

# **Usage Policy (UPL) Research for GriPhyN & iVDGL**

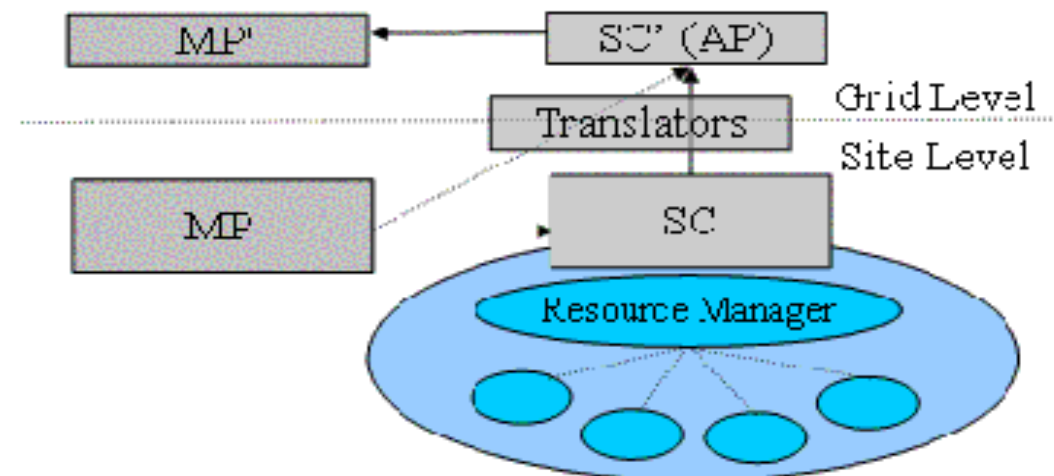
Catalin L. Dumitrescu, Michael Wilde, Ian Foster  
The University of Chicago

# Outline

- Grid03 Deployment Model
- What is UPL
- Motivating Scenario
- Problems
- Time Frame
- Evaluation Methodology
- Open Questions

# Grid03 Deployment Model

- MP: Manager Policy – the description provided by the person in charge about how resources must be used [Site Level]
- SC: Administrator Policy – the technical description written by site's administrator, in short, the RM's configuration files [Site Level]
- SC' (AP): Abstract Policies – Grid level *understanding* of site policies
- MP': AP translation for verification and conformance purposes
- Translators: reverse SC to MP, but to a Grid understanding, using percentages [slide 5], and trying to abstract to a common RM model
- AP + MP' + translators  
+ others = UPL Service



# What is UPL?

- UPL: resource owners' (local policy makers) statements about how their resources must be allocated (high level descriptions)
  - *high level GOAL or MP*
- RM Priorities/Rules: resource administrators mappings of resource owners' statements to different software RMs' syntaxes – *the local POLICY that is actually implemented or SC*
- Abstract Policies: grid level understanding of the UPL, extracted from the SC – reverse translation from SC to MP done by means of automated tools

# Example

- The resource owners' statements (**MP**) for site X is:
  - We have a cluster with 380 CPUs
  - At any time: ATLAS has a 30% ; the other VOs together have just 10%
  - When additional resources are available, Grid03's VOs can grab these resources
- The Condor priorities (**SC**) used to realize the above description is:
  - `% condor_userprio -setfactor atlas 2`
  - `% condor_userprio -setfactor others 9`
- The Grid understanding (**AP**) example:
  - RM type: Condor
  - RM allocations: ATLAS:30% Others:10%
  - UPL type: VOESF

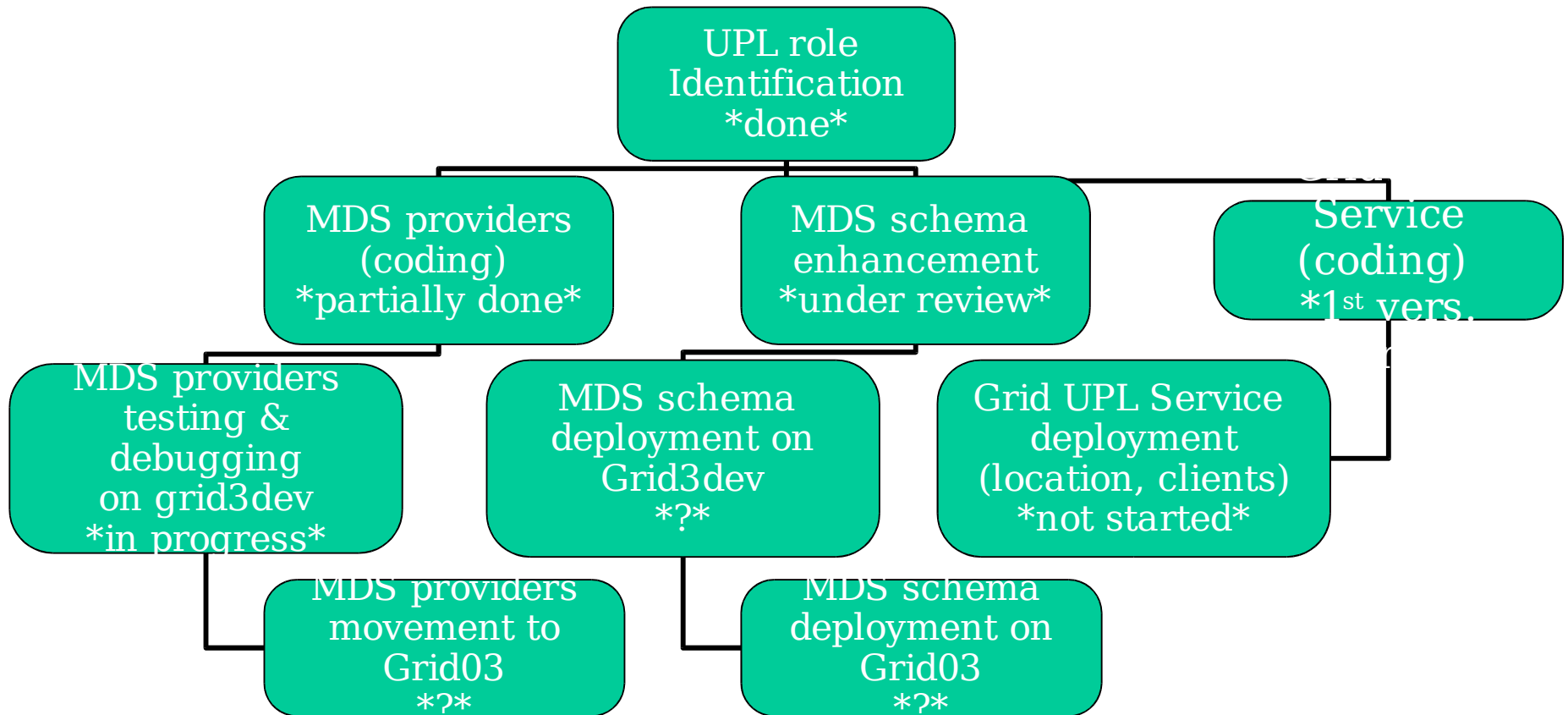
# Problems

- *Is the UPL GOAL really necessary? Is it useful for the Grid environment? If yes, why?*
- Roles identification
- Amount of information to be made available from individual sites
- Heterogeneity considerations:
  - × Different RM models (Condor, PBS, LSF, others)
  - × RM priorities:
    - local vs. remote users
    - Atlas vs. CMS vs. CMStest

# Deployment Technicalities

- Site Level:
  - × MDS providers: collect SC, translate to AP and publish it into MDS
  - × MDS schema enhancement: UPL-specific objects and attributes for storing RM type and per-VO allocations
- Grid Level / UPL **OGSA service**:
  - × *SC collection and translation to AP support*
  - × Smart UPL answers: “*From the list  $L$  of sites, which is the subset  $S$  of sites where VO  $V$ 's workload is possibly to run?*”, then “*Which is the best site  $X$  to send VO  $V$ 's workload?*”
  - × Criteria for \*best\* site: #CPUS free, lowest cost, most required files available, most free space, etc

# Time Frame



# Gains

- Additional information that give grid schedulers hints about where to submit jobs – for example, when a site is busy with work from a VO which had grabbed all resources when they were free
- Time-based entitlement to resources – VOs are guaranteed under different FS policies that they can use resources when they need them instead of maintaining constant workloads

# Evaluation Methodology

- Metrics:
  - × UPL accuracy: achieved vs. allocated
  - × Response Time: time interval from submission to start
  - × RoundTrip Time: time interval from submission to end
- Real Workloads (simultaneously running on Grid03 resources):  
Bio, Atlas, CMS, bTeV.

# Open Questions

- Is UPL based resource allocation really necessary?
- Is the proposed model good enough to achieve initial goals?
- How do I know that I have succeeded?