

# Performance Analysis of Grid workflows in K-WfGrid and ASKALON

Hong-Linh Truong and Thomas Fahringer

Distributed and Parallel Systems Group

Institute of Computer Science, University of Innsbruck

{truong,tf}@dps.uibk.ac.at

<http://dps.uibk.ac.at/projects/pma>

With contributions from:

Bartosz Balis, Marian Bubak, Peter Brunner, Schahram Dustdar, Jakub Dziwisz, Andreas Hoheisel, Tilman Linden, Radu Prodan, Francesco Nerieri, Kuba Rozkwitalski, Robert Samborski



AURORA



eGEE  
Enabling Grids for  
E-science in Europe



K-Wf Grid



# Outline

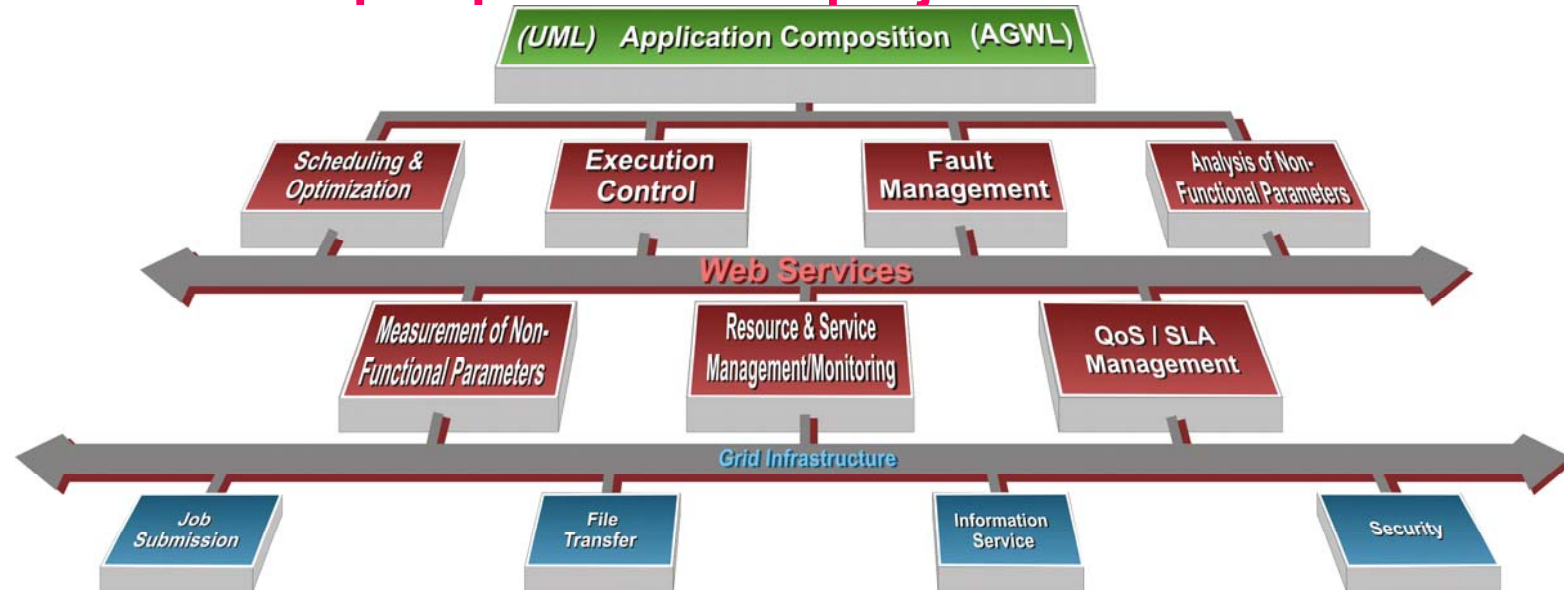
- ❖ Motivation
- ❖ Objectives and approach
- ❖ Performance metrics and ontologies
- ❖ Architecture of Grid monitoring and analysis
- ❖ Conclusions

# Performance Instrumentation, Monitoring, and Analysis for the Grid

- ❖ Challenging task because of dynamic nature of the Grid and its applications
  - Combination of fine-grained and coarse-grained models
  - Multilingual applications
  - Heterogeneous and dynamic systems
- ❖ Not just performance but also dependability
- ❖ The focus of the talk
  - Our concepts, architecture, interfaces and integration

# ASKALON Toolkit

<http://dps.uibk.ac.at/projects/askalon/>



eGEE

K-Wf Grid

A S G

AUSTRIAN  
GRID

AURORA

- ❖ Programming and execution environment for Grid workflows
  - Integrated various services for scheduling, executing, monitoring and analyzing Grid workflows
- ❖ Target applications
  - Workflows of scientific applications (C/Fortran)
    - Material science, flooding, astrophysics, movie rendering, etc.

# The K-WfGrid Project

## ❖ An FP6 EU project

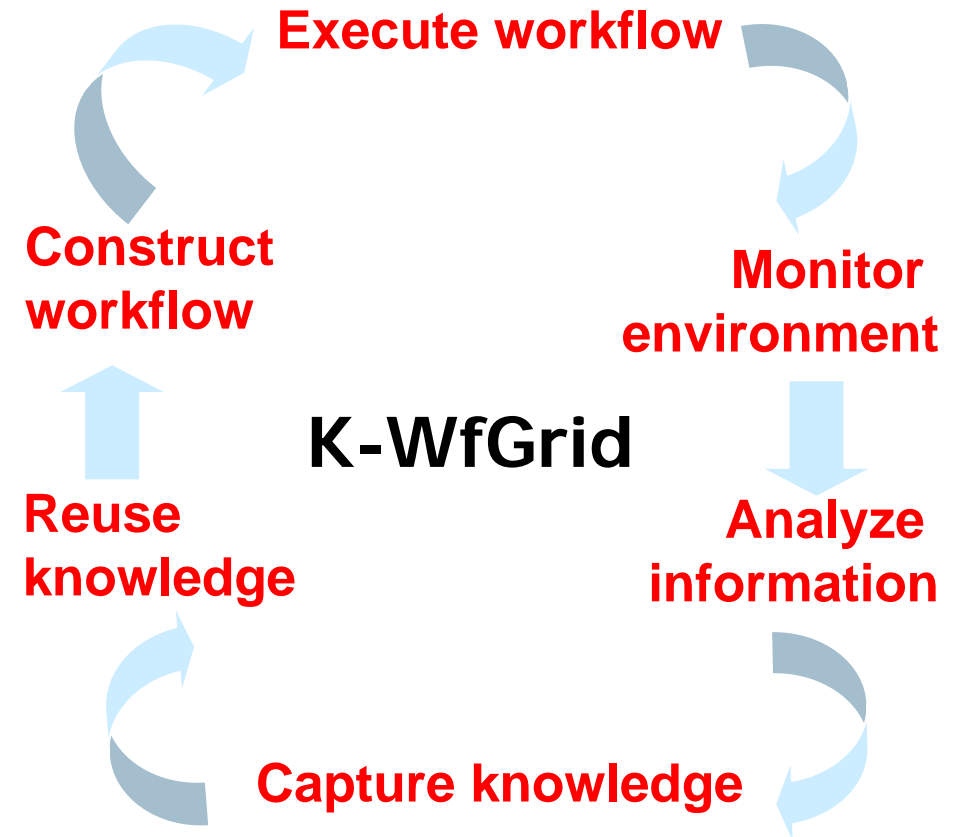
- [www.kwfgrid.net](http://www.kwfgrid.net)

## ❖ Focuses

- Automatic construction and reuse of workflows based on knowledge gathered through execution

## ❖ Target applications

- Workflows of Grid/Web services
- Grid/Web services may invoke legacy applications
- Business (Coordinated Traffic Management, EPR) and scientific (e.g., Food simulation) applications



# The K-WfGrid Project

## ❖ An FP6 EU project

- [www.kwfgrid.net](http://www.kwfgrid.net)

## ❖ Focuses

- Automatic construction and reuse of workflows based on knowledge gathered through execution

## ❖ Target applications

- Workflows of Grid
- Grid/Web services and legacy applications
- Business (Coordination Management, EPR) and scientific (e.g., Food simulation) applications

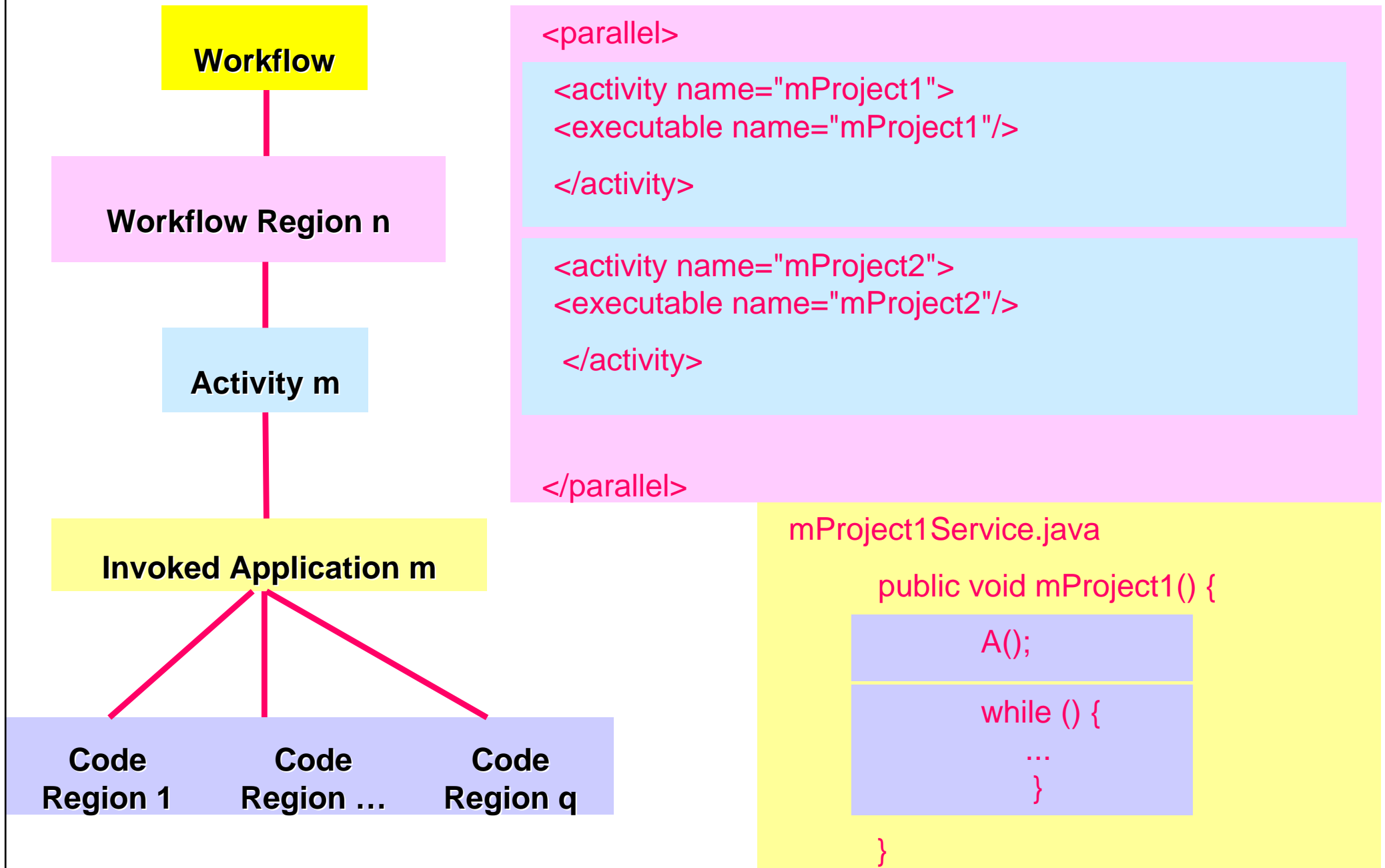
Many Grid users and developers we know emphasize the interoperability, integration and reliability, not just performance



# Objectives, Requirements, Approaches

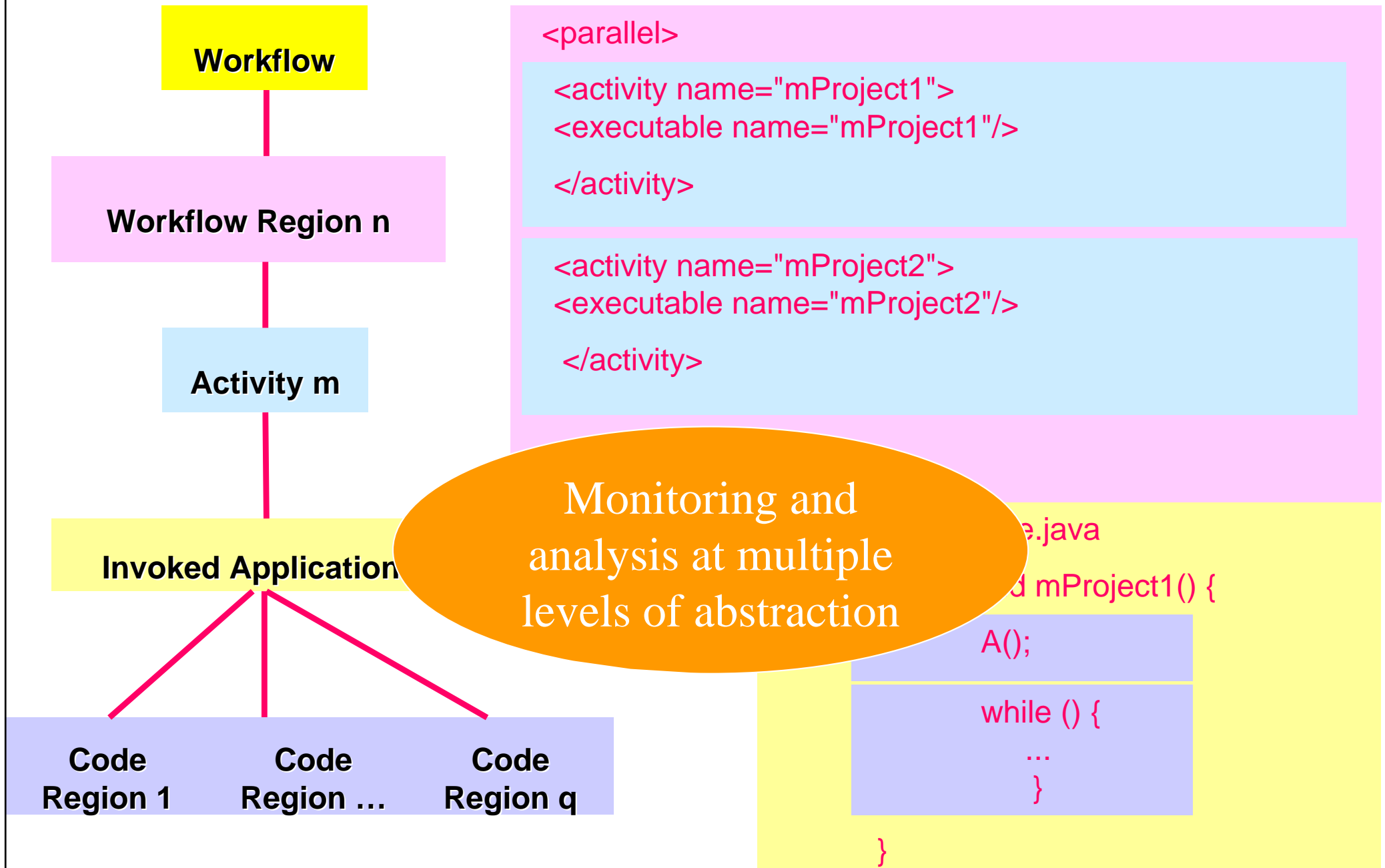
- ❖ Performance monitoring and analysis for Grid workflows of Web/Grid services and multilingual components
- ❖ Performance and dependability metrics
- ❖ Monitoring and performance analysis
  - Service-oriented distributed architecture and peer-to-peer model
  - Unified monitoring and performance analysis system covering infrastructure and applications
  - Standardized data representations for monitoring data and events
  - Adaptive and generic sensors, distributed analysis, performance bottleneck search

# Hierarchical View of Grid Workflows

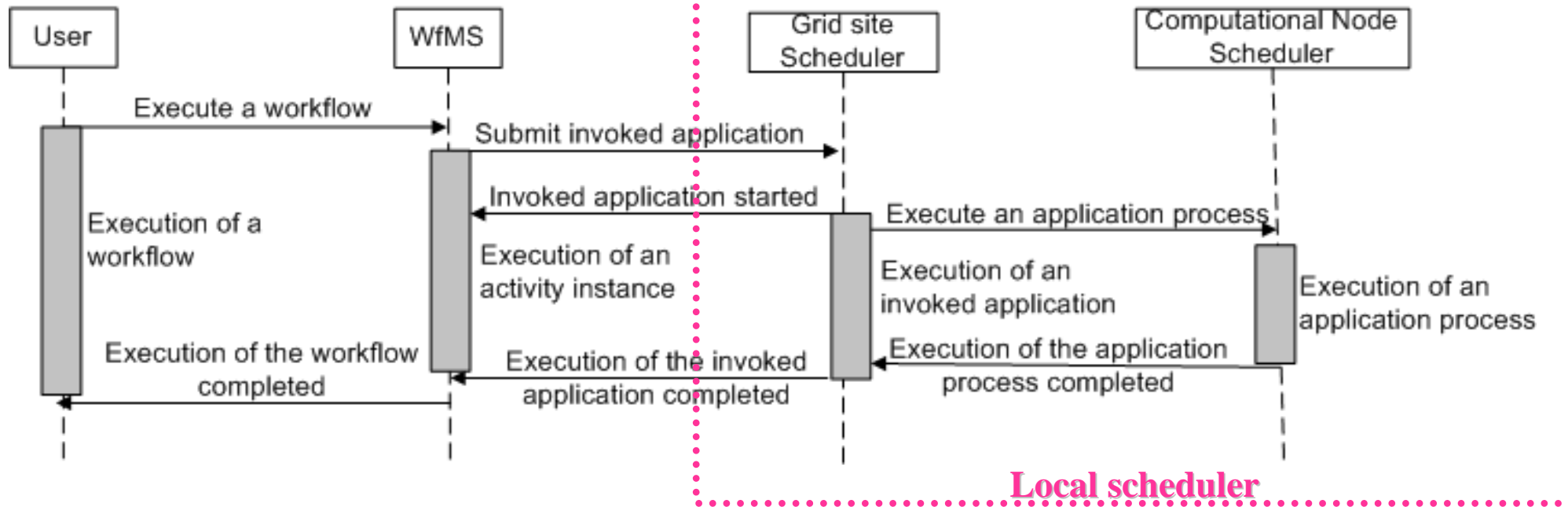




# Hierarchical View of Grid Workflows



# Workflow Execution Model (Simplified)



## ❖ Workflow execution

- Spanning multiple Grid sites
- Highly inter-organizational, inter-related and dynamic

## ❖ Multiple levels of job scheduling

- At workflow execution engine (part of WfMS)
- At Grid sites

# Performance Metrics of Grid Workflows

- ❖ Interesting performance metrics associated with multiple levels of abstraction
  - Metrics can be used in workflow composition, for comparing different invoked applications of a single activity, adaptive scheduling, etc.
- ❖ Five levels of abstraction
  - Code region, Invoked application, Activity ,Workflow Region, Workflow
  - Topdown PMA: from a higher level to a lower one

# Monitoring and Measuring Performance Metrics

- ❖ Performance monitoring and analysis tools
  - Operate at multiple levels
  - Correlate performance metrics from multiple levels
- ❖ Middleware and application instrumentation
  - Instrument execution engine
    - Execution engine can be distributed or centralized
  - Instrument applications
    - Distributed, spanning multiple Grid sites
- ❖ Challenging problems: the complexity of performance tools and data
  - Integrate multiple performance monitoring tools executed on multiple Grid sites
  - Integrate performance data produced by various tools

**We need common concepts for performance data associated with Grid workflows**

# Utilizing Ontologies for Describing Performance Data of Grid workflows

## ❖ Metrics ontology

- Specifies which performance metrics a tool can provide
- Simplifies the access to performance metrics provided by various tools

## ❖ Performance data integration

- Performance data integration based on common concepts.
- High-level search and retrieval of performance data

## ❖ Knowledge base performance data of Grid workflows

- Utilized by high-level tools such as schedulers, workflow composition tools, etc.
- Used to re(discover) workflow patterns, interactions in workflows, to check correct execution, etc.

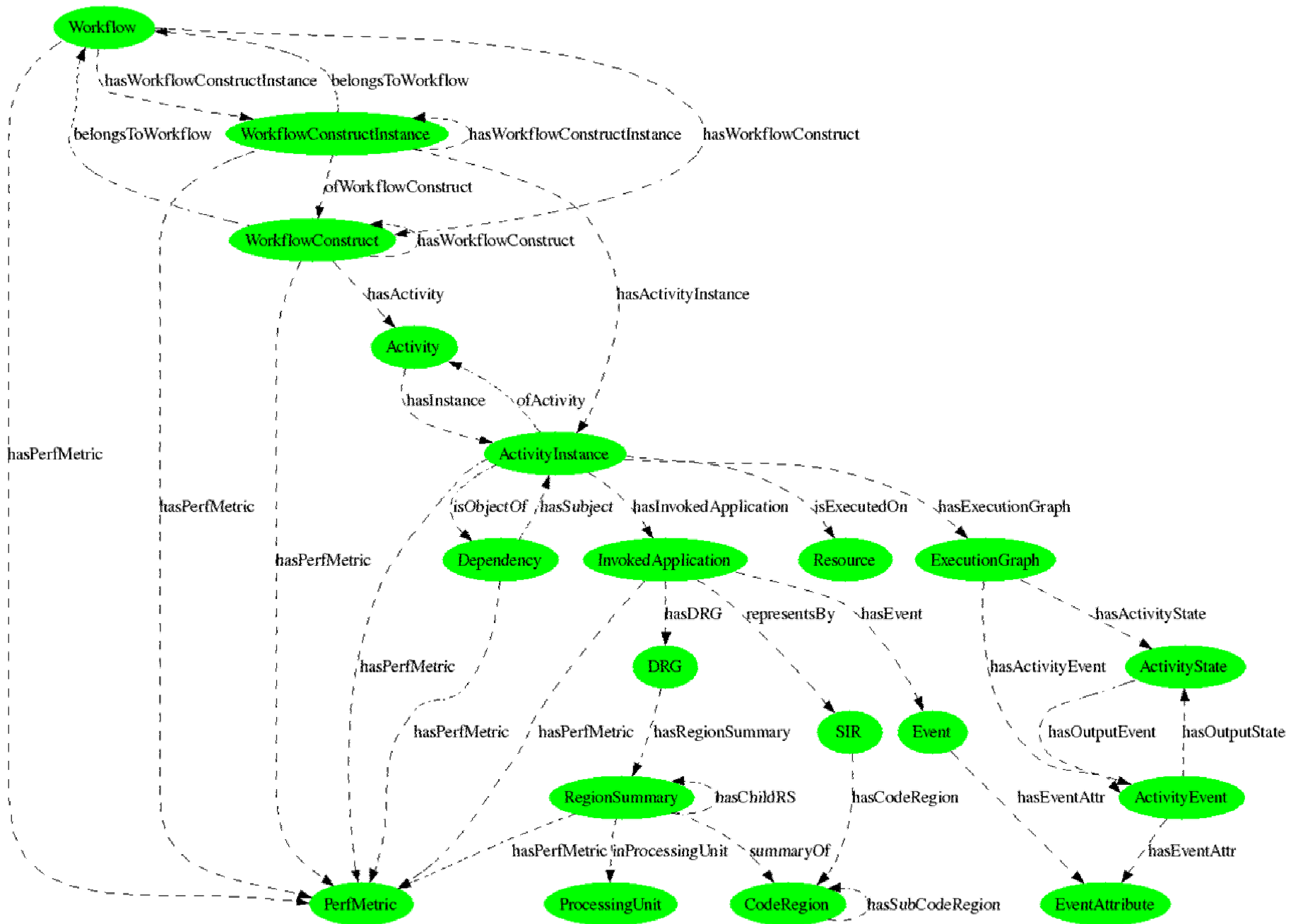
## ❖ Distributed performance analysis

- Performance analysis requests can be built based on ontologies

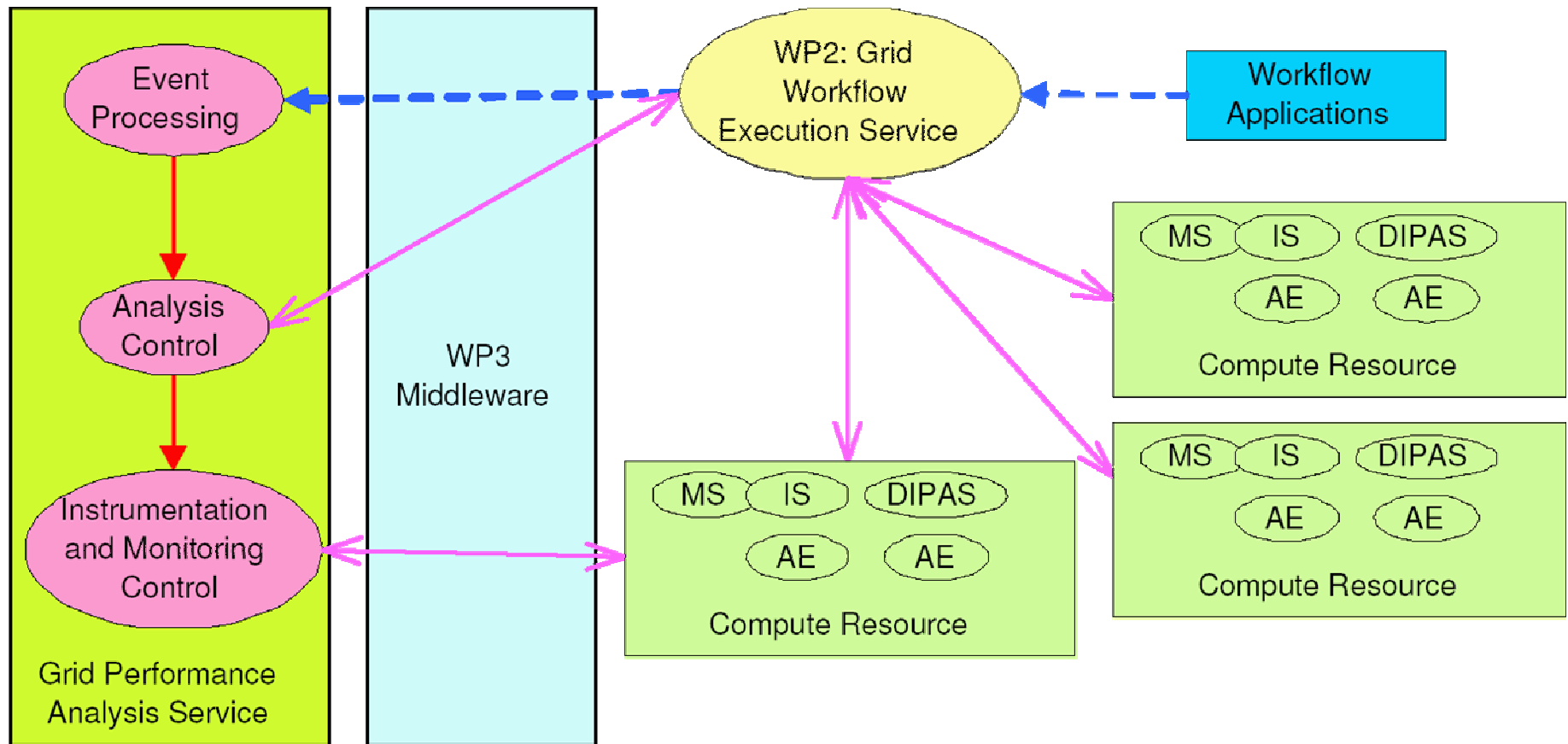
# Workflow Performance Ontology

- ❖ WfPerfOnto (Ontology describing Performance data of Grid Workflows)
  - Specifies performance metrics
  - Basic concepts
    - Concepts reflect the hierarchical structure of a workflow
    - Static and dynamic workflow performance data
  - Relationships
    - Static and dynamic relationships among concepts

# WfPerfOnto



# Monitoring and Analysis Scenario



MS: Monitoring Service, IS: Instrumentation Service, AE: Application Executable,  
 DIPAS: Distributed and Intelligent Performance Analysis Service

← - - - : Data flow   ← - - - : Control flow   ← - - - : Both data and control flows



# Components for Grid PMA

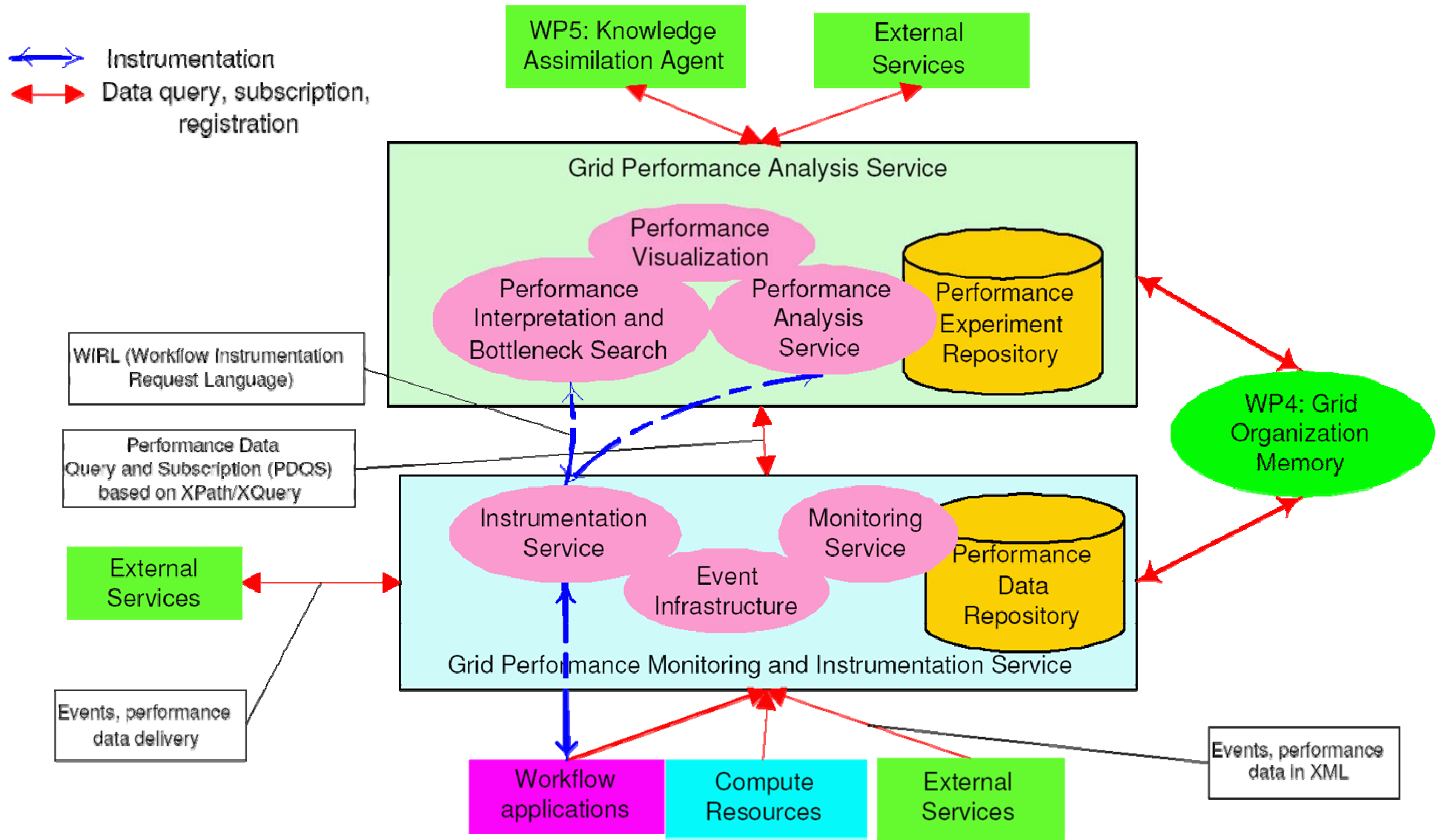
## ❖ Three main parts of a unified performance monitoring, instrumentation and analysis system

- Monitoring and instrumentation services
- Performance analysis services
- Performance service interfaces and data representations
  - We must reuse existing tools and techniques as much as possible

## ❖ Integration model

- Loosely coupled: among Grid sites/organizations
  - Utilizing SOA for performance tools
- Tightly coupled: within services deployed in a single Grid site/organization
  - Interfacing to existing (parallel) performance tools

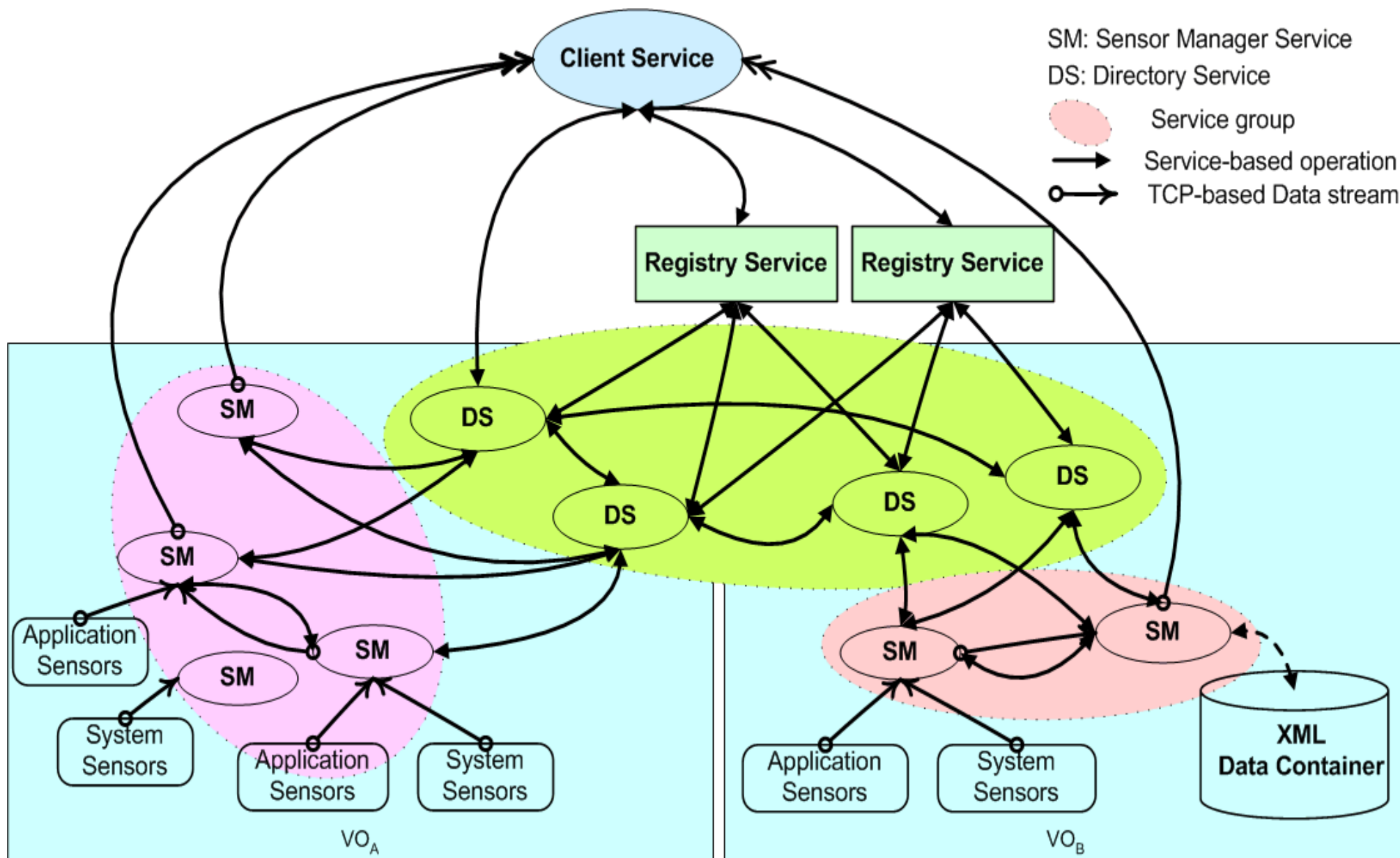
# K-WfGrid Monitoring and Analysis Architecture



# Self-Managing Sensor-Based Middleware

- ❖ Integrating diverse types of sensors into a single system
  - Event-driven and demand-driven sensors for system and applications monitoring, rule-based monitoring
- ❖ Self-managing services
  - Service-based operations and TCP-based stream data delivery
  - Peer-to-peer Grid services for the monitoring middleware
- ❖ Query and subscription of monitoring data
  - Data query and subscription
  - Group-based data query and subscription, and notification

# Self-Managing Sensor Based Monitoring Middleware



# Workflow Instrumentation

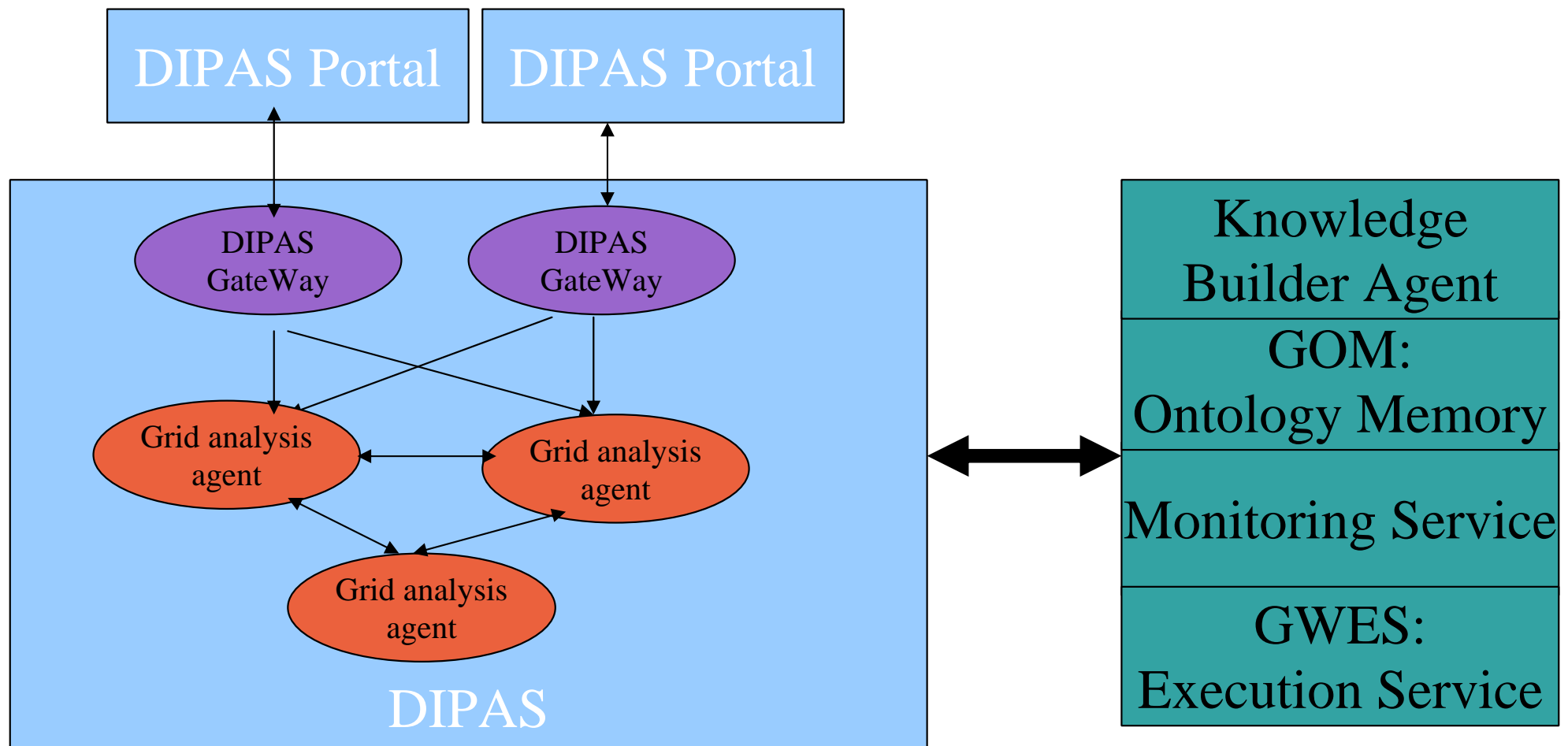
## ❖ Issues to be addressed

- Multiple levels of instrumentation
- Instrumentation of multilingual applications (C/Java/Fortran)
- Must be dynamic (enabled) instrumentation

## ❖ ASKALON and K-WfGrid approach

- Utilizing existing instrumentation techniques
  - OCM-G (Roland Wismueller and Marian Bubak)
    - dynamic enabled instrumentation C/Fortran
  - Dyninst (Barton Miller, Jeff Hollingsworth)
    - for binary code generated from C/Fortran
  - Source/dynamic instrumentation of Java (from Java 1.5)
- Using APART standardized intermediate representation (SIR)
- XML-based request for controlling instrumentation

# DIPAS: Distributed Performance Analysis



- ❖ Grid analysis agent accepts WARL and returns performance metrics described in XML under a tree of metrics
  - WARL (workflow analysis request language): based on concepts and properties in WfPerfOnto

# Workflow Overhead Classification

## ❖ Middleware

- Scheduler
- Resource Manager
- Execution management
  - Control of parallelism
  - Loss of parallelism

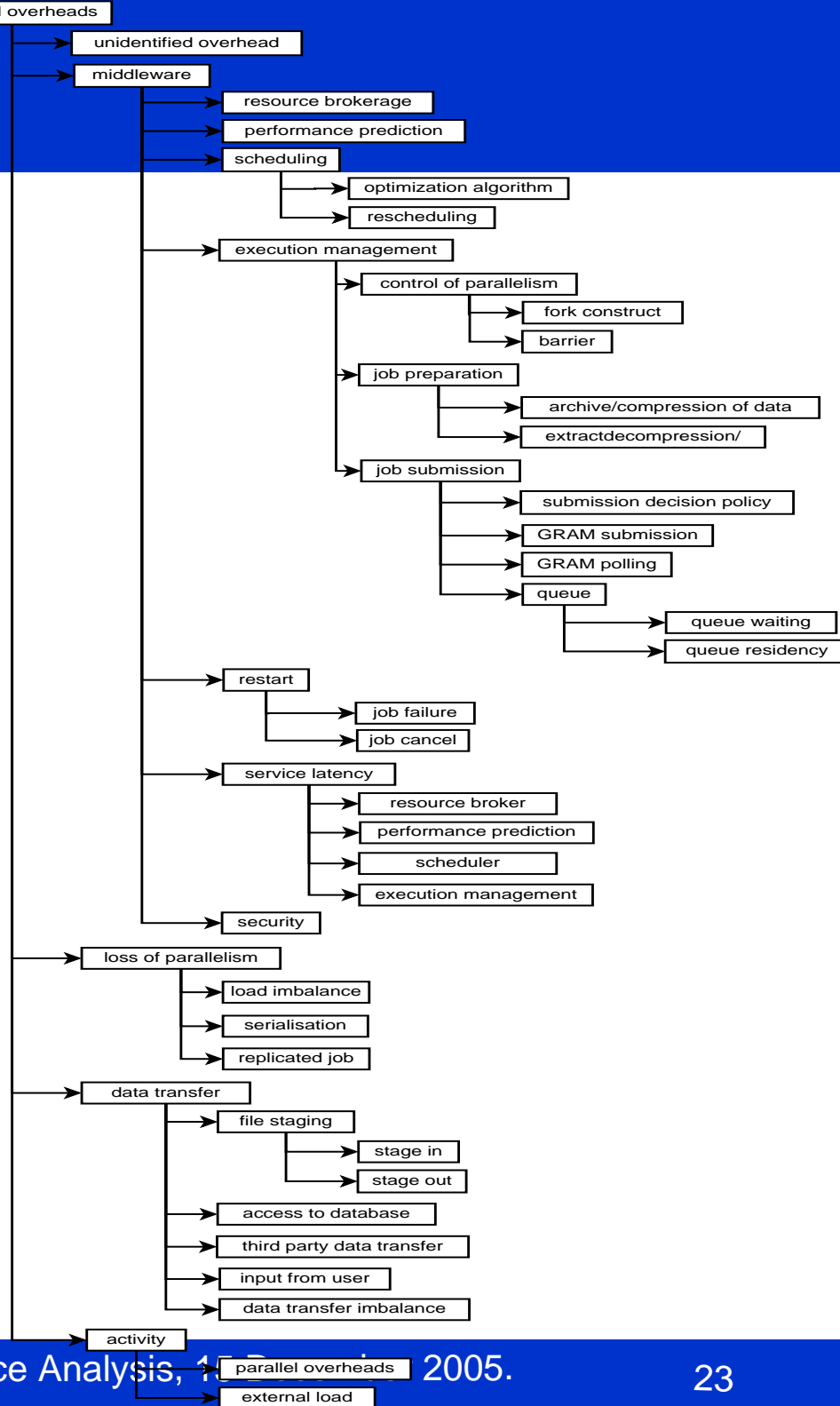
## ❖ Loss of parallelism

- Load imbalance
- Serialization
- Replicated job

## ❖ Data transfer

## ❖ Activity

- Parallel processing overheads
- External load



# Current status of the implementation

## ❖ SOA-based

- Monitoring, Instrumentation and Analysis services are GT4 based
- XML-based for performance data representations and requests

## ❖ Monitoring and Instrumentation Services

- gSOAP, GSI-based dynamic instrumentation service
- Java-based dynamic instrumentation
- OCM-G
- But we need to integrate them into a single framework for workflows and it is a non trivial task

## ❖ Analysis services

- GT4-based with distributed components
- Simple language for workflow analysis request (WARL), designed based on WfPerfOnto
- Metrics are described in XML



# Example: Dynamic Instrumentation

**Active Activities**

- Active Activities
- tRawImage2
- tRawImage1
- mProject11
- mProject12

**User Processes**

Username	Status	PID	PPID	ProcName
truong	S	16304	1	globus-job-manager
truong	S	16309	1	/home/truong/projects...
truong	O	16339	16338	/usr/bin/ps
truong	S	16337	1	./sismsfactory
truong	R	16312	16309	mProject
truong	S	14752	14748	-csh
truong	S	16143	15395	./sismsfactory
truong	S	15390	14752	dtterm

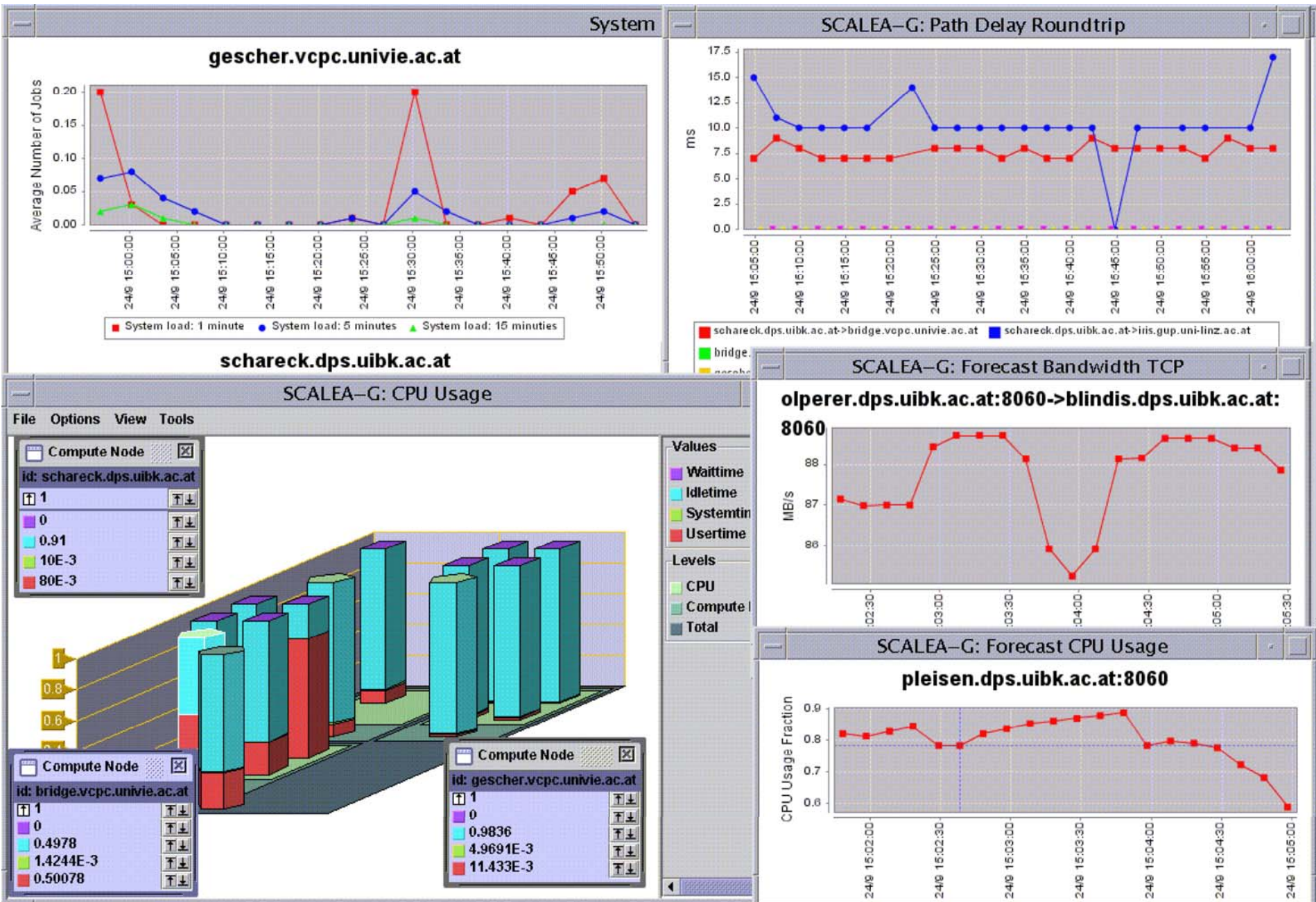
**IRL Editor**

```
<?xml version="1.0" ?>
<irl>
<experiment>
  <applicationName>mProject</applicationName>
  <jobID>16312</jobID>
  <experimentID>Montage</experimentID>
  <activityID>mProject11 </activityID>
</experiment>
<request name="instrument">
  <task>
    <coderegion unit="main" name="readTemplate" id="f4...
    <metrics>WTIME</metrics>
  </task>
  <task>
    <coderegion unit="main" name="readFits" id="f60_rea...
```

**Application SIR**

- mProject
  - SIR
    - main(function,u\_main)
    - debugCheck(function,u\_debugCheck)
    - checkHdr(function,u\_checkHdr)
    - readTemplate(function,u\_readTemplate)
    - readFits(function,u\_readFits)
    - pix2wcs(function,u\_pix2wcs)
    - convertCoordinates(function,u\_convertCoordinates)
    - wcs2pix(function,u\_wcs2pix)
    - UpdateBounds(function,u\_UpdateBounds)
    - printFitsError(function,u\_printFitsError)
    - computeOverlap(function,u\_computeOverlap)
    - printError(function,u\_printError)
    - fitsCheck(function,u\_fitsCheck)
    - FITSerror(function,u\_FITSerror)

# Snapshot: Online System Monitoring



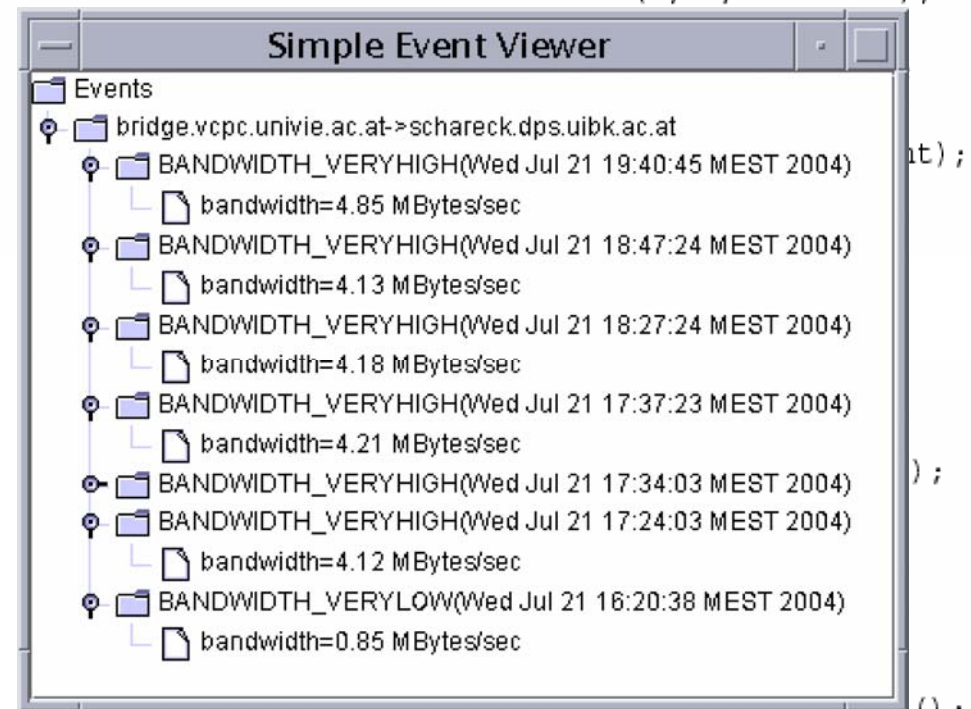
# Rule-based Monitoring

- ❖ Sensors use rules to analyze monitoring data
- ❖ Rules are based on IBM ABLE toolkit

- ❖ **Example:**

- A network path in the Austrian Grid, bandwidth < 5MB/s (based on Iperf)
- Define a fuzzy variable with VERY LOW, LOW, MEDUM, HIGH, VERY HIGH
- Fuzzy rules: Events send when bandwidth VERY LOW or VERY HIGH

```
Fuzzy bandwidth= new Fuzzy( 0,5) {  
    Shoulder VERYLOW = new Shoulder(0, 1, ARL.Left);
```



```
    }  
    R_OTHER: doNormalReaction();
```



# Online Infrastructure Monitoring

gridsphere portal framework  
open-source / portlet jsr168 compliant

Abmelden  
Willkommen, kwfgrid

Willkommen WP3:PMA

Configuration PDQS Display data Workflow Analysis Error

Data Query and Subscription Portlet

DataTypeID=>ResourceID: path.predict.bandwidth.capacity.TCP=>hydra.gup.uni-linz.ac.at:40010->altix1.jku.austriangrid.at:40010

DataFilter: path.predict.bandwidth.capacity.TCP=>hydra.gup.uni-linz.ac.at:40010->altix1.jku.austriangrid.at:40010  
path.predict.bandwidth.capacity.TCP=>hydra.gup.uni-linz.ac.at:40010->schalberg.coma.sbg.ac.at:40010  
path.predict.bandwidth.capacity.TCP=>schalberg.coma.sbg.ac.at:40010->altix1.jku.austriangrid.at:40010  
path.predict.bandwidth.capacity.TCP=>schalberg.coma.sbg.ac.at:40010->hydra.gup.uni-linz.ac.at:40010  
path.predict.bandwidth.capacity.TCP=>schalberg.coma.sbg.ac.at:40010->altix1.uibk.ac.at:40010

Time from: path.predict.bandwidth.capacity.TCP=>hydra.gup.uni-linz.ac.at:40010->altix1.uibk.ac.at:40010

Generate PDQS Query

path.delay.roundtrip=>altix1.uibk.ac.at->altix1.jku.austriangrid.at  
path.delay.roundtrip=>altix1.uibk.ac.at->rosnicka.ui.savba.sk  
host.system.loadavg=>altix1.uibk.ac.at  
host.mem.used=>altix1.uibk.ac.at  
user.ps=>altix1.uibk.ac.at  
host.cpu.used=>altix1.uibk.ac.at

Query Data Subscribe Data

**A little guide**

Now you can choose the DataType and the ResourceID of the Request. You can add a DataFilter or set the subscription time if you want to subscribe. There are datatypes that can only be queried and not subscribed. Then click on "Generate PDQS Query". After this you can select if you want to query or subscribe by selecting the corresponding button. Now you can go to the next tab "Display Data". If you think that an error has occurred, click on "Error"

gridsphere portal framework  
open-source / portlet jsr168 compliant

Abmelden  
Willkommen, kwfgrid

Willkommen WP3:PMA

Configuration PDQS Display data Workflow Analysis Error

Display Data

**Data display**

ResultID: 76b5c80-4d3c-11da-937f-addf82c480a

DataTypeID: host.system.loadavg

ResourceID: altix1.uibk.ac.at

Time: 11/4/05 3:08 PM

SessionID: DB7A11039D59826998CE1CA51372B81A

View Data

**altix1.uibk.ac.at**

Average Number of Jobs

System load: 1 minute System load: 5 minutes System load: 15 minutes

**A little guide**

Here you can see the data you subscribed or you queried. If there is no picture, it could be that the data hasn't been fully transmitted. To update or change to another monitoring please push on "View Data". If you think an error has occurred please go to the "Error"

## Display static Data

Hostname	zeus70.cyf-kr.edu.pl
Resource id	zeus70.cyf-kr.edu.pl
Addresses:	149.156.9.110
System model	null
Physical memory	2020.7 MB
Virtual memory	2651.3 MB
Number of CPUs	2
CPU type	x86
CPU speed	2800 MHz
OS name	Linux
OS version	2.4.27-2-686-smp
Harddisk Size	75 GB

# DAG-based workflow monitoring and analysis

Online application profiling

Monitoring data of workflow activity execution

The screenshot shows the SCALEA-G: Application Profile Analysis interface. At the top, there's a menu bar with 'File', 'Setting', 'View', and 'Analysis'. The main area is divided into several panels:

- DAG View:** A Directed Acyclic Graph showing workflow activities. Nodes include tRawImage1, tRawImage2, mimgtbl11, mimgtbl12, mProject11, mProject12, tProjectedImage1, tProjectedImage2, and mimgtbl2. Arrows indicate dependencies between these tasks.
- Online Application Data Tree:** A hierarchical tree view showing the application's execution context. It includes 'Experiment Montage', 'mProject2', 'Process 7924', 'Thread 0', and various regions like 'Region 41:pix2wcs[0:0:0]', 'Region 44:computeOverlap[0:0:0]', etc.
- Metric Table:** A table with 'Metric Name' and 'Metric Value' columns. It lists 'nsubs' (0), 'ncalls' (3,368,877), and 'lwttime' (53.914.889).
- Historical Data Reg:** A line graph titled 'SCALEA-G: Historical Data Reg' showing 'Value (s)' on the y-axis (0 to 125) and 'ncalls' on the x-axis (0 to 750,000). The data points show a steady upward trend.
- Forecast CPU Usag:** A line graph titled 'SCALEA-G: Forecast CPU Usag' for 'hafner.dps.uibk.ac.at: 8060'. The y-axis is 'CPU Usage Fraction' (0.50 to 1.00) and the x-axis shows time intervals from 14:10 14:57:00 to 14:10 15:00:00. The usage is mostly around 0.50, with a spike to 1.00 at the end.
- Active Activities:** A list of activities with a legend for 'submitted' (red), 'active' (blue), and 'completed' (green). The list includes tUncorrectedMosaic, mAdd, mimgtbl2, tProjectedImage2, tProjectedImage1, mProject12, mimgtbl12, mProject11, mimgtbl11, tProjectedImage, tProjectedImage, mimgtbl2, mAdd, and tUncorrectedMc.

Monitoring data of computational node



# K-WfGrid: Online Workflow Monitoring and Analysis

The screenshot displays the GridSphere Portal interface in Mozilla Firefox. The main content area shows a workflow diagram on the left and a monitoring dashboard on the right. The workflow diagram includes nodes for 'generateAreaFile' and 'calculateTrafficFlow', with data flows labeled 'netFileURL', 'toZone', 'from', 'generateAreaFileReturn', 'areaFileURL', and 'calculateTrafficFlowReturn'. The monitoring dashboard features a progress bar for 'Workflow: Execution Time (s)' ranging from 0.0 to 4.0, with a legend for 'Initializing', 'Queueing', 'Processing', and 'Finalizing'. Below the progress bar is a scatter plot of 'Activity' over time, and a list of 'Active Activities' including 'generateAreaFile(truong-03', 'calculateTrafficFlow(truong-03', and 'calculateEmission(truong-03'. A console window at the bottom shows the applet's startup logs and the URL 'http://pc6163-c703.uibk.ac.at:40370/wsrf/services/kwfguid/DIPASFactory'.

GridSphere Portal - Mozilla Firefox  
File Edit View Go Bookmarks Tools Help  
http://goedis.dps.uibk.ac.at:40250/gridsphere/gridsphere?cid=applet&JavaScript=enabled  
Kostenlose Hotmail Links anpassen Windows Media Windows

Test Workflow Workflow Experiments Analysis  
truong-03331c50-4537-11da-953e-e3f268ddc24d  
Workflow ID : truong-03331c50-4537-11da-953e-e3f268ddc24d  
Current Status: completed @ Tue Oct 25 11:09:18 CEST 2005

Workflow: Execution Time (s)  
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0  
■ Initializing ■ Queueing ■ Processing ■ Finalizing

Active Activities  
generateAreaFile(truong-03  
calculateTrafficFlow(truong-03  
calculateEmission(truong-03

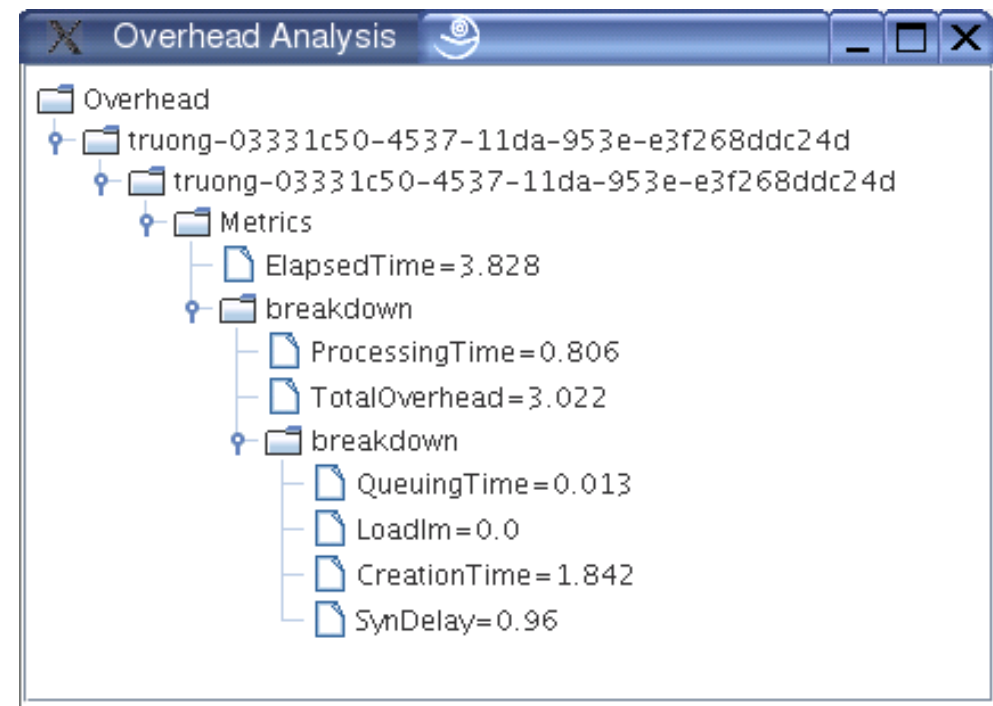
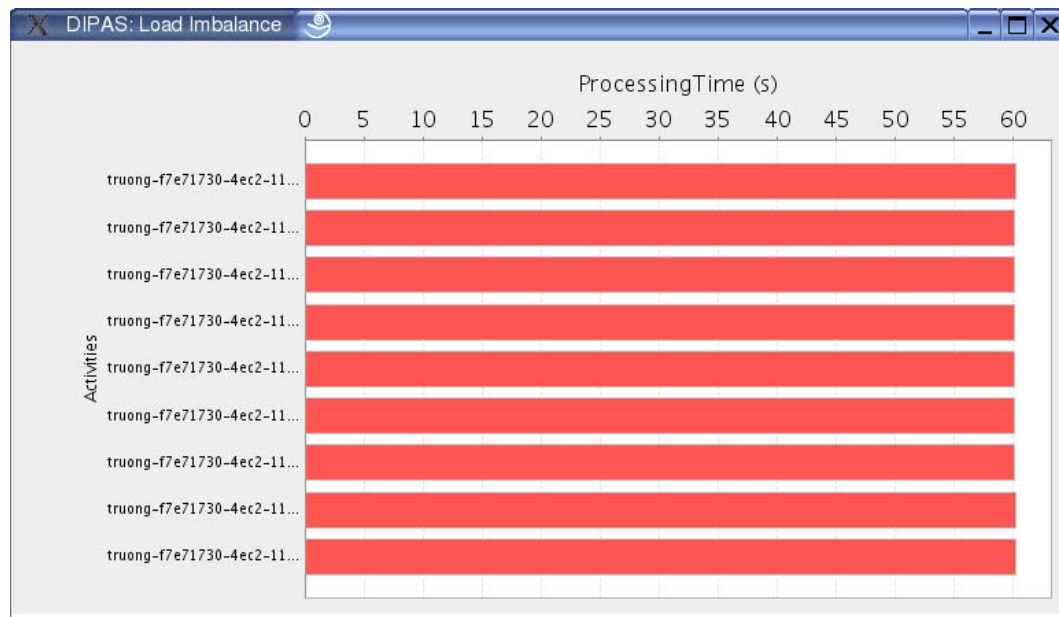
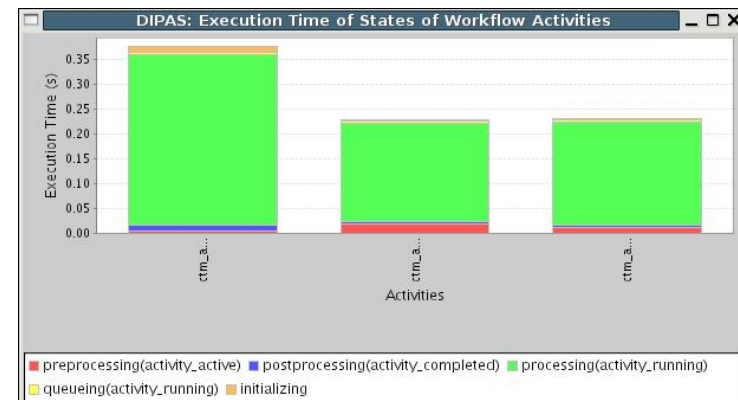
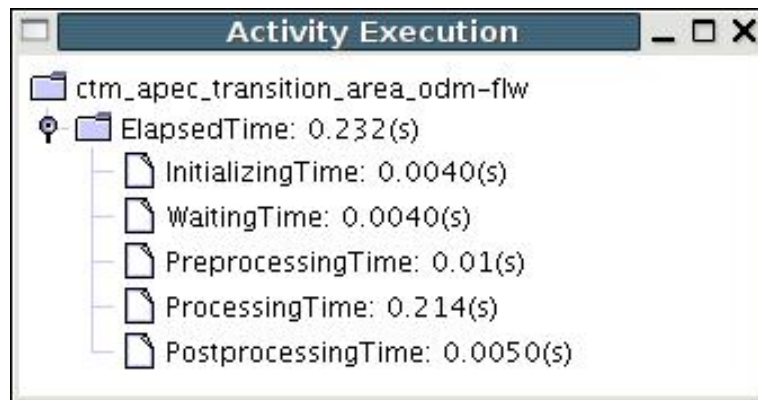
Analyze WS Operation

{APPLET READY} #make sure container is deployed #check and change host if necessary  
setServerURI()  
ID list empties, URI set-> get new WorkflowIDs  
getWorkflowID()  
http://pc6163-c703.uibk.ac.at:40370/wsrf/services/kwfguid/DIPASFactory

Applet net.kwfguid.dipas.client.ActivityApplet started

Start Inbox for tru... 3 Firefox 2 SSH Secu... K-WfGrid\_wp... kwfguid IrfanView DE 23:52

# Online Workflow Analysis



# Conclusions

- ❖ The architecture of monitoring and analysis service must tackle the dynamics and the diversity of the Grid
  - Service-oriented, peer-to-peer model, adaptive sensors
- ❖ Integration and reuse are important issues
  - Loosely coupled and tightly coupled
  - Do not neglect data representations and service interfaces
- ❖ Performance metrics and ontology for Grid workflows
  - What performance metrics are important and how to measure them
  - Common and generic concepts and relationships monitored and analyzed
- ❖ Given well-defined service interfaces, data representation, performance metrics and ontology
  - Simplify the integration among components
  - Towards automatic, intelligent and distributed performance analysis

UIBK perf. work: <http://dps.uibk.ac.at/projects/pma/>  
Papers: <http://dps.uibk.ac.at/index.pl/publications>