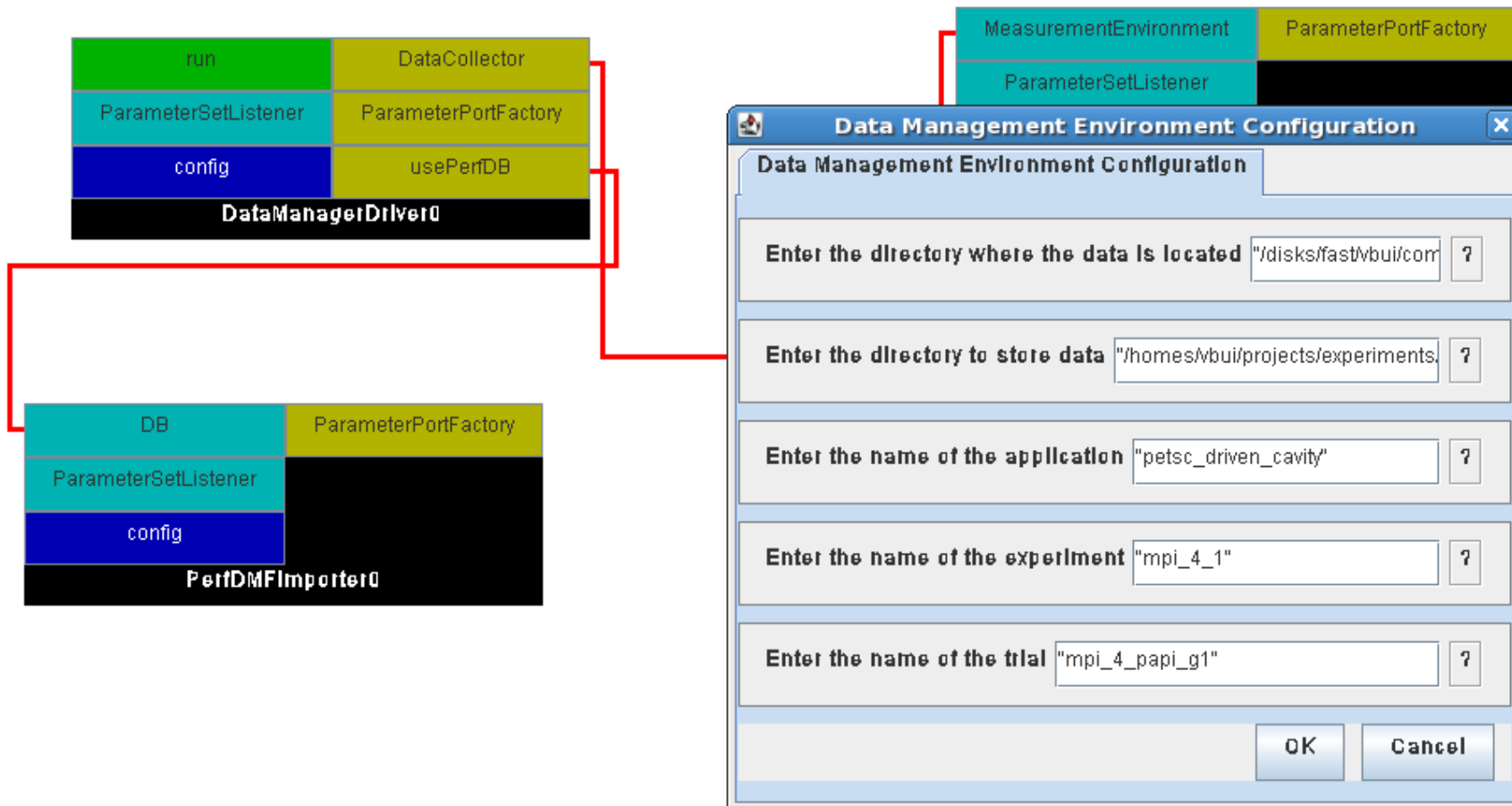
Analysis Phase

Data Management

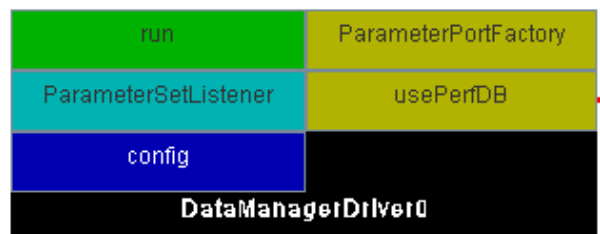
- Prepare performance data for storage
- Store metadata and performance data to database



Data Manager Configuration



Database Configuration



Data Loading Configurations

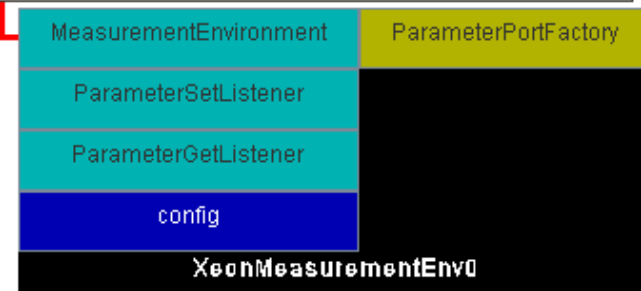
Data Loading Configurations

Enter the name of the CQoS data loader: "CQoSDataLoader_fat" ?

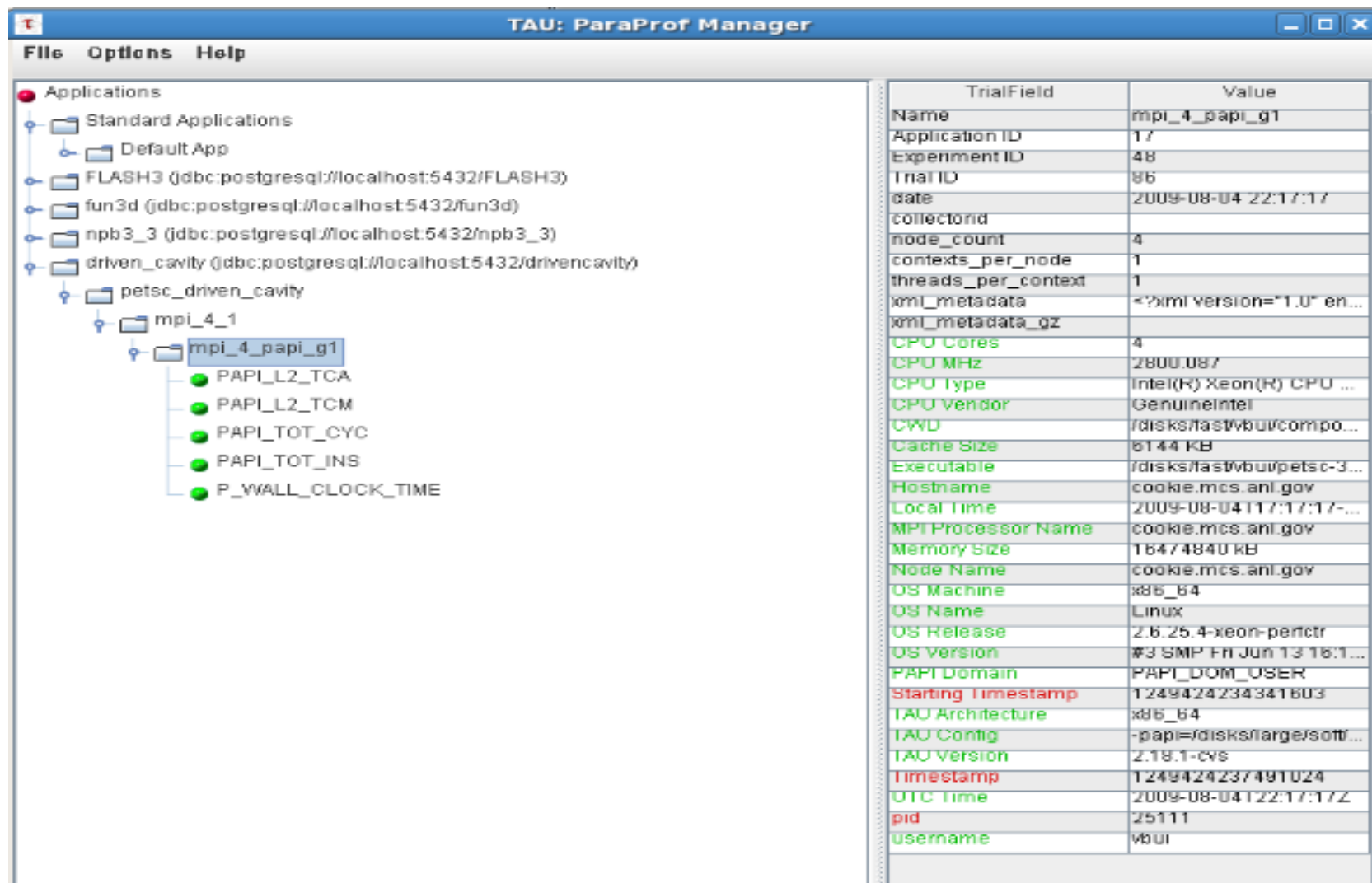
Enter the location of the CQoS data loader: "/homes/vbui/project" ?

Enter the PerfDMF configuration name: "driven_cavity" ?

OK Cancel



Visualize Data Imported



The screenshot displays the TAU: ParaProf Manager interface. On the left, a tree view shows the application hierarchy. The selected application is `mpi_4_papi_g1`, which is a sub-application of `mpi_4_1` under the `petsc_driven_cavity` application. The tree view includes the following items:

- Applications
 - Standard Applications
 - Default App
 - FLASH3 (jdbc:postgresql://localhost:5432/FLASH3)
 - fun3d (jdbc:postgresql://localhost:5432/fun3d)
 - npb3_3 (jdbc:postgresql://localhost:5432/npb3_3)
 - driven_cavity (jdbc:postgresql://localhost:5432/drivencavity)
 - petsc_driven_cavity
 - mpi_4_1
 - mpi_4_papi_g1**
 - PAPI_L2_TCA
 - PAPI_L2_TCM
 - PAPI_TOT_CYC
 - PAPI_TOT_INS
 - P_WALL_CLOCK_TIME

On the right, a table displays the trial fields and their values for the selected application:

TrialField	Value
Name	mpi_4_papi_g1
Application ID	17
Experiment ID	48
Trial ID	86
date	2009-08-04 22:17:17
collectorid	
node_count	4
contexts_per_node	1
threads_per_context	1
xml_metadata	<?xml version="1.0" en...
xml_metadata_gz	
CPU Cores	4
CPU MHz	2800.087
CPU Type	Intel(R) Xeon(R) CPU ...
CPU Vendor	GenuineIntel
CWD	/disks/fast/vbui/compo...
Cache Size	6144 KB
Executable	/disks/fast/vbui/petsc-3...
Hostname	cookie.mcs.anl.gov
Local Time	2009-08-04 17:17:17-...
MPI Processor Name	cookie.mcs.anl.gov
Memory Size	16474840 KB
Node Name	cookie.mcs.anl.gov
OS Machine	x86_64
OS Name	Linux
OS Release	2.6.25.4-xeon-perfctr
OS Version	#3 SMP Fri Jun 13 16:1...
PAPI Domain	PAPI_DOM_USER
Starting Timestamp	1249424234341603
TAU Architecture	x86_64
TAU Config	-papi=/disks/large/soft/...
TAU Version	2.18.1-cvs
Timestamp	1249424237491024
UTC Time	2009-08-04 12:17:17Z
pid	25111
username	vbui

Performance Components

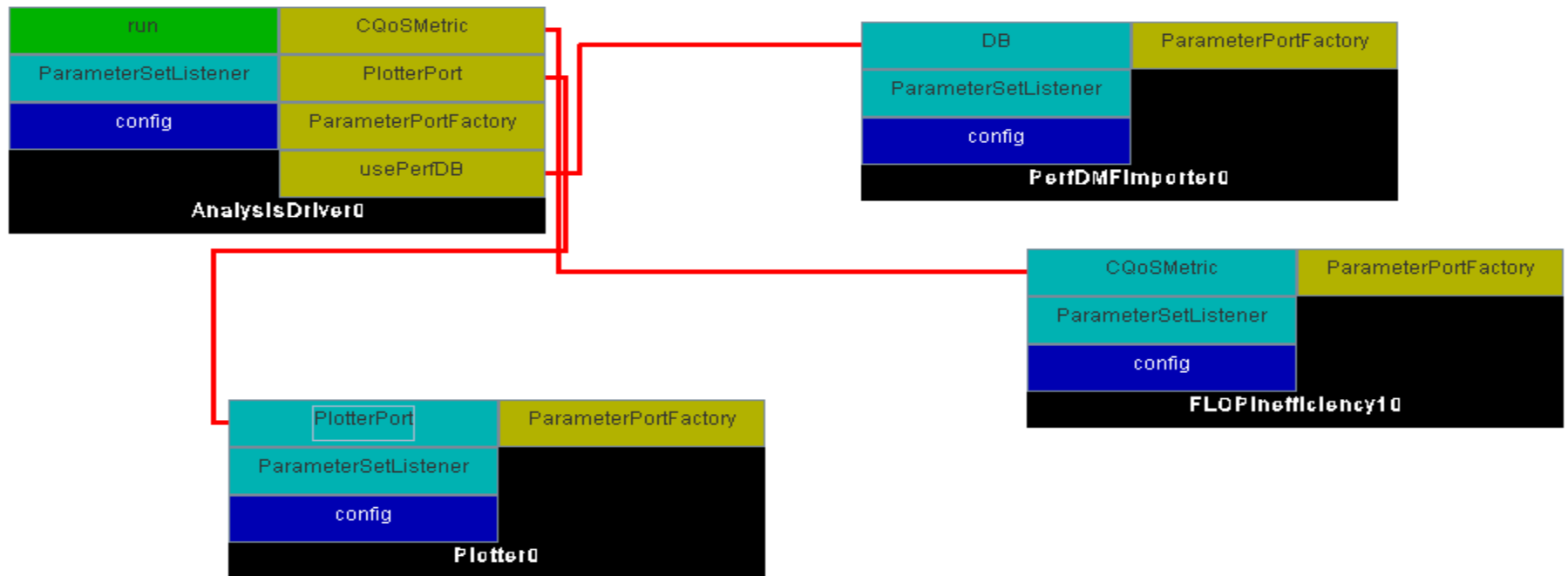
Experiment Setup and Collection

Data Management

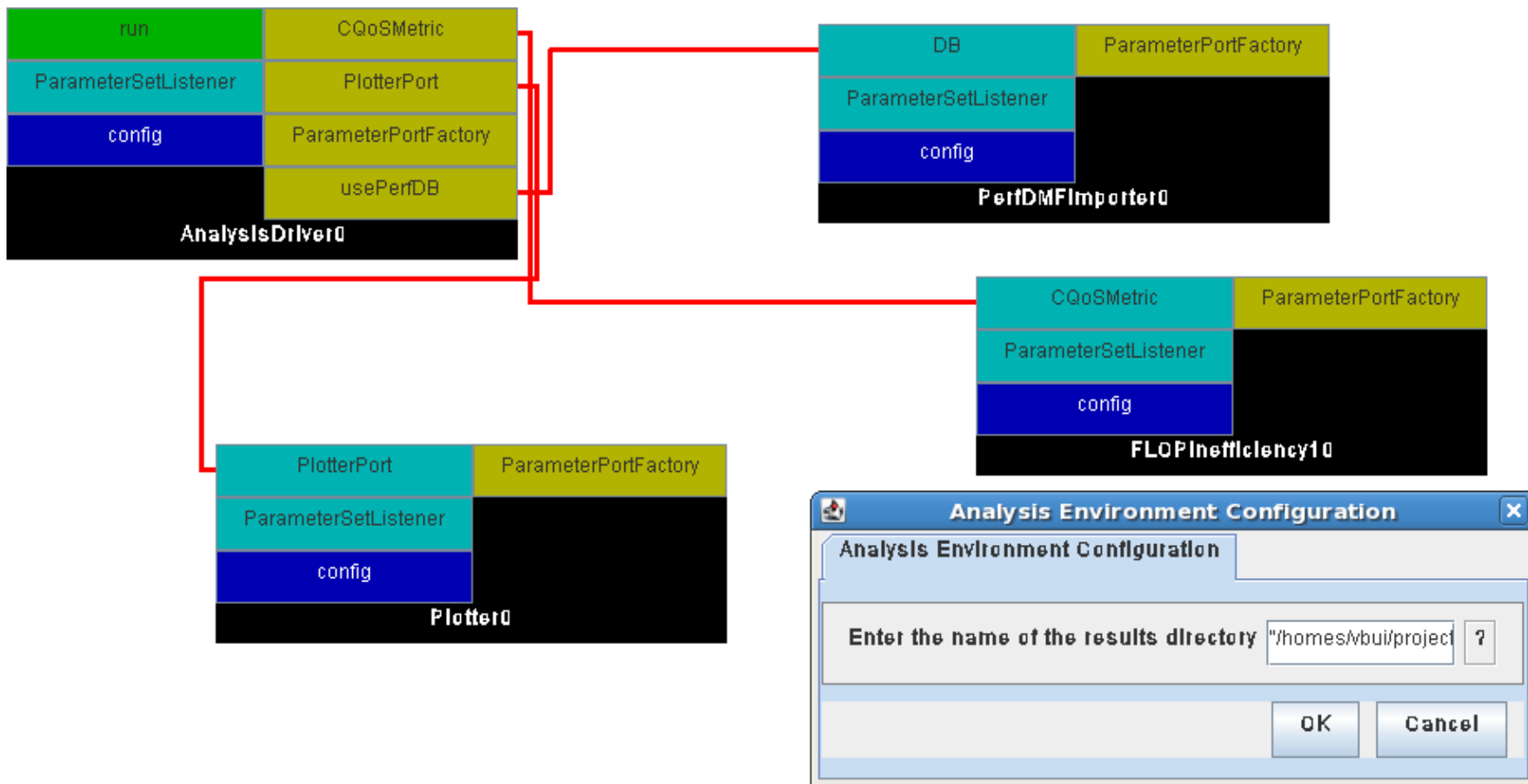
Analysis Phase

Analysis Phase

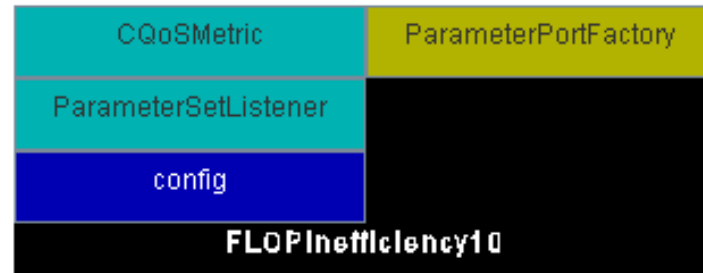
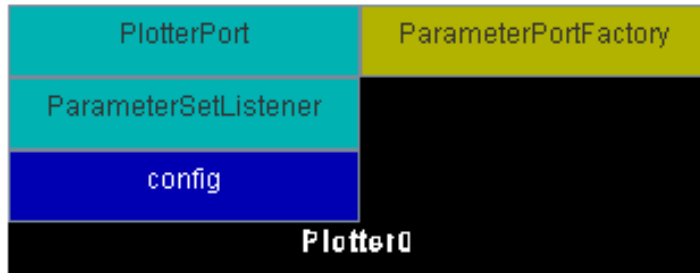
- Specify analysis for a given set of trials
- Determine type of analysis to perform



Analysis Driver Configuration



Analysis Configuration



Analysis Environment Configuration

Analysis Environment Configuration

Enter the analysis script directory ?

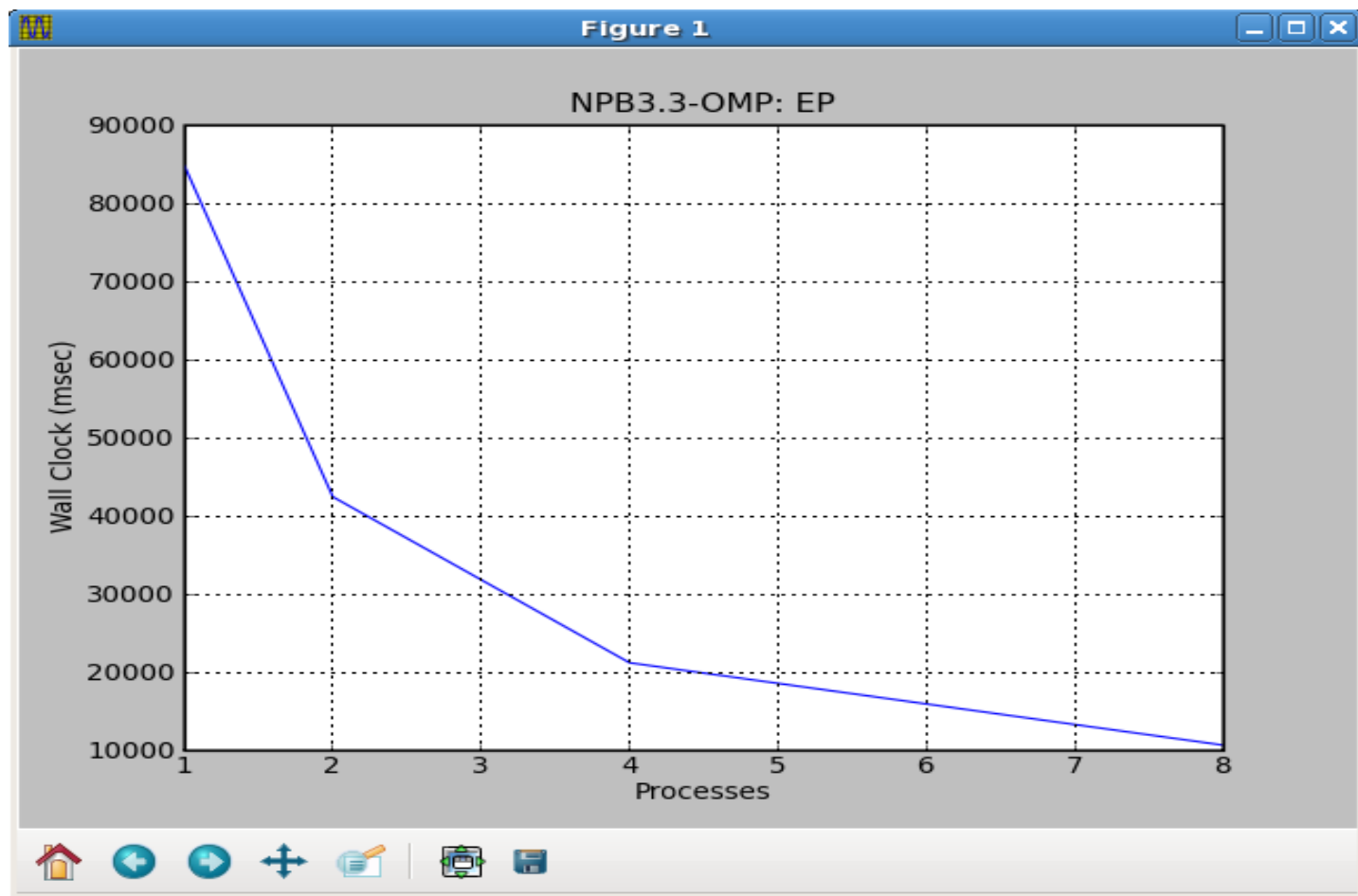
Enter the analysis script name ?

OK Cancel

Sample Code for Plotting Wall Clock

```
for exp in experiments:                                # retrieve experiments
    .....
    for tr in trials:                                  # retrieve trials
        .....
        for event in trial.getEvents():                # retrieve events
            wallSum = 0
            if event == '@PROGRAM_EVENT@':
                for p in range(node_count):
                    wallClock = trial.getInclusive(p, event,
"PAPI_TOT_CYC")/@MHZ@                                # retrieve event value
                    wallSum += wallClock
                    data[node_count] = wallSum / (node_count)
generatePlot(data)                                    # generate plot
```

Plotter: Wall Clock Time



Ccaffeine Script

■ Instantiate component

```
instantiate cqos.perf.AnalysisDriver cqos_perf_AnalysisDriver
```

■ Parameter configuration

```
parameter cqos_perf_AnalysisDriver config resultsdir  
"/homes/vbui/projects/experiments/driven_cavity"
```

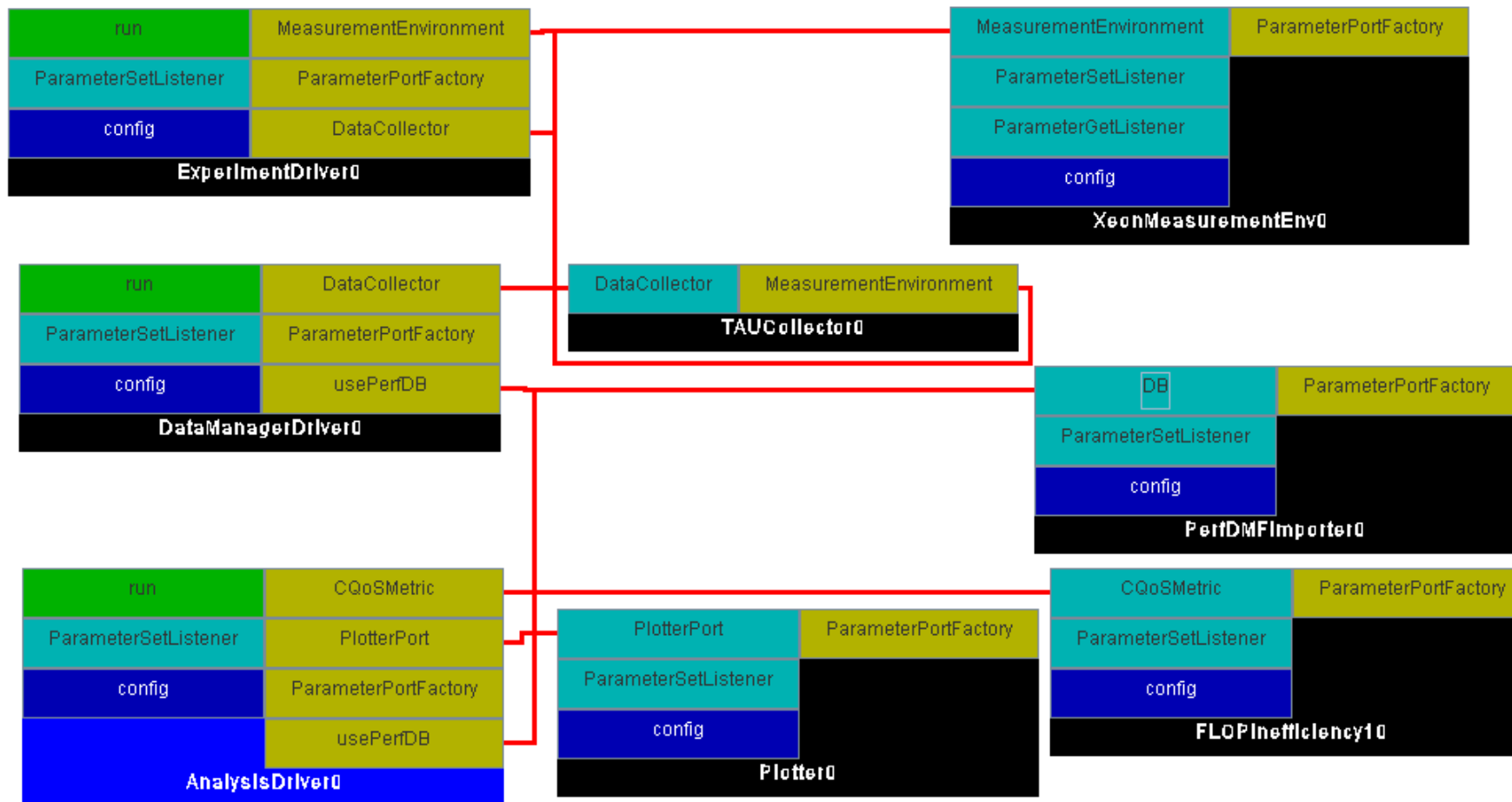
■ Connect ports

```
connect cqos_perf_AnalysisDriver usePerfDB cqos_perf_PerfDMFImporter  
DB
```

■ Invoke driver go

```
go cqos_perf_AnalysisDriver run
```

Assembly of All Components



Summary

- Develop components to automate process of running multiple performance experiments
- Provide a uniform interface integrating support for multiple underlying tools and technology
- Raising the level of efficiency in performance tuning

Future Work

- Extensions to support multiple...
 - Platforms, application spaces, performance tools, database interfaces, and analysis techniques
- Python-based implementation layer of performance components
- Dynamic substitution and reconfiguration of component implementations
- Evaluating the tools with scientific apps and extending based on their needs

Additional Information

■ Support from DOE SciDAC Institutions

- Technology for Advanced Scientific Component Software (TASCS)
- Performance Engineering Research Institute (PERI)

■ Trac Website

- <https://trac.mcs.anl.gov/projects/cca/wiki/performance>