

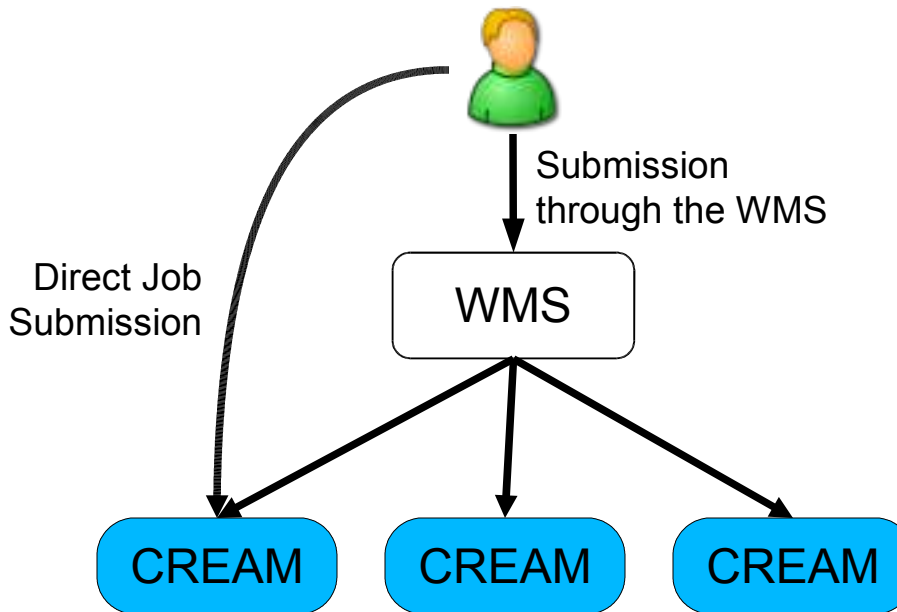
CREAM-WMS Integration

Luigi Zangrando (zangrando@pd.infn.it)

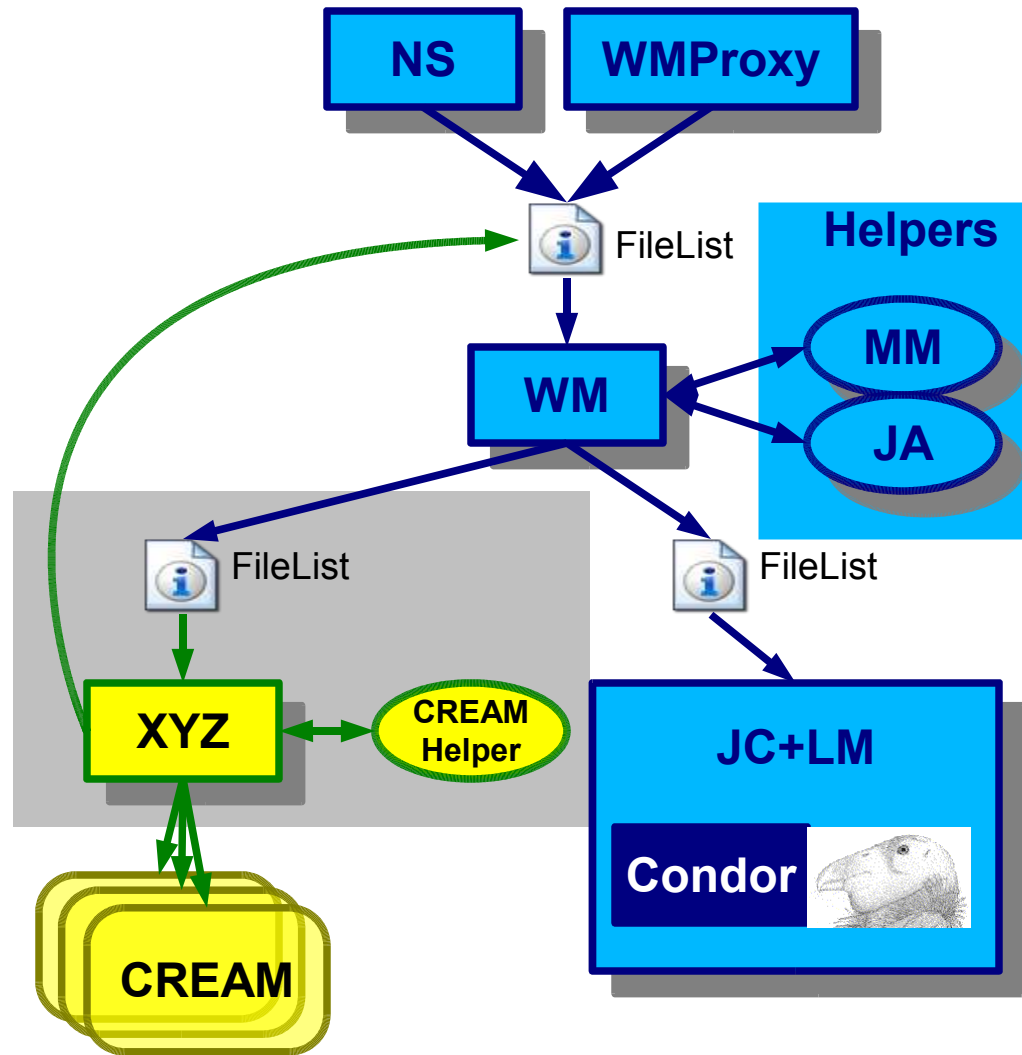
Moreno Marzolla (marzolla@pd.infn.it)

- **We had a meeting in PD between Francesco G., Marco C. and the JRA1-PD team; in the meeting we discussed about CREAM-WMS integration**
 - We also had some preliminary discussions with the JRA1-CZ people about interaction with LB
- **We now present the outcome of the meeting**

- **CREAM should be invoked:**
 - Directly from the UI (direct job submission)
 - Through the WMS

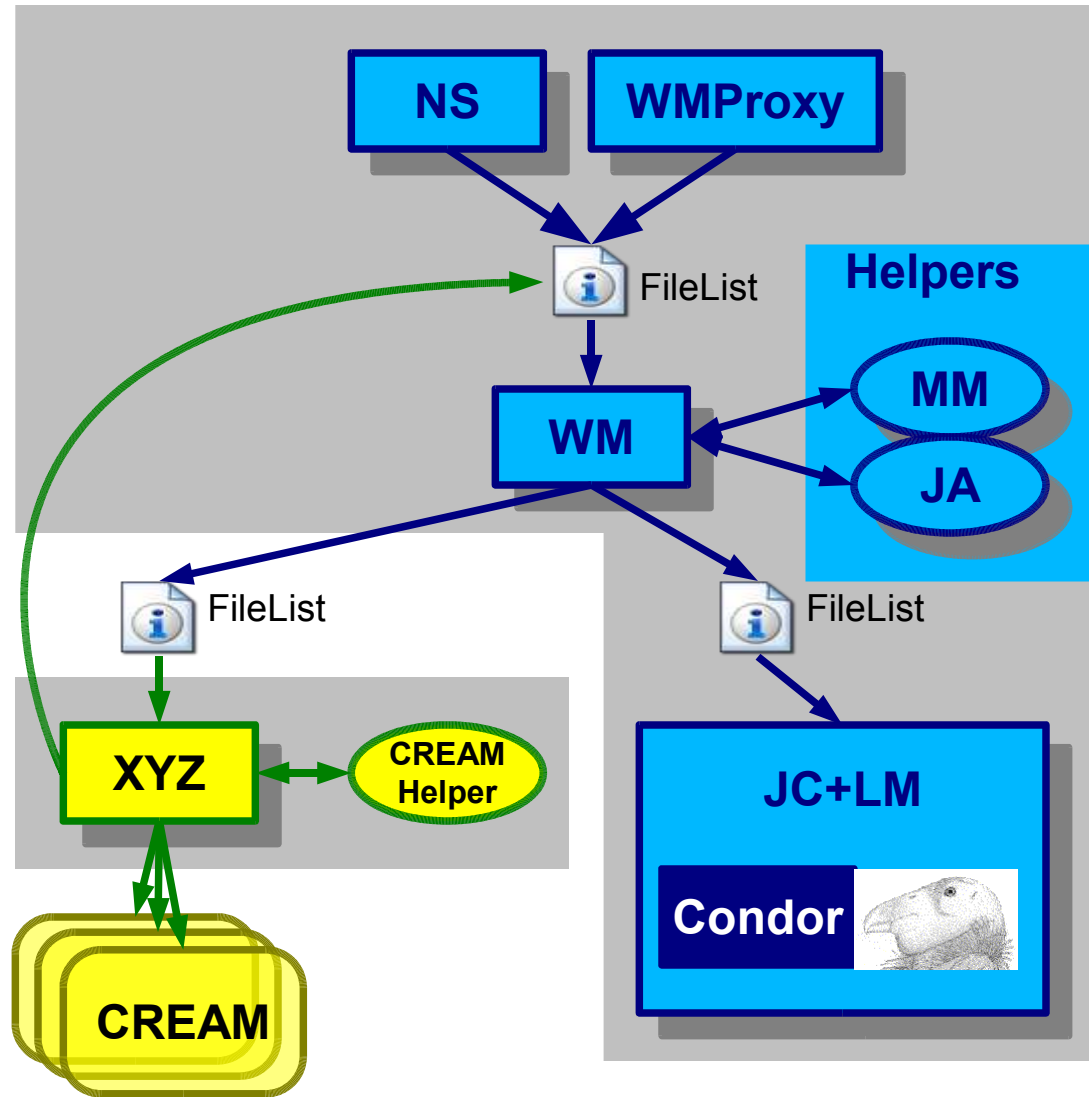


- In analogy with JC+LM, a new component “XYZ” is needed for the interaction with CREAM (not found yet a fancy name)
- XYZ takes the job management requests from its filelist (the new mechanism proposed by Francesco G. is not yet available; in any case the migration should be straightforward)
- The WM chooses the filelist where to put jobs by reading the ce-id (e.g., grid005.pd.infn.it:8443/cream-lsf-grid02); not nice, but don't have a better solution at the moment
- XYZ invokes a CREAM Helper for formatting the JDL to be CREAM-compliant (basically needed to rearrange the input/output sandbox parameters)
- XYZ manages the submission to CREAM (see next slides); XYZ keeps the mapping between the GridjobId and CREAMjobId
- Failed submissions are reinserted into the WM's filelist as in the current implementation

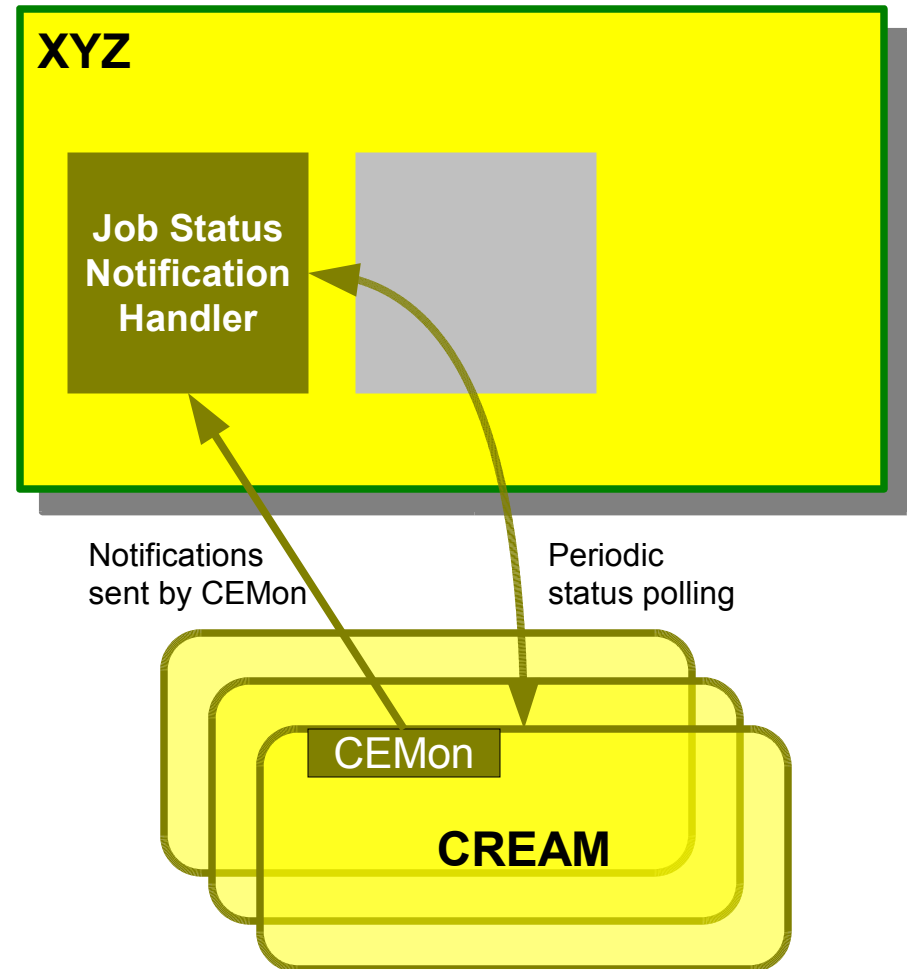


- **XYZ Implementation approach:**

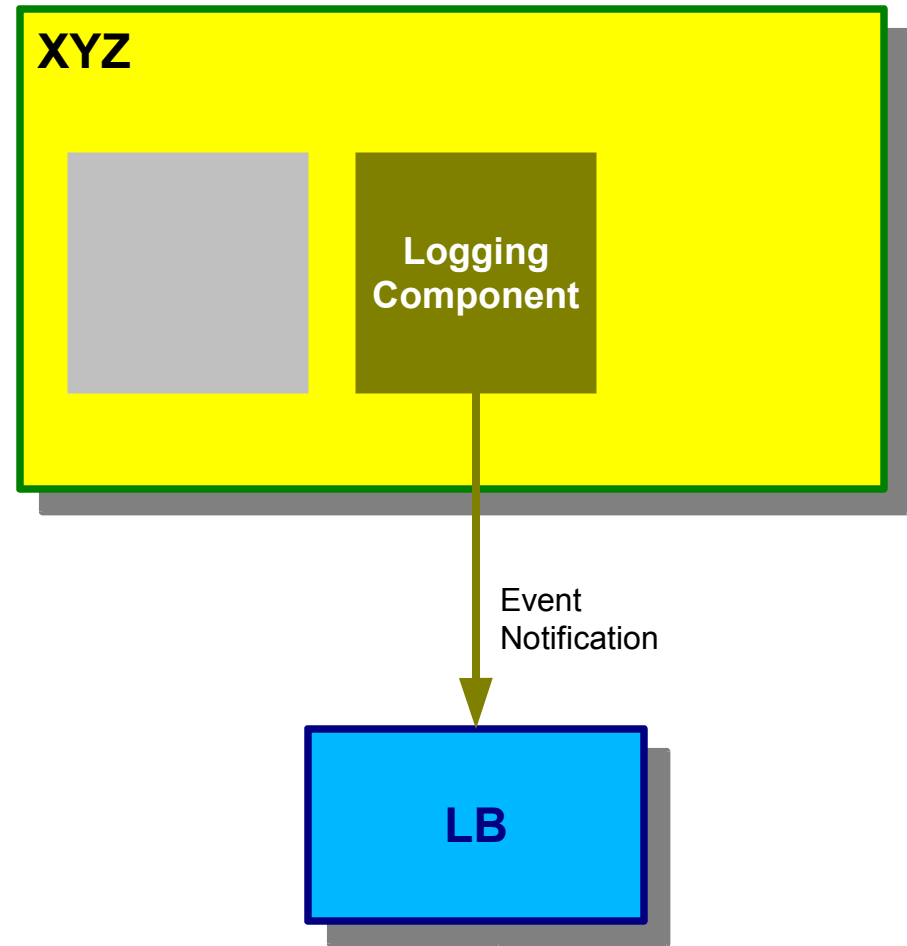
- Requirement: XYZ should have the lowest possible impact on the current WMS architecture (less chance to avoid major changes to existing components); CREAM should be easily “plugged” inside the WMS
- Interim solution: XYZ is an external process which communicates with WM using a filelist
- In the long term we will explore the possibility to make XYZ an internal component (thread) of WM
 - ★ This clearly requires a well-defined API in order not to couple XYZ with WM; there should be an abstraction layer between WM and CE-specific support engines



- A thread of XYZ receives notifications about the job status changes from CEMon closely coupled with CREAM CE
- As a fail-safe mechanism, another thread is needed to poll the status of all jobs still alive
- For each status change, XYZ must do basically what LM does now for job status changes



- The existing LB states are ok also for this WMS-CREAM integration
- Since CREAM supports suspending jobs, a new **SUSPENDED** status would be useful
 - Or can we have a SUSPENDED flag attached to SCHEDULED and RUNNING states is enough? (as suggested by Ales)
- **XYZ will log to LB the same kind of events logged by JC+LM for non-CREAM CEs (e.g., Condor submissions)**
- The Job Wrapper will log to LB as currently done

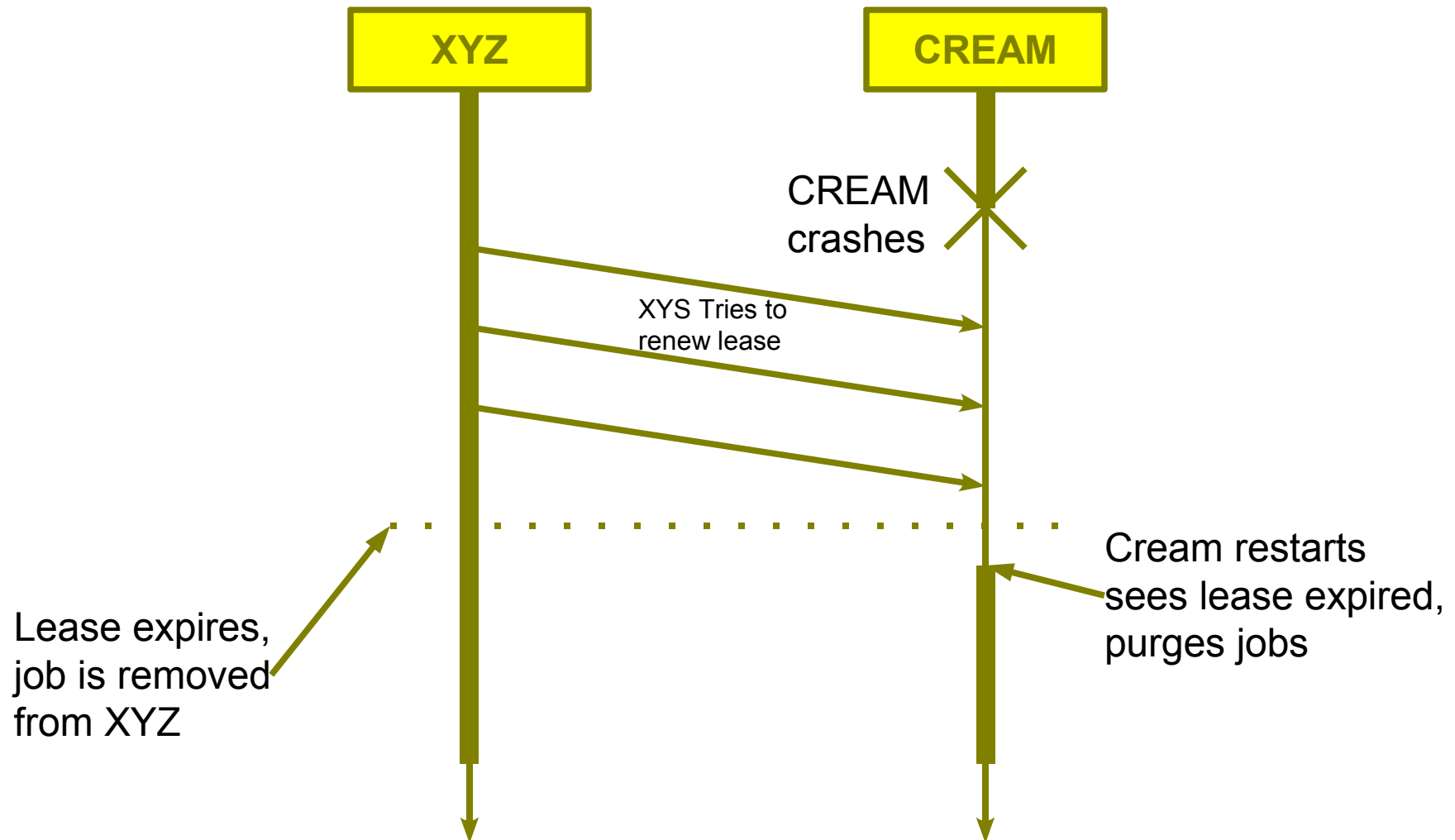


- At the moment they are handled by Condor DAGMan, which is not present in the proposed scenario
- CNAF is exploring a different implementation not using Condor but this will not be finalized in the short term
- Support for DAGs (which will imply support for parametric and collection jobs as well) will be implemented in CREAM
- **Idea: allow submission of a whole DAG directly to the CREAM CE**
 - This is a requirement coming from many users
 - To be studied when a whole DAG should be handled directly by CREAM
 - ★ To start, this could be specified by the user in the jdl

- The protocol between XYZ and CREAM must be reliable and well thought
- There are potential problems when software components or connections fail
 1. XYZ crashes
 2. CREAM crashes
 3. XYZ → CREAM link crashes (logically equivalent to 1 or 2)
- We have to ensure that jobs do not get lost in the system
 - “Zombie” jobs
- We are thinking to implement a *lease*-based mechanism as suggested also by Francesco G.

- Standard mechanism commonly used in distributed systems
- Each CREAM job has an attribute (the *lease*) which is basically the “time to live” of the job
 - When the **lease expires**, the job is removed from the system on both sides (WMS and CREAM)
 - Leases are **renewed** by XYZ as long as XYZ and CREAM can talk to each other

- **XYZ crashes**
 - Data from the WM comes via filelist; no problem here
 - If the lease for a given job is over, jobs have been removed on CREAM
 - When XYZ comes alive, it checks all jobs and renew the leases if necessary; jobs which have been removed by CREAM will be removed by XYZ as well
- **CREAM crashes**
 - CREAM has a simple FS-based journalling mechanism which ensures that no jobs get lost
 - When CREAM comes up again, it checks the leases and purges expired jobs; if the lease is “long enough”, CREAM has the possibility to be restarted in time
- **We are thinking about a utility to renew the lease if the WMS is brought down, e.g., for maintenance**



- When XYZ realizes that a job proxy has been renewed on the WMS node, XYZ must invoke the appropriate proxy renewal function on CREAM
- How does XYZ realize that a proxy has been renewed?
 - Polling or something better?