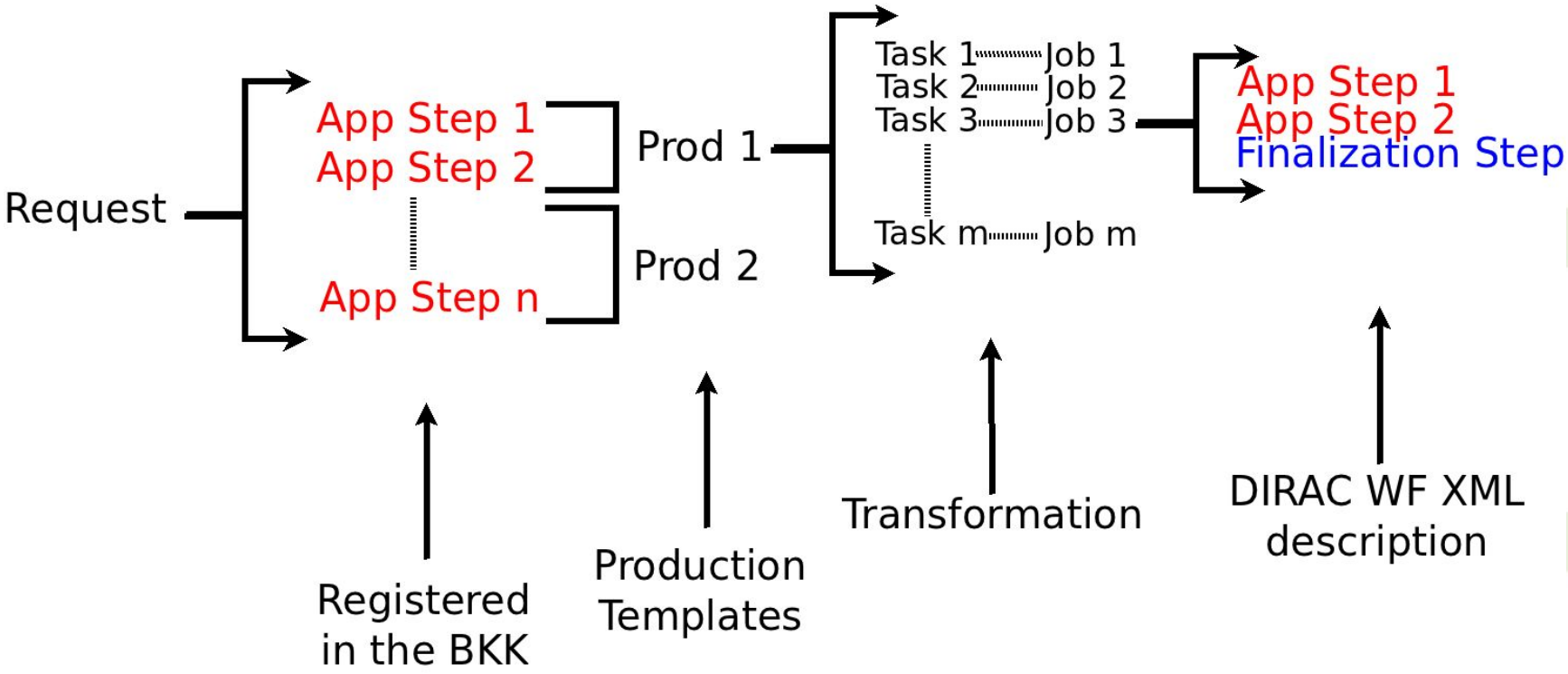
LHCb Production System

Federico Stagni, on behalf of the LHCbDirac team

NB

I prepared these slides in 2012!
for the 3rd DUW

Putting concepts together



DIRAC Workflows

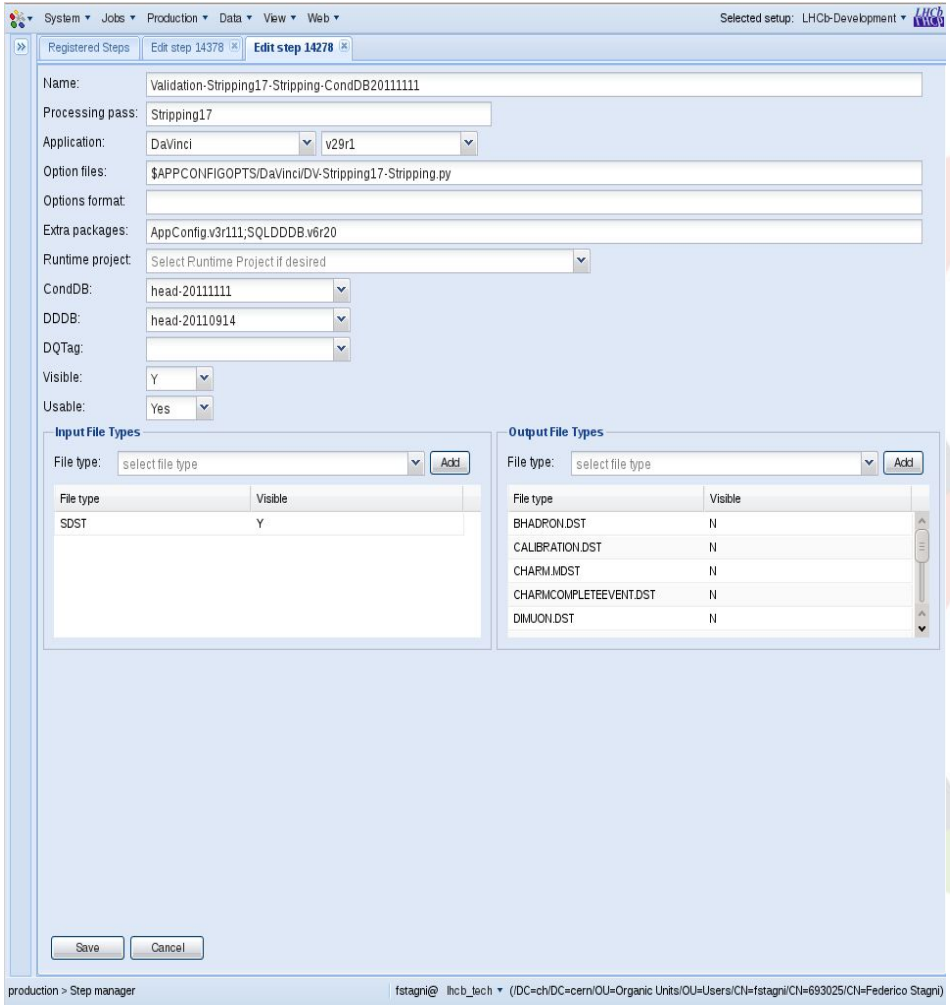
DIRAC user meeting 2011 (click!)

- Xml ↔python dict
- A workflow connects steps together
- dirac-jobexec aWorkflow.xml

Jdl:

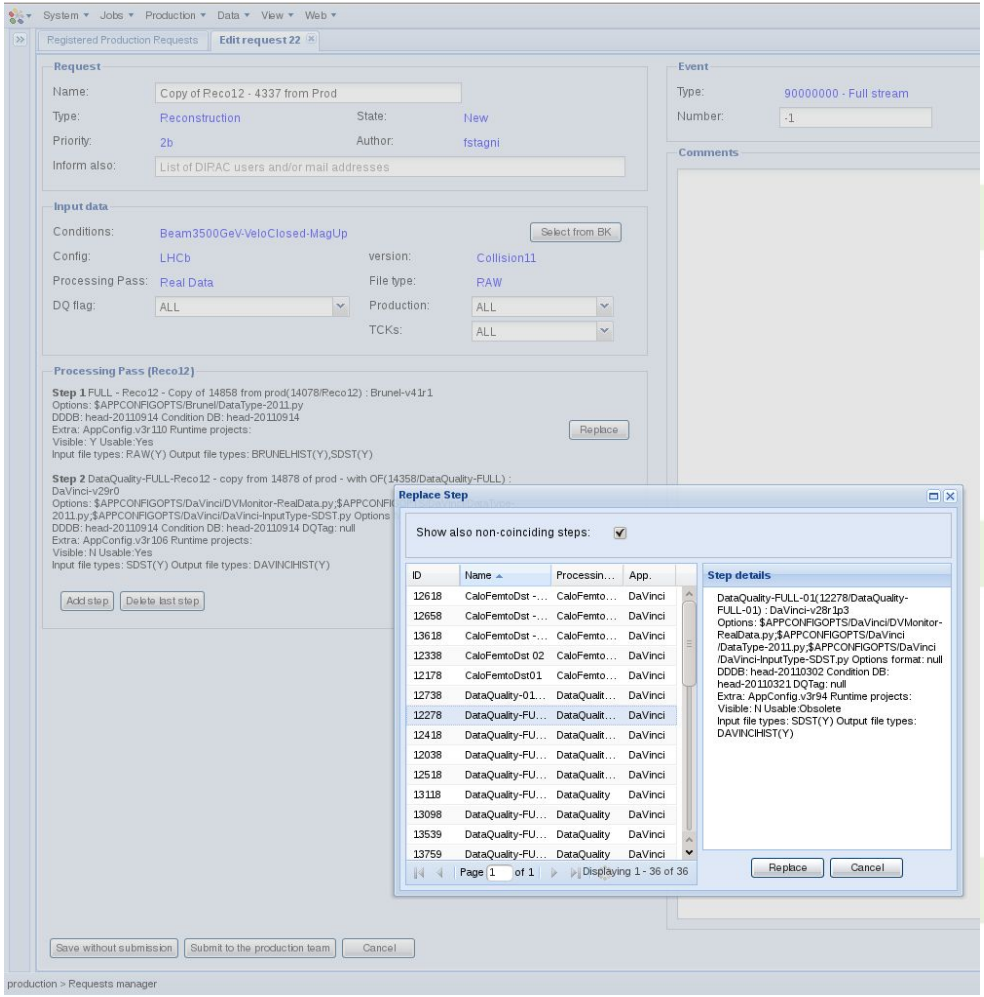
- Executable = "$DIRACROOT/scripts/dirac-jobexec";
- Arguments = "jobDescription.xml -o LogLevel=verbose";

Production Request System

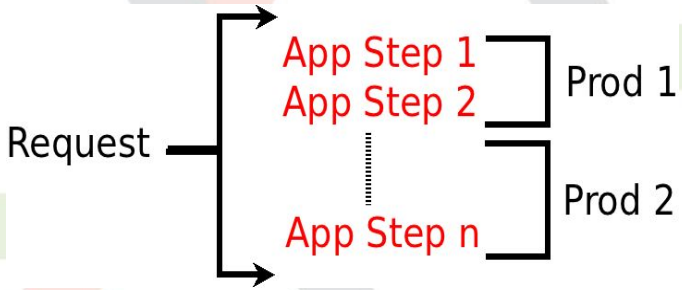


- Application Managers defines application steps
- “What to run” to go from X to Y
- LHCb application
- A step “translates” in a workflow application step

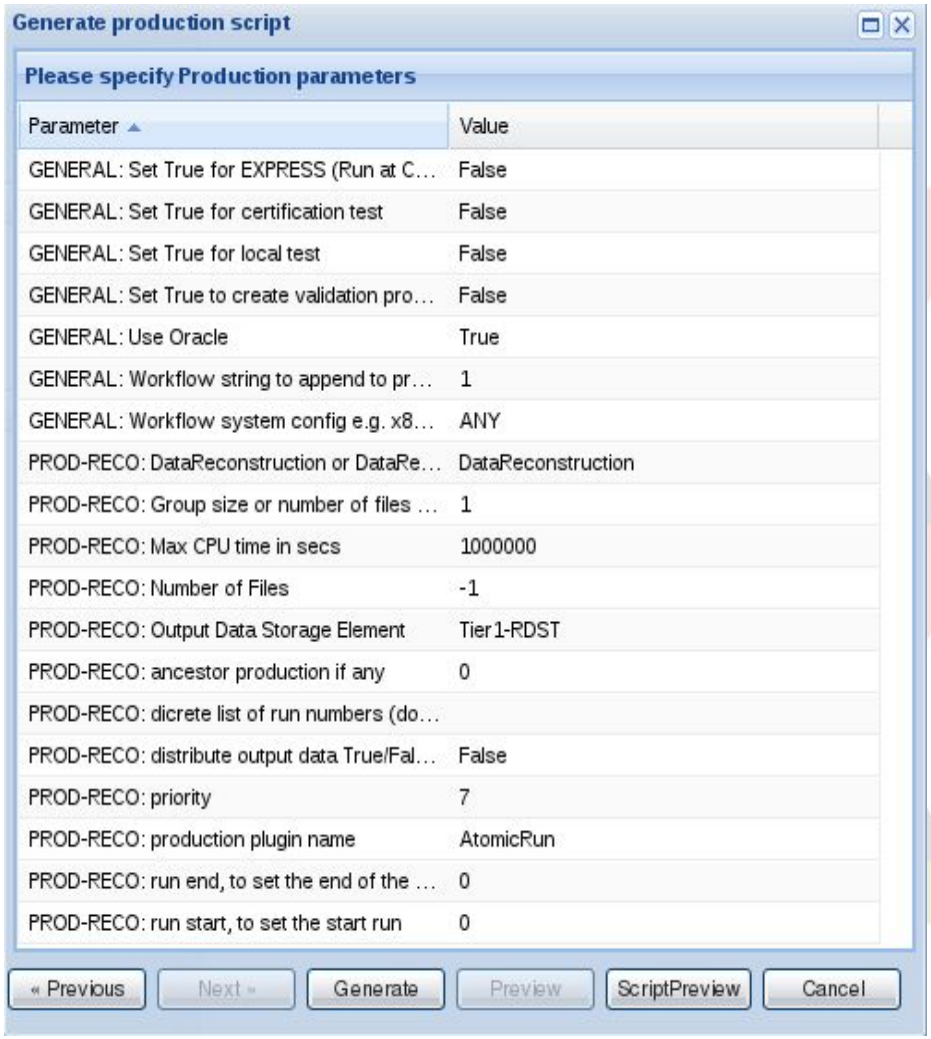
Production Request System /2



- Steps are combined in production requests (e.g. MC, or Reconstruction)



Production Request System /3



- Production requests are submitted using production templates
- e.g.: priority, which plugin, where the outputs are stored, DIRAC CPU, etc.
- Each production is created using the Production API

LHCbDirac TS

- Extension of the DIRAC TS, mostly for interacting with the BKK
  - DB:
    - Physics RUNs information
    - BKK queries (supersedes TransformationInputDataQuery)
  - Service and clients extended for the DB extension

LHCbDirac TS /2

- Agents
  - BookkeepingWatchAgent
    - Looks for BKK queries, and fills the TransformationFiles table
    - Threaded, uses pickle file for caching
  - DataRecoveryAgent
    - Resets input files in “Unused” status, in case the jobs failed
    - A counter is kept, with a maximum of re-trials
  - Extensions for cleaning, and closing productions

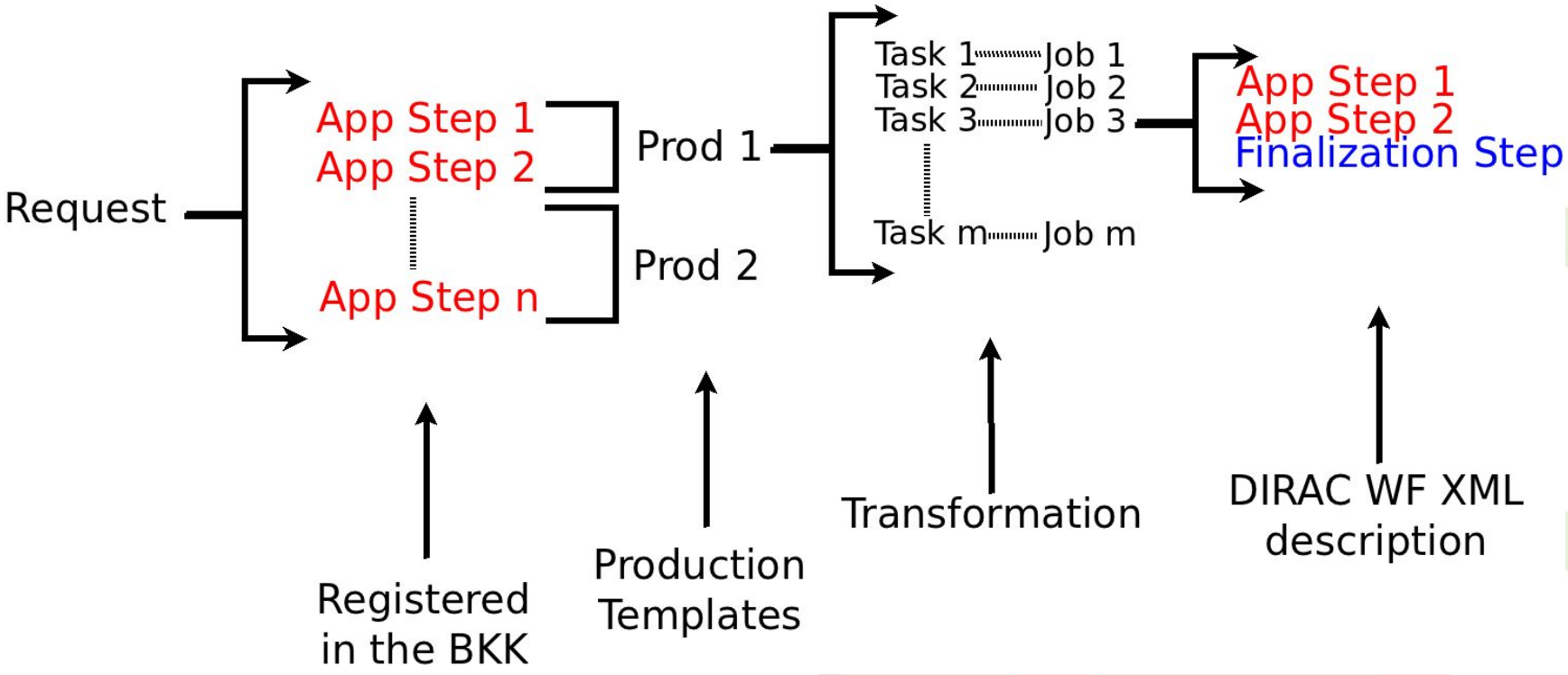
LHCbDirac TS /3

- Plugins (LHCbDIRAC.TransformationSystem.Agent.TransformationPlugins)
  - Many LHCb plugins coded
    - e.g. ByRun,with flushing...
    - This is where you want to extend

Production API

- Expose functionalities to connect together TS, BKK and Production Request System
- Use LHCbJob.py (extension of DIRAC.Interfaces.API.Job.py) to create a DIRAC workflow, whose xml is uploaded to the Transformation DB
- python modules are run within the workflow, grouped within steps. Application steps and Finalization steps are present

Putting concepts together



WRT to the proposed implementation

- We have in ProductionRequestDB a table for ProductionRequest description
- The association ProductionID -> TransformationIDs is in the “TransformationFamily” field in TransformationDB/Transformations
- We have an agent for updating the transformations and the productions status, and a state machine