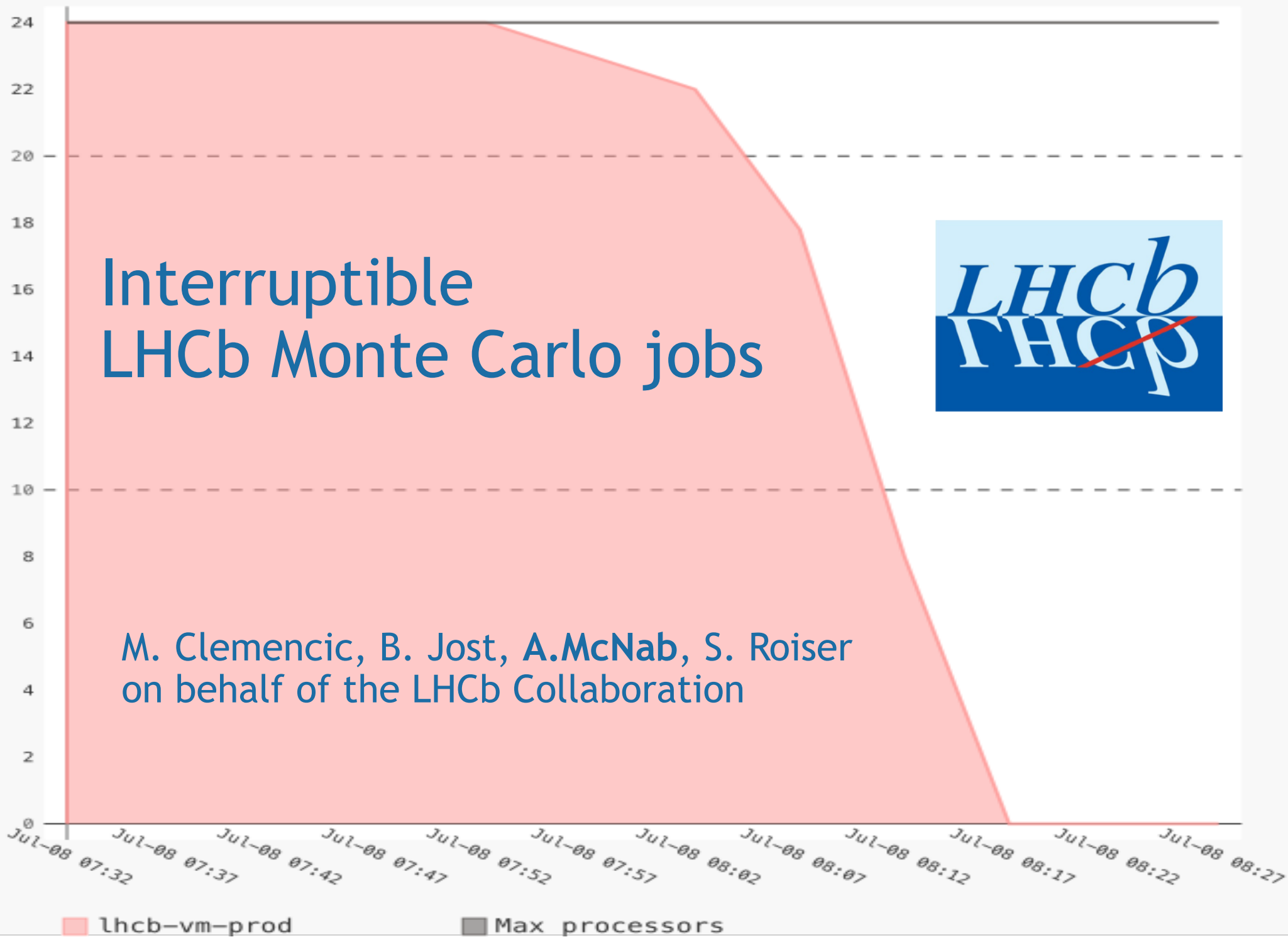# Interruptible LHCb Monte Carlo jobs

M. Clemencic, B. Jost, **A.McNab**, S. Roiser
on behalf of the LHCb Collaboration

Overview

- Monte Carlo jobs are the bulk of LHCb's offline workload

- LHCb has used "elastic" MC jobs for several years

  - Estimate job time limit and CPU power and calculate how many events will fill the slot

- Interruptible MC jobs go further: dynamically stop when time runs out or when external events mean we need to release the slot

- Applications in "job masonry"

  - backfilling the High Level Trigger farm

  - when sites need to reclaim nodes for kernel updates etc

  - filling n-processor slots with multiple single processor jobs

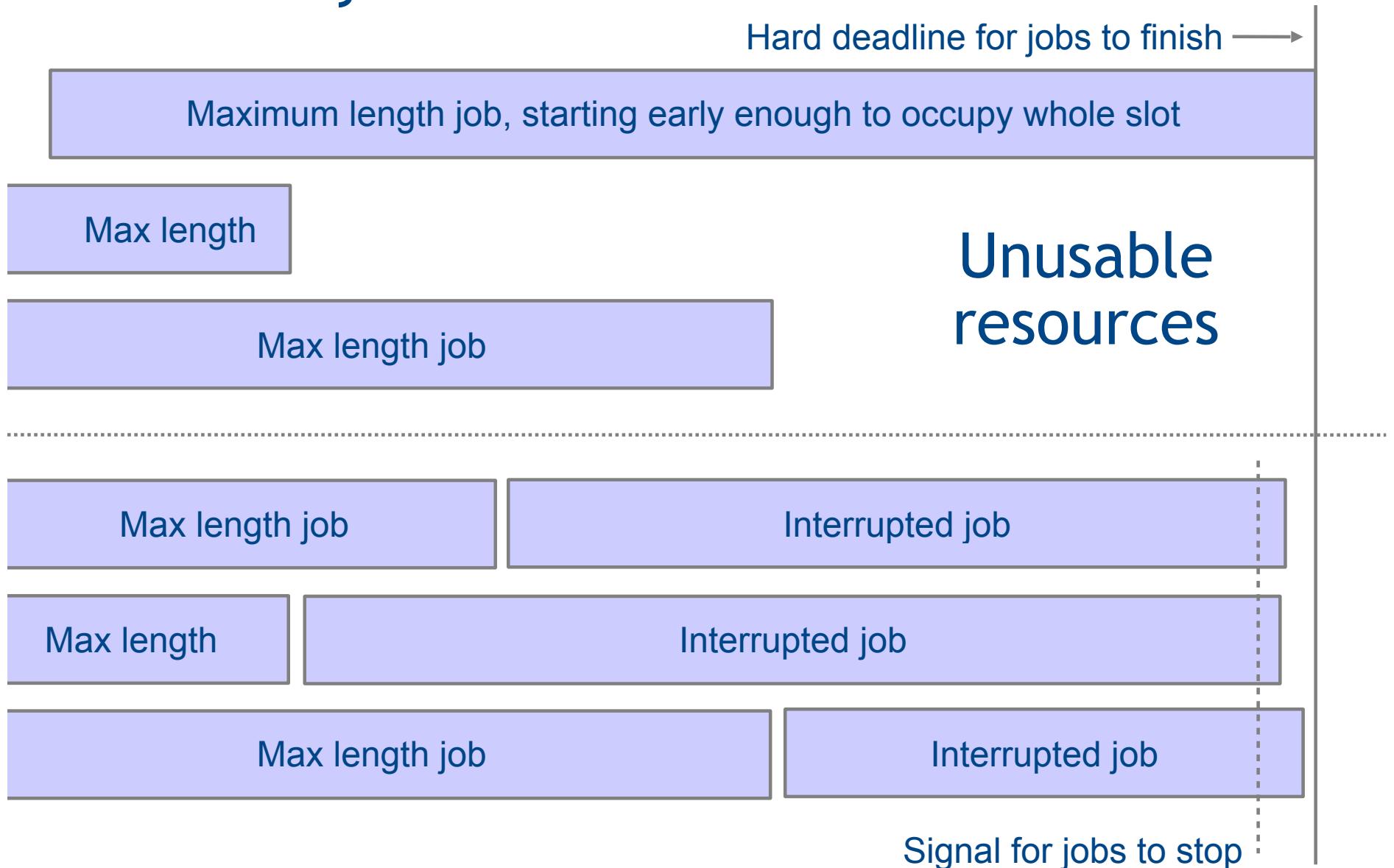- And Manchester "eviction" work with Vcycle and/or OpenStack

LHCb Monte Carlo

- The LHCb Monte Carlo application has an event loop with a configurable Unix signal handler which will stop processing **once the current event is done**

- We've used the SIGUSR1 signal for the mechanisms described in this talk

  - SIGXCPU is "for" this, but can also be sent by batch systems and we want to be able to identify where the signal has come from

- If the MC is interrupted like this, then it is treated as if fewer events were planned at the start of the job

- The production system expects jobs with varying numbers of events so not a problem

The Masonry Problem…

The Masonry Problem

Hard deadline for jobs to finish ⟶

Maximum length job, starting early enough to occupy whole slot

Max length

Unusable resources

Max length job

Max length job

Interrupted job

Max length

Interrupted job

Max length job

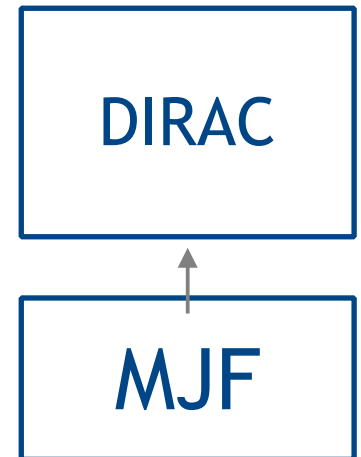Interrupted job

Signal for jobs to stop

High Level Trigger farm

- LHCb has run Monte Carlo on HLT machines for several years

  - We run DIRAC pilots on the machines, under the control of the online shift team

  - Able to run MC during technical stops this way

- In 2017 the HLT team added the ability to use the MC signal handling to interrupt running MC jobs cleanly from the online farm controls when HLT jobs slots needed again

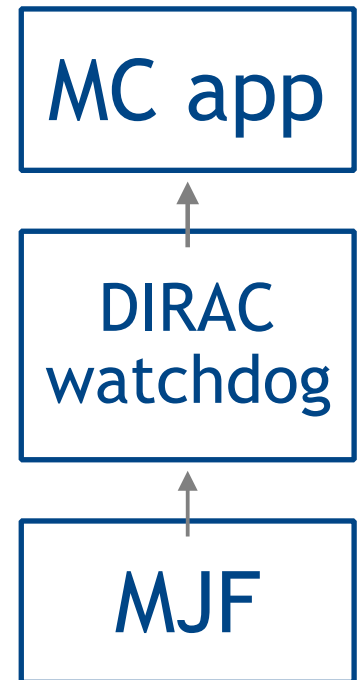- See Joel Closier's HLT talk for more details

Signal
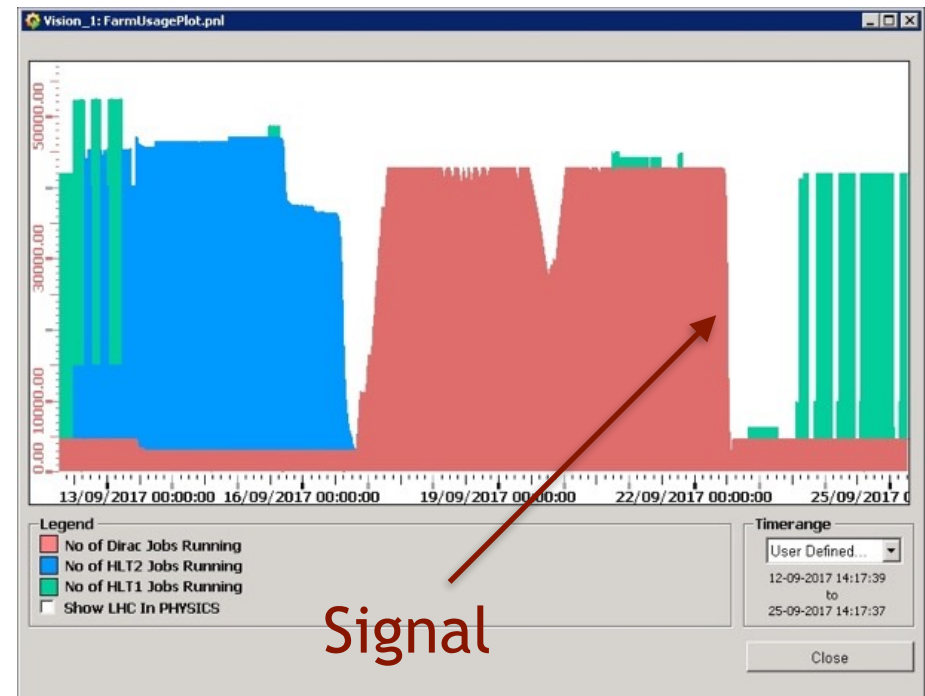
Machine/Job Features

- DIRAC jobs can (imperfectly) discover how long their job slot is

  - either using the WLCG Job Features mechanism

  - or by running batch system commands like qstat

- The machine features and job features mechanisms can also provide shutdown_time which can be used to interrupt jobs after they have started

- In 2017 an MJF module was added to DIRAC

  - presents a consistent estimate for the DIRAC scripts which set up the environment for jobs

  - Including any dynamically created shutdown_time limit

```
┌─────────┐
│  DIRAC  │
└─────────┘
     ↑
┌─────────┐
│   MJF   │
└─────────┘
```

DIRAC Watchdog

- Each payload job that DIRAC runs has a watchdog associated with it

  - Checks include whether the job has run out time

  - Either because job slot end is approaching or because of interruption from the site

- The Watchdog has been extended to use the time left information and send an interruption signal

  - This is configured within the payload job's JDL in a generic way - not necessarily for LHCb Monte Carlo

  - Specify the signal number, processes to send it to, and grace period

```
┌──────────────┐
│              │
│    MC app    │
│              │
└──────────────┘
        ↑
┌──────────────┐
│    DIRAC     │
│   watchdog   │
│              │
└──────────────┘
        ↑
┌──────────────┐
│              │
│     MJF      │
│              │
└──────────────┘
```

Production evaluation with VMs

- This "StopSig" mechanism has been used in production Monte Carlo jobs running in DIRAC VMs

- Example with a draining scenario

  - Pick shutdown time for the VMs after the VMs have started

  - Create MJF shutdown_time file on (most) hosts

  - Grace period is 30 minutes

    - Reflected in the distribution of stop times
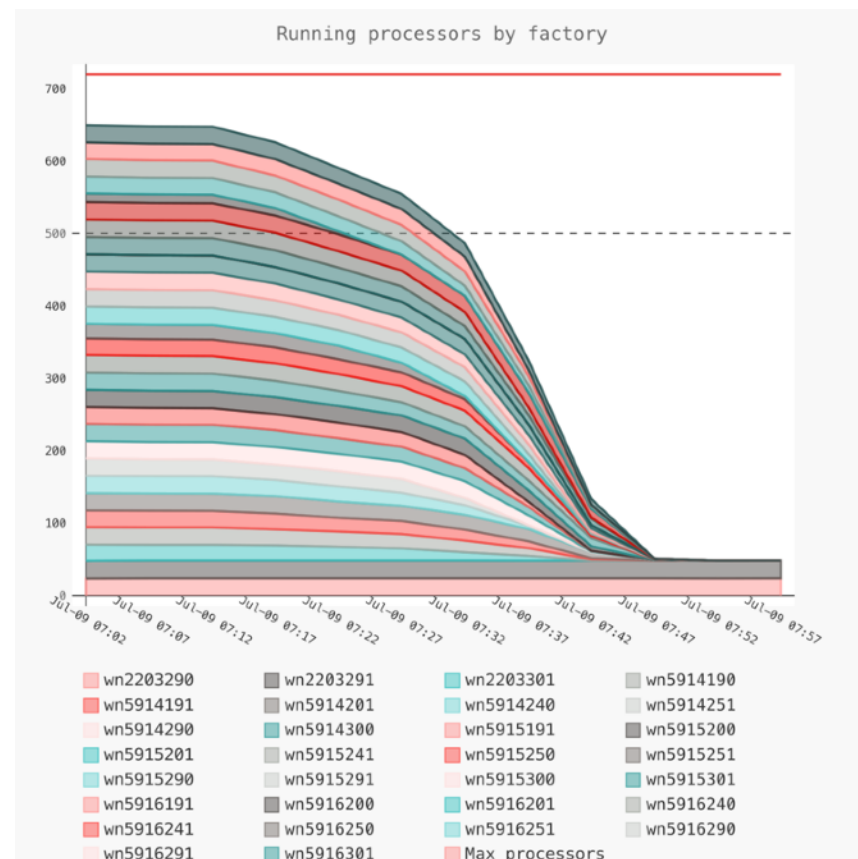
Full deployment

- We will be able to deploy this mechanism across all grid sites when LHCb adopts DIRAC v6r20 in the next few weeks

- This is limited to Monte Carlo jobs though

  - The StopSig options added to MC JDLs

  - But that's the bulk of LHCb workloads though

- Our intention is to submit MC jobs with short time estimates so they can still be matched into short slots as time limits approach

  - But longer duration estimates for data processing and user jobs so they only match into "full size" job slots

- Once this is done, sites will be able to use Machine/Job Features to specify a time to drain by

  - MC jobs will carry on doing useful work up to that time

Summary

- Monte Carlo application's mechanism to stop after simulating the current event has been brought into production usage

- Used routinely on HLT to fit MC into short periods of availability
  - Under the control of the online shift team

- The DIRAC Watchdog has been extended to send the interruption signals in response to job slot limits and shutdown time from MJF

- Evaluations of this with production MC jobs have been done with jobs running in VMs
  - Draining scenarios, and synchronising single processor in 8-processor slots scenarios

- Once the next version of DIRAC is deployed (currently in certification) then this will be available to all sites