CMS Production Workflow Management System

Peter Elmer
Princeton University
WLCG Asian Tier-2 Workshop
03 Dec, 2006

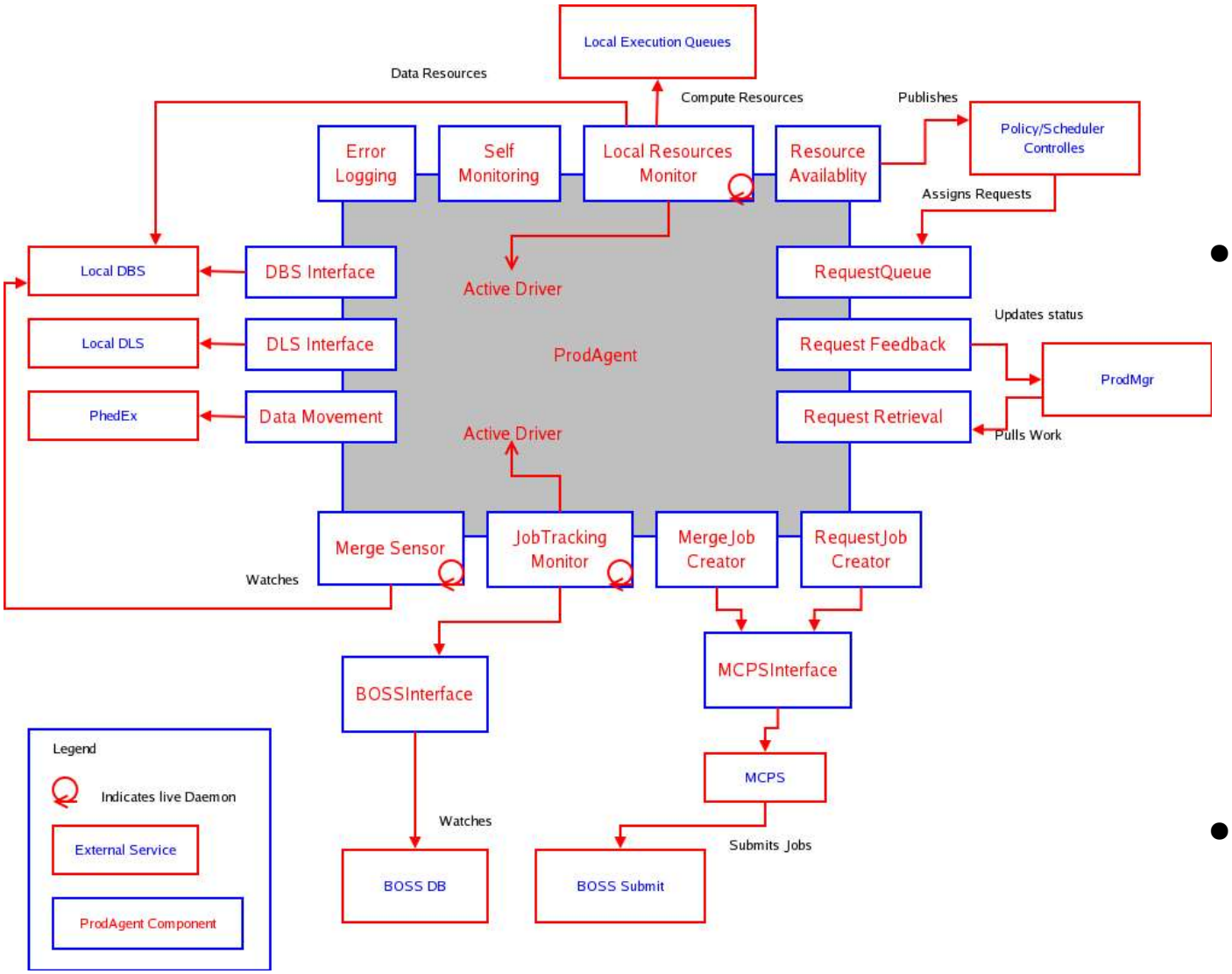
Production Workflow System

- CMS uses a single Production Workflow System
- Currently we use for MC production, skimming, etc., but it is designed to be used also with real data as a general workflow management system
- We also plan that this system can be used as the backend for bulk analysis work, in combination with CRAB
- In this talk I'll describe what it is and what sites (e.g. Tier-2's) should expect from it, with a bit less emphasis as to how to install and operate it

ProdAgent

- ProdAgent:
  - Manages job submission (LCG + OSG), job tracking and file and dataset bookkeeping
  - Tracks the files output by the jobs, performs merges to get reasonable file sizes
  - Publishes (merged) results into global dataset bookkeeping
  - Uses the same components (DM, job tracking, etc.) as we are using for analysis, etc.
  - Data Handling system “unto itself”
  - Maintains priority queue for job submissions (planned)
  - Requests work allocations from ProdMgr (planned)

ProdAgent Components



- This is the software architecture of ProdAgent
- In practice it boils down to a set of python daemons which can be started with single command, communicating via a database used internally
- The SW installation is via rpm, as I described earlier

Status of ProdAgent

- We began using ProdAgent in spring, 2006, initially providing release validation samples for CMSSW releases
- Starting from July, 2006, we used it for the CSA06 MC pre-production
  - 4 teams, each with a ProdAgent instance (3 LCG + 1 OSG)
  - Each team assigned a set of sites to which to submit jobs
- During CSA06 it was used for skimming and re-reconstruction
- We are now continuing with post-CSA06 MC production using CMSSW_1_1_x (and hopefully soon CMSSW_1_2_x)

We are looking for a new ProdAgent operation team

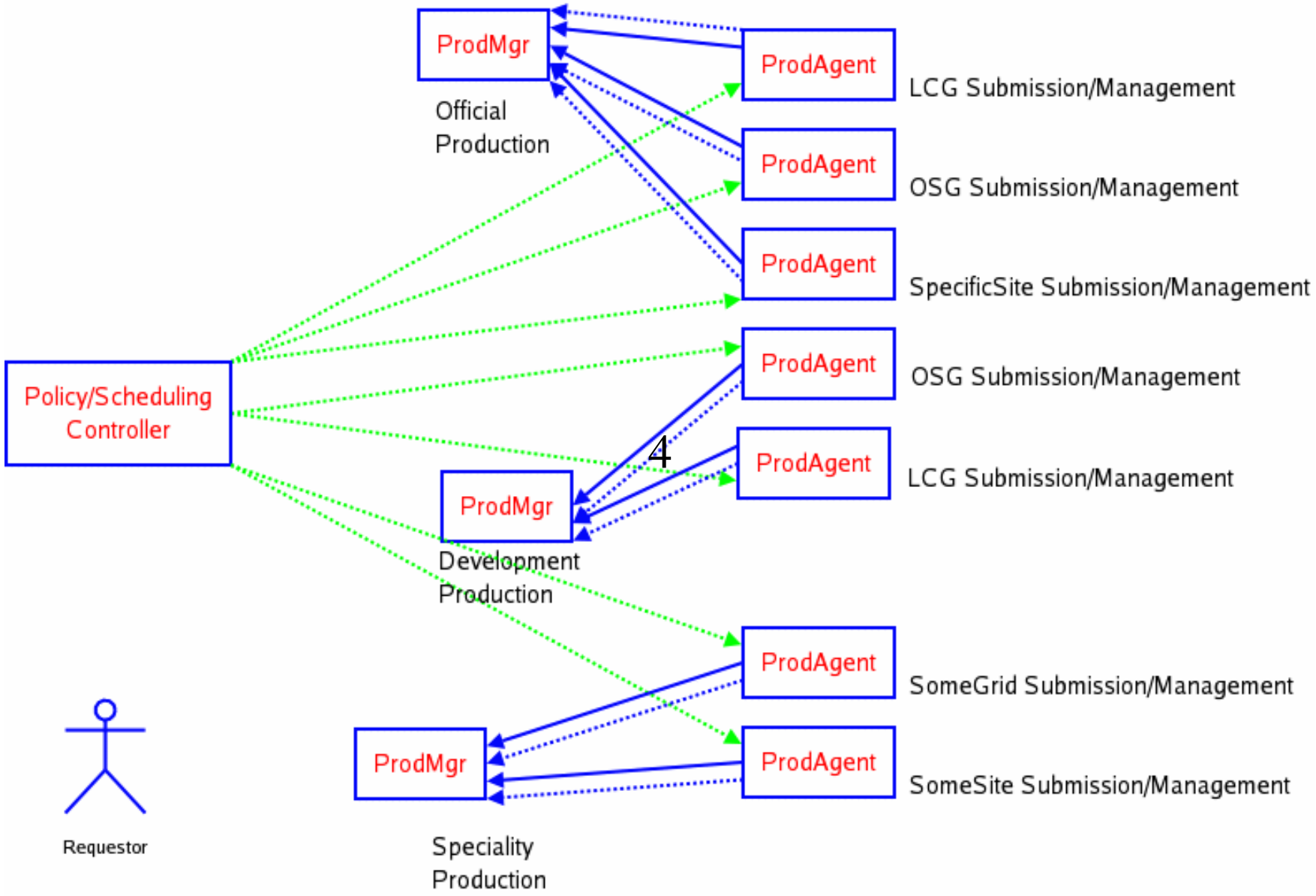
Production Requests

- While we have been deploying the new MC production system with CMSSW and the new DM/EDM (i.e. during CSA06 and the current MC production), the model for making production requests has been the following:
  - Someone creates a working CMSSW cfg file
  - That cfg is checked into CVS (typically in one of the Configuration/XXX packages)
  - The production system takes it from there and some additional information is added to create a "workflow spec" for ProdAgent
- Workflow (spec) – this is basically just a CMSSW cfg file + some additional information like dataset name, LFN namespace to use, number of events, etc. ProdAgent can take a workflow spec and produce the requested dataset.
- ProdAgent teams take a workflow spec, and use it with their ProdAgent.

Production Requests (2)

- The somewhat manual model I described on the previous slide works fine for small number of cfg files, but we need something more to handle larger numbers of requests.
- Two additional components:
  - PRODREQUEST – a web interface for users/analysis groups to introduce their cfg files and make requests and the MC production manager to approve requests
  - PRODMGR – a system for automatically passing production tasks to multiple prodagents

General Architecture



ProdMgr

- Obtains new, approved requsts from PRODMGR
- Allocates jobs to one or more ProdAgents
- Scorekeeping for allocated, completed and failed jobs
- Manages approved workflows for requested MC samples until they are completed
- Normally there will be one of these, at CERN
- We expect to deploy this “in anger” in January, 2007
- Interaction with Policy Manager to prioritize jobs (future)

MC production at Tier-2 sites

- The aim is to have jobs that run 4-8 hours, but we still do have jobs that run 24 hours or more (memory footprint from 500MB to 1GB)
- The MC jobs run and (in general) stage out their output file to the local storage element in the /store/unmerged part of the file namespace.
- A subsequent job will run which merges the output of some number of MC jobs. The resulting merged file will also be staged out to the local storage element and the (unmerged) files from the local storage element.
- The merged file is subsequently moved to another site by PHEDEX. (Currently we only copy it, and remove the file at the site by hand, but we are expecting to deploy a "move" (in the unix sense) in the near future.)
- We expect that for pileup that o(200-250GB) of disk space will be needed to host the pileup (input) sample.

Analysis use of Production System

- "Me, My Friends, The Grid" concept
- Thin out the number of things managed by the user (me) services for managing/tracking jobs/DM, etc. moved typically to Tier-2 (My Friends). This service interacts with the Grid.
- User tools (on laptop/workstation/UI) environment are minimal and the real complexity sits in a "My Friends" site, typically at a Tier-2 to which the user is associated.
- Use CRAB as interface, but PA manages jobs and interacts with DM system
- Type one command to pass task definition to "My Friends", disconnect laptop and go home.
- ProdAgent is being designed to eventually be the core of such a "My Friends" service, with CRAB as user interface.

Questions/discussions

- If you have questions the HyperNews forums to use are the following:

  "MC Production System Operations"

  https://hypernews.cern.ch/HyperNews/CMS/get/mcOps.html

  hn-cms-mcOps@cern.ch

  "MC Production System Development"

  https://hypernews.cern.ch/HyperNews/CMS/get/mcDevelopment.html

  hn-cms-mcDevelopment@cern.ch