# STAR Experience with Automated High Efficiency Grid Based Data Production Framework at KISTI/Korea

HEPiX Spring 2015 Workshop

Authors:

Lidia Didenko

Levente Hajdu

Jerome Lauret

Coauthors:

Wayne J. Betts

Seo-Young Noh

Jaikar Amol

Photo Taken By Levente Hajdu 2013

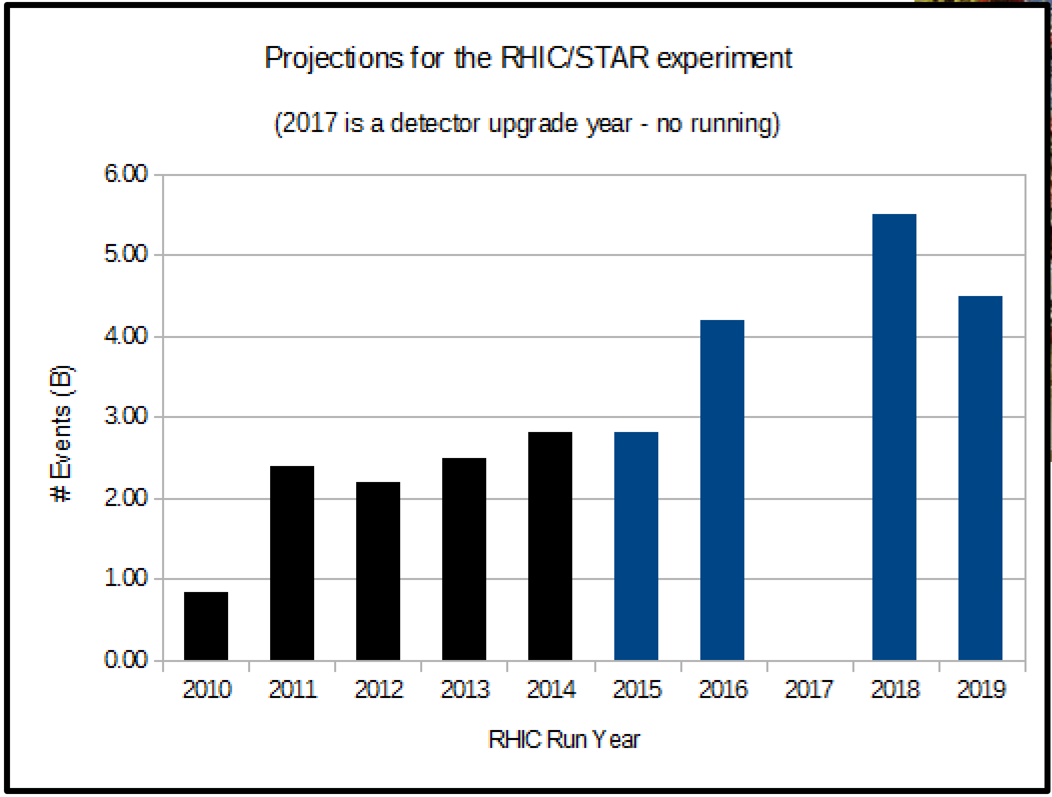
# Intro

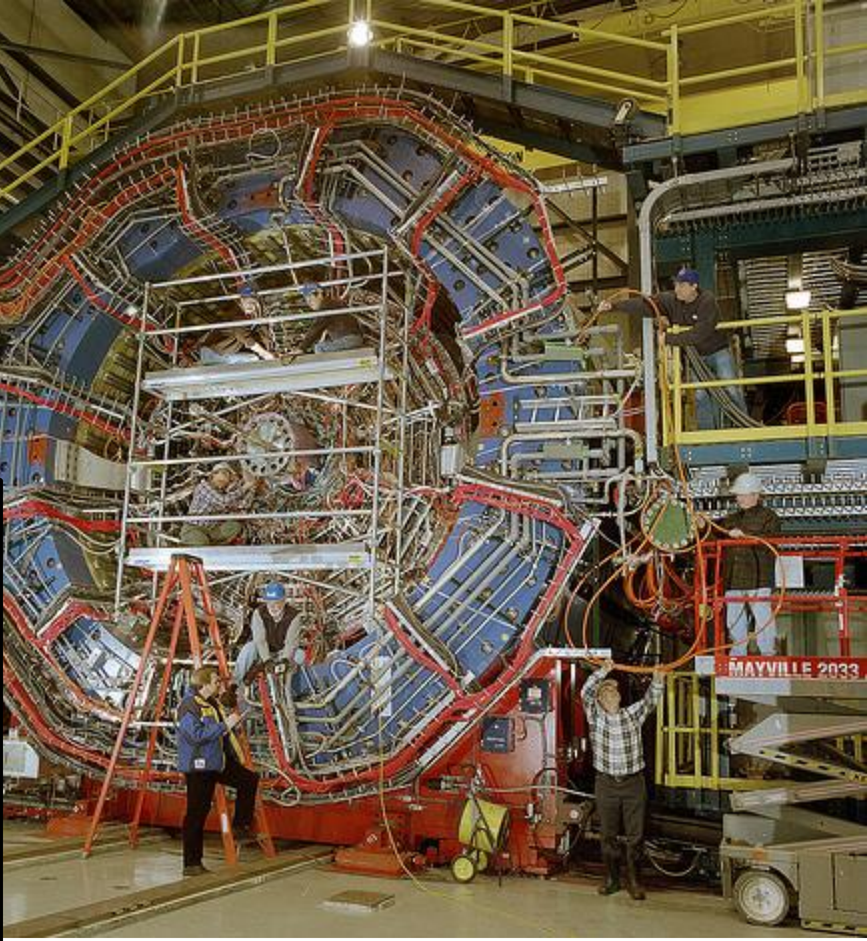
- Automated Real Data Reconstruction Over A Grid Framework
  - Grid based simulation has been running (in STAR) on a routine basis. With little input and modest output, such jobs are perfectly fit for grid operations
  - However STAR takes a few billion real events every year and has to process them in a timely fashion, so there is demand for supplementing real production with external resources
- Objectives:
  - Explore the practicality of offloading part of real data production to a STAR Tier-1 site
  - Investigate staging data from tape to the remote worker nodes and transferring the results back to BNL (STAR policy)
  - Utilize the STAR site at KISTI for data production
- Outline:
  - Framework structure
  - Statistics and outcomes
  - Future work and improvements

# What is STAR ?

- STAR is a detector located in one of the interaction regions of the RHIC (Relativistic Heavy Ion Collider)

- Took its first data in year 2000 - currently on our 15th physics run (year of data taking).





- *Black = already known*
- *Blue = incoming / projected*

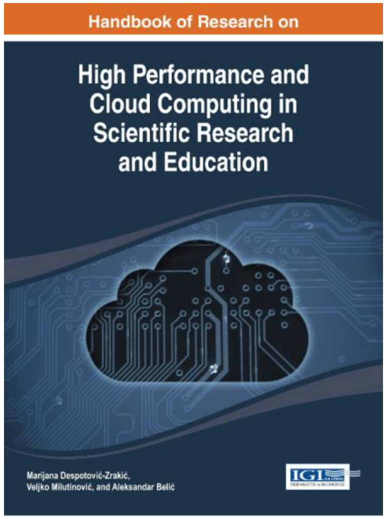
# Working on Grids and Clouds for 15 Years

- STAR has a long experience testing and offloading productions to a diversity of platforms (Cloud/Grid) and resources (Amazon, Universities, National Labs, ...) which has made us face different challenges and provided useful lessons learned.
- "**High Performance and Cloud Computing in Scientific Research and Education**" Chapter 13, IGI Global, Levente Hajdu, Jérôme Lauret, Radomir A. Mihajlovic ISBN13: 9781466657847, ISBN10: 1466657847, EISBN13: 9781466657854
- "**Offloading peak processing to virtual farm by STAR experiment at RHIC**", Proc. of the 14th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT2011), Uxbridge, West London, United Kingdom, September 5-9, 2011, J. Phys. Conf. Ser. 368 (2012) 01211.
- "**When STAR meets the Clouds: Virtualization & Cloud computing experiences**", Proc. of the 18th International Conference on Computing n High Energy and Nuclear Physics (CHEP2010), Taipei, Taiwan, October 18-22, 2010, J. Phys. Conf. Ser. 331 (2011) 062016.
- "**Contextualization in practice: The Clemson experience**", Proc. of the 13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT2010), Jaipur, India, February 22-27, 2010, Pos ACAT2010 (2010) 027.
  Chapter 13 Grids, Clouds, and Massive Simulations
- "**Integrating Xgrid into the HENP distributed computing model**", Proc. of the International Conference on computing in High Energy and Nuclear Physics (CHEP07), Victoria, British Columbia, Canada, September 2-7, 2007, J. Phys. Conf. Ser. 119 (2008) 072018.

Comparison of Cloud Approaches

Overview and Perspectives from STAR

Levente HAJDU
Lidia DIDENKO
Jérôme LAURET
Doug OLSON



Handbook of Research on

High Performance and Cloud Computing in Scientific Research and Education

Marijana Despotović-Zrakić, Veljko Milutinović, and Aleksandar Belić

When STAR meets the Clouds – Virtualization & "Cloudified" grids experience

Jérôme Lauret
RHIC/STAR collaboration

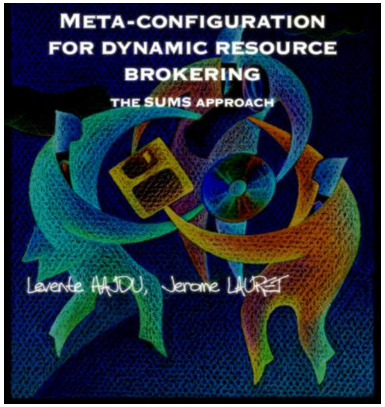
*Authors:* Jérôme Lauret, Matthew Walker, Sebastien Goasguen
*Co-authors:* Lance Stout, Micheal Fenn, Jan Balewski, Levente Hajdu, Kate Keahey, ...

Workflow Generator and Tracking at the Rescue of Distributed Processing.

Automating the Handling Grid Production for the STAR Collaboration

LAURET, Jérôme
HAJDU, Levente (Presenter)
DIDENKO, Lidia

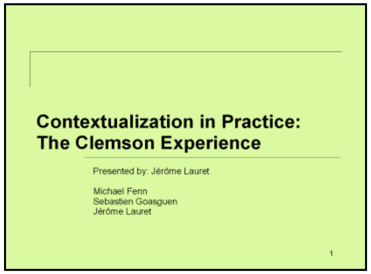
Computing in High Energy and Nuclear Physics
Prague | Czech Republic | 21 - 27 March 2009
BROOKHAVEN NATIONAL LABORATORY
Office of Science
STAR



META-CONFIGURATION FOR DYNAMIC RESOURCE BROKERING

THE SUMS APPROACH

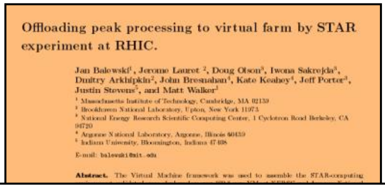
Levente HAJDU, Jerome LAURET



Contextualization in Practice: The Clemson Experience

Presented by: Jérôme Lauret

Michael Fenn
Sebastien Goasguen
Jérôme Lauret



Offloading peak processing to virtual farm by STAR experiment at RHIC.

Jan Balewski[1], Jerome Lauret [2], Doug Olson[3], Iwona Sakrejda[3], Dmitry Arkhipkin[2], John Bresnahan[4], Kate Keahey[4], Jeff Porter[3], Justin Stevens[5], and Matt Walker[1]

[1] Massachusetts Institute of Technology, Cambridge, MA 02139
[2] Brookhaven National Laboratory, Upton, New York 11973
[3] National Energy Research Scientific Computing Center, 1 Cyclotron Road Berkeley, CA 94720
[4] Argonne National Laboratory, Argonne, Illinois 60439
[5] Indiana University, Bloomington, Indiana 47408

E-mail: balewski@mit.edu

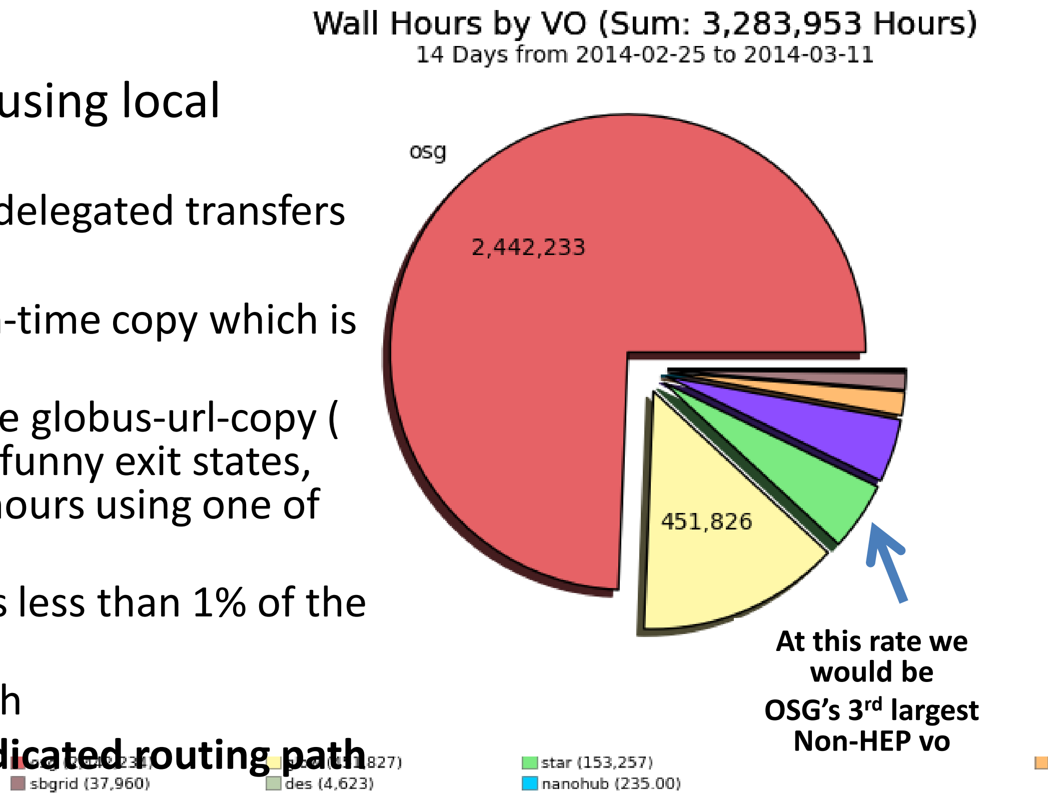
**Abstract.** The Virtual Machine framework was used to assemble the STAR-computing

- KISTI joined as a STAR institution in 2008
- Provided 1,000+ dedicated slots running the HTCondor batch system with OSG gatekeeper
- Implementation choices derived from site constraints
- Accessed via a handful of local accounts (not open to all STAR users )
- No user help desk & support
  - Implementation needs to avoid using local services requiring maintenance
    - Use in-job-run-time transfers vs. delegated transfers
    - Reuse existing components
    - We have hardened an in-jobs-run-time copy which is working at very high reliability
    - Identified many error states of the globus-url-copy ( freezes, crashes, “GK not found”, funny exit states, ….), delay and retry for up to 24 hours using one of three randomly selected GKs
    - Copy time (input and output) was less than 1% of the jobs run time
    - Enhanced / fine tune transfer path
    - **Thanks to ESnet and Kreonet dedicated routing path**

KISTI www.kisti.re.kr

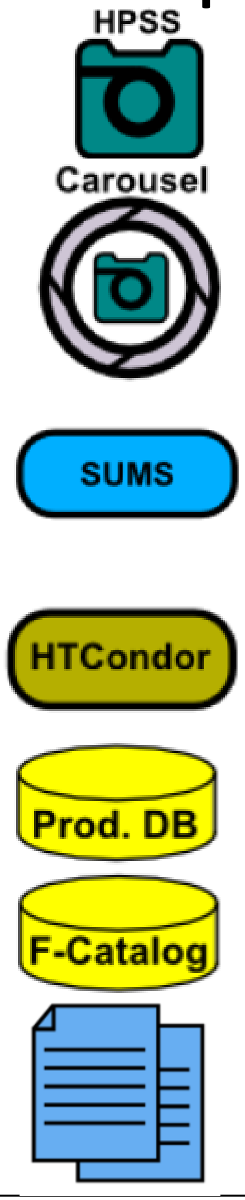
*(Korea Institute of Science and Technology Information )*

# Non-HEP VOs by vo



**At this rate we would be OSG’s 3rd largest Non-HEP vo**

# Components of the Grid Production System



**HPSS** = Tape silo system at BNL

**Data Carousel** = STAR tool for queueing and optimizing requests for the restoration of files from tape by minimizing mount and dismount cycles through reordering

**SUMS (STAR Unified Meta Scheduler)** = SUMS provides a unified interface for submitting jobs to sites and wrapping of the input file and user executable into a job. Feeding can also be turned on to limit the number of jobs submitted at one time.

**HTCondor with Globus** = provides authentication of users between sites and a mechanism to interface with the sites local batch system.

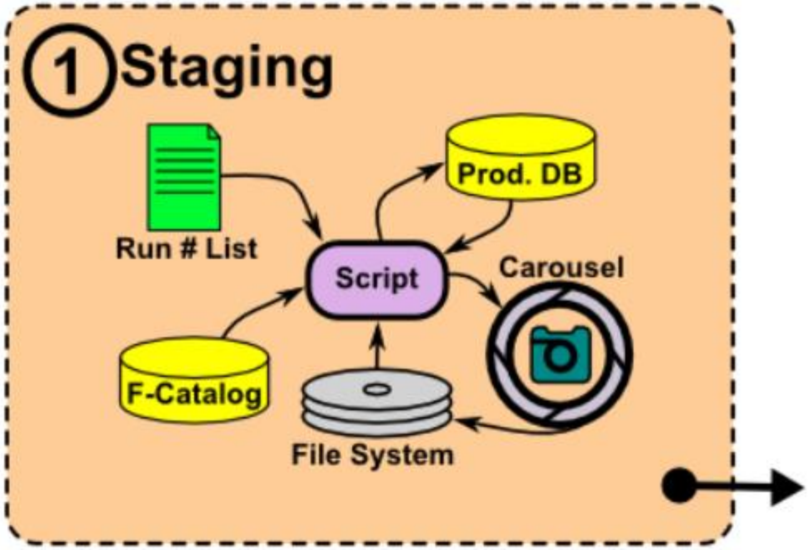
**Production Database** = Database for holding the state of each job

**STAR File Catalog** = includes PFN, LFN and MetaData

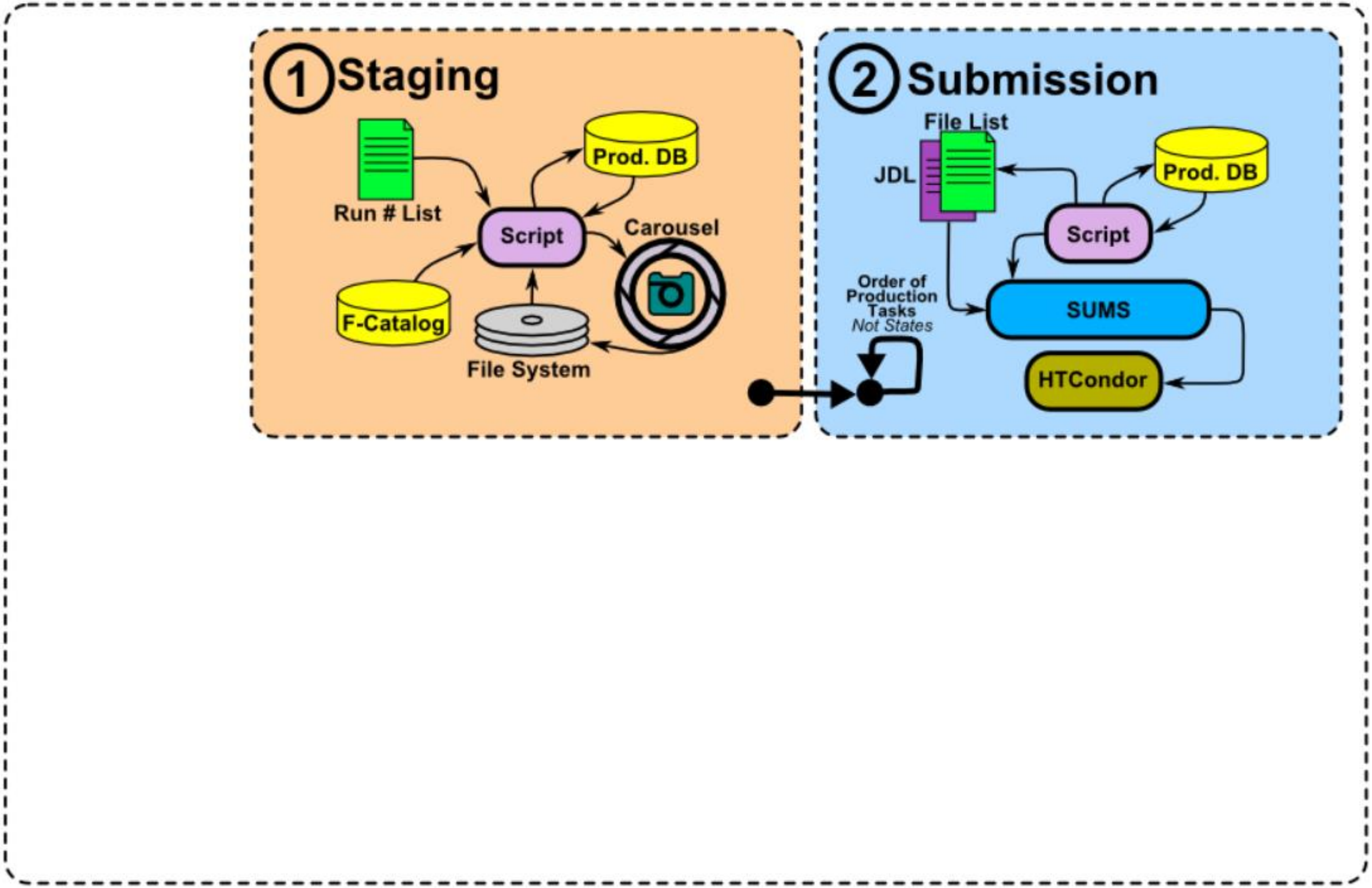
***.Daq Files** = raw detector input files for reconstruction

***.MuDst Files** = reconstruction output files

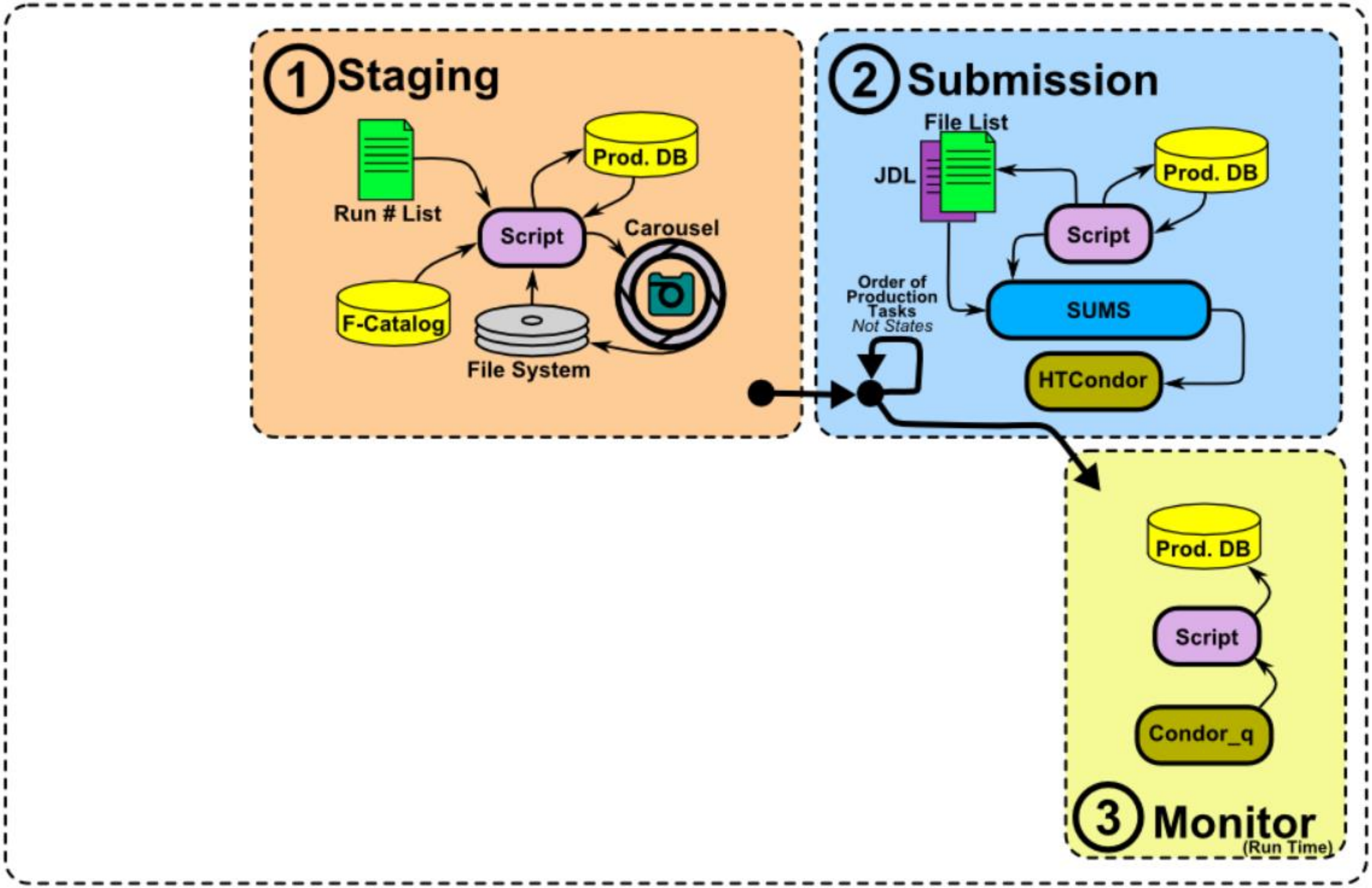
# Detailed Stages of Production



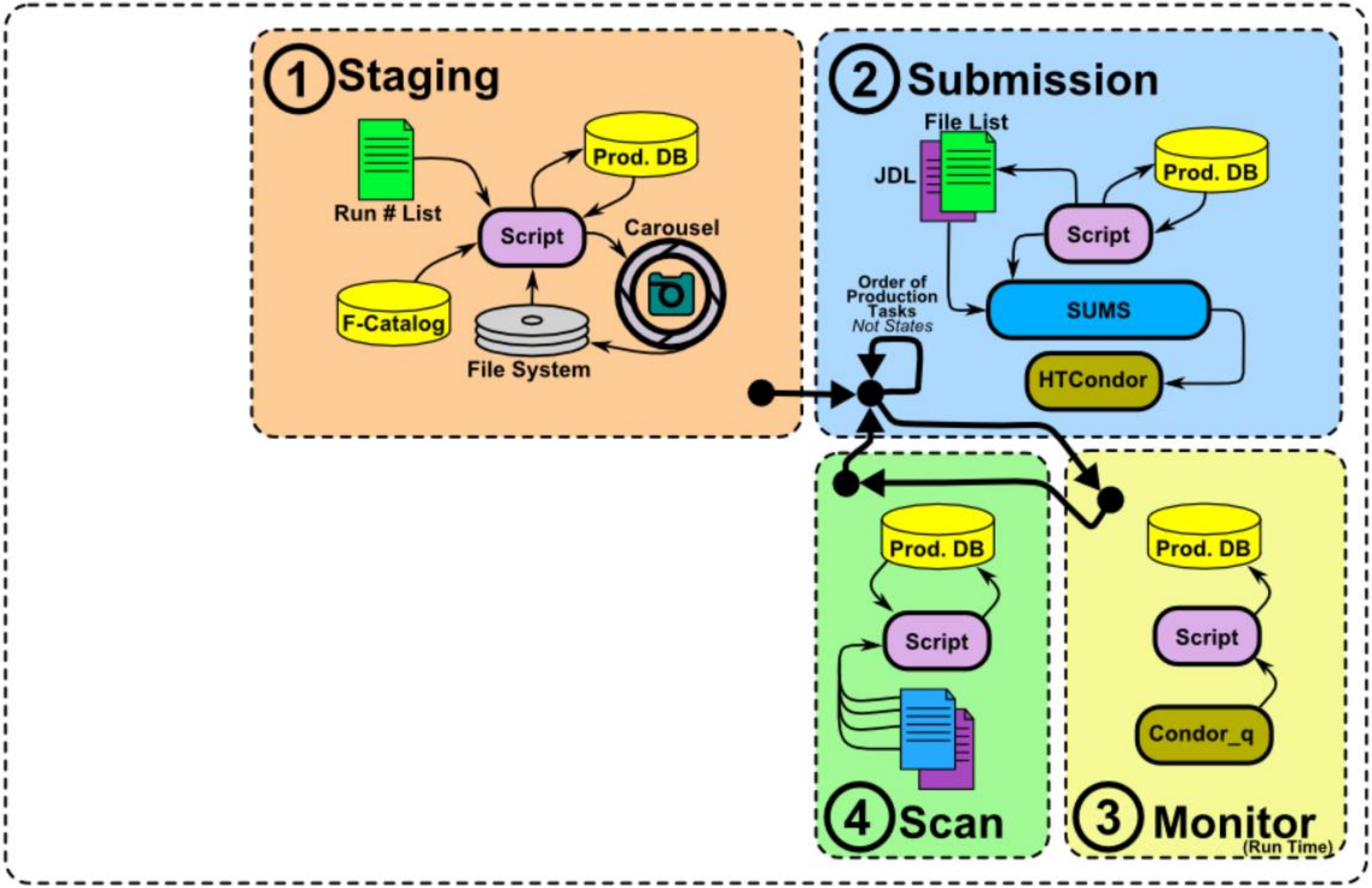
# Detailed Stages of Production



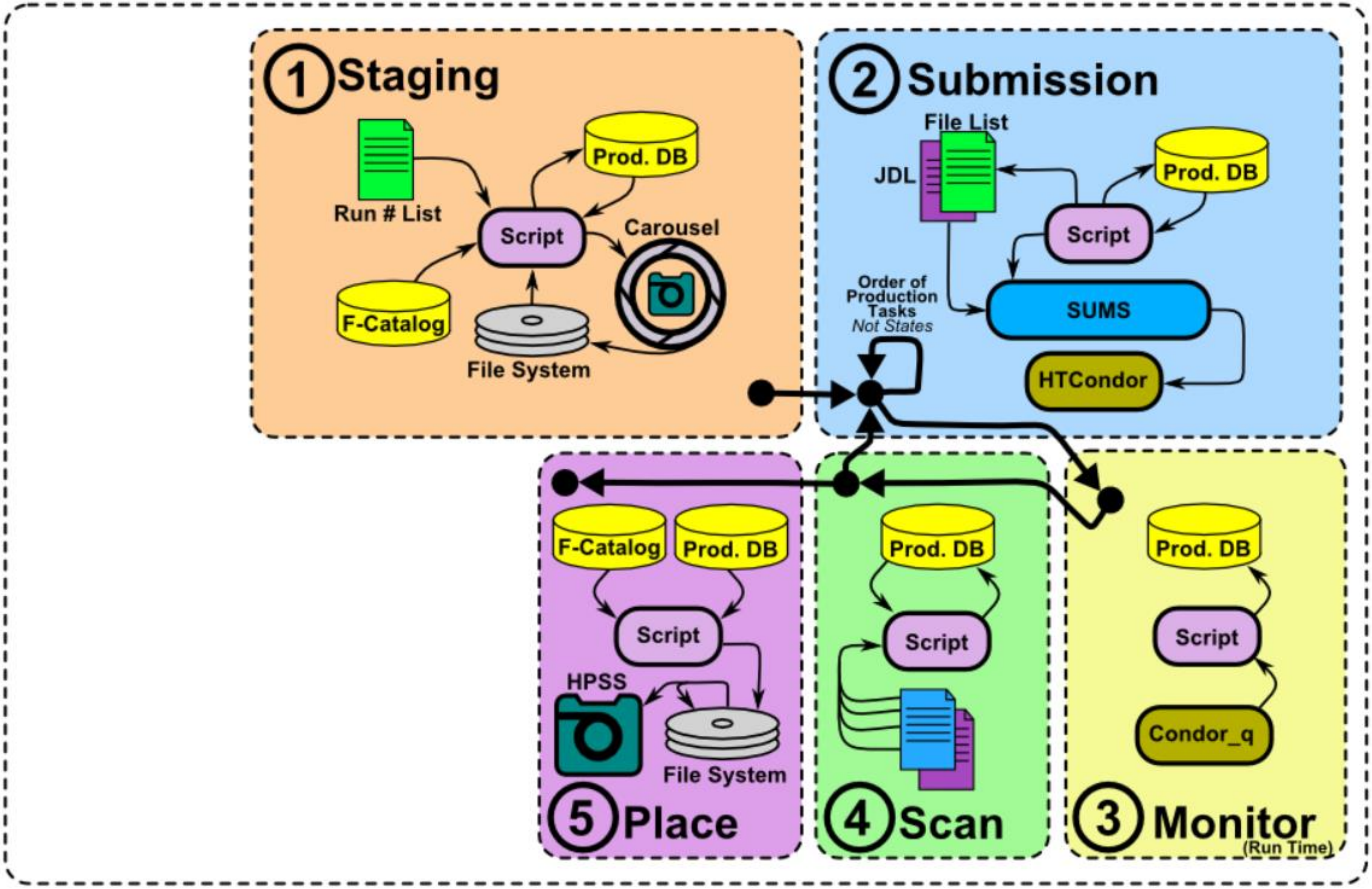
# Detailed Stages of Production



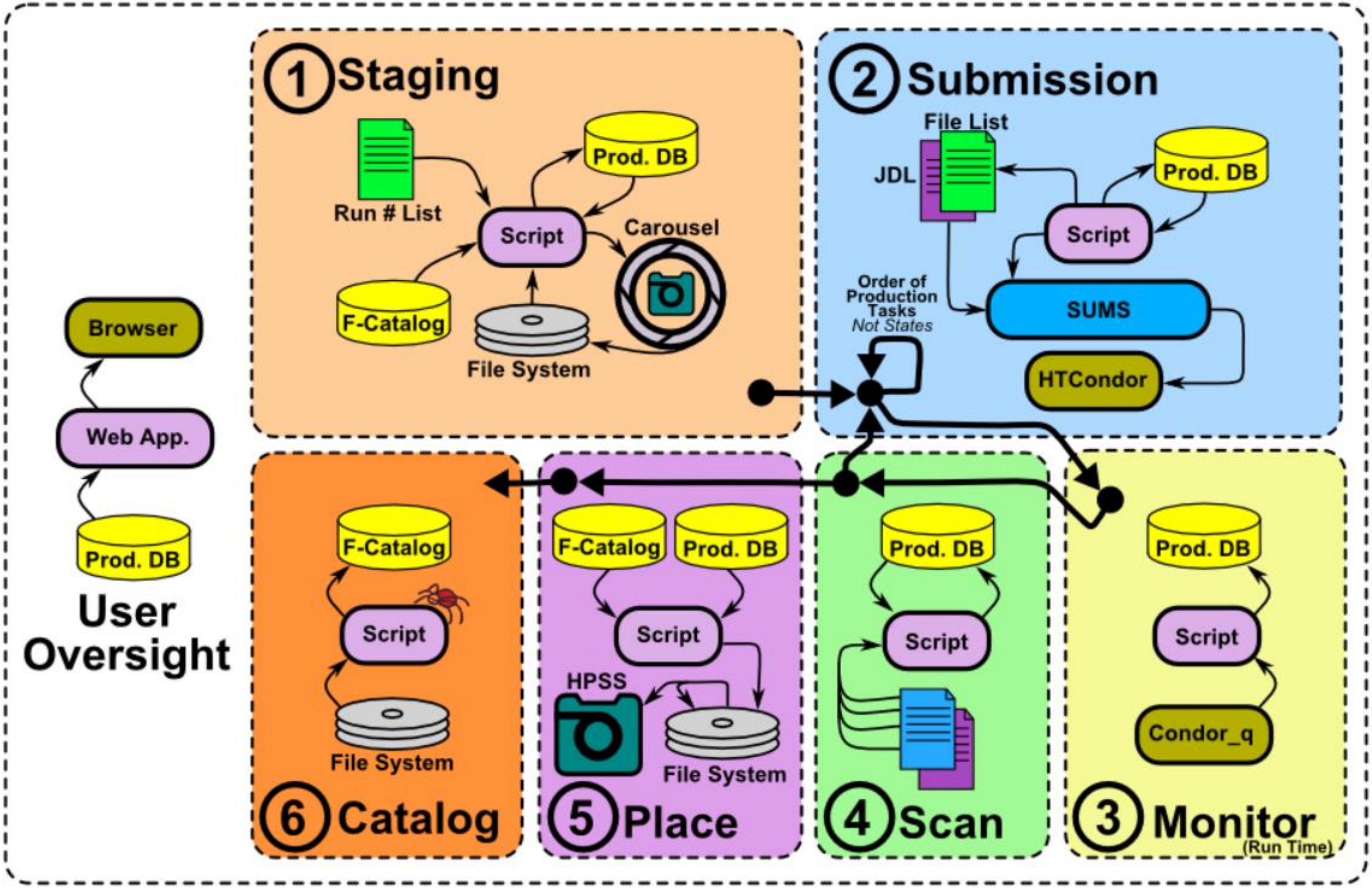
# Detailed Stages of Production



# Detailed Stages of Production

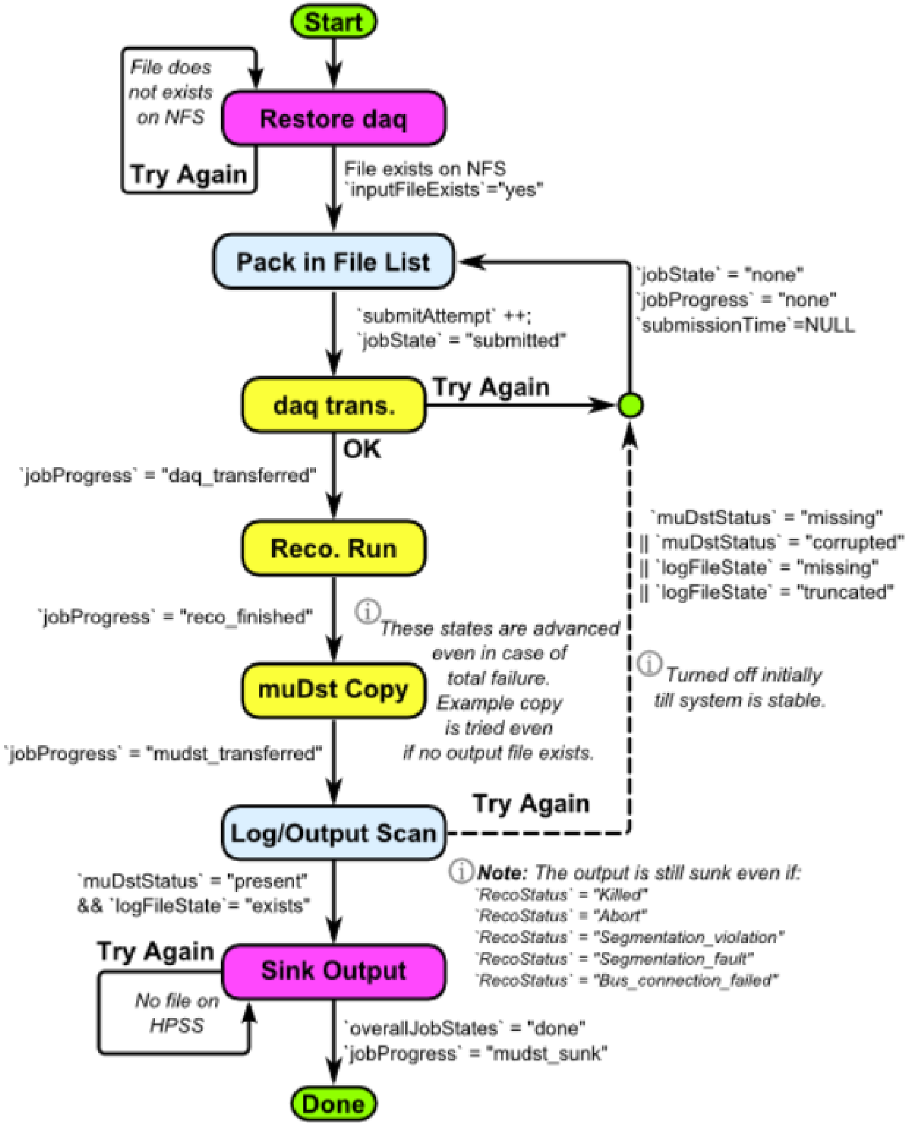


# Detailed Stages of Production



# Grid Production Framework State Diagram

- Finite state checking exists to verify each stage of the production
- Central DB at BNL holds each job's state
- Each job is associated with:
  - One Input file
  - Batch System ID
  - Output file(s)
    - Event processing log
    - Batch System log
- System gathers information from:
  - File sizes are checked after each transfer
  - Batch system is polled every hour to get the current state of each job
  - Jobs send messages at each stage:
    - job start up (copy input starts), input transfer done (reconstruction starts), reconstruction done (output transfer starts), output transfer done
  - Log files (batch and reconstruction) are scanned for error states



Start

File does not exists on NFS

Try Again

Restore daq

File exists on NFS
`inputFileExists`="yes"

Pack in File List

`jobState` = "none"
`jobProgress` = "none"
`submissionTime`=NULL

`submitAttempt` ++;
`jobState` = "submitted"

daq trans.

Try Again

OK

`jobProgress` = "daq_transferred"

Reco. Run

`muDstStatus` = "missing"
|| `muDstStatus` = "corrupted"
|| `logFileState` = "missing"
|| `logFileState` = "truncated"

`jobProgress` = "reco_finished"

ⓘ These states are advanced even in case of total failure. Example copy is tried even if no output file exists.

ⓘ Turned off initially till system is stable.

muDst Copy

`jobProgress` = "mudst_transferred"

Try Again

Log/Output Scan

ⓘ **Note:** The output is still sunk even if:
`RecoStatus` = "Killed"
`RecoStatus` = "Abort"
`RecoStatus` = "Segmentation_violation"
`RecoStatus` = "Segmentation_fault"
`RecoStatus` = "Bus_connection_failed"

`muDstStatus` = "present"
&& `logFileState`= "exists"

Try Again

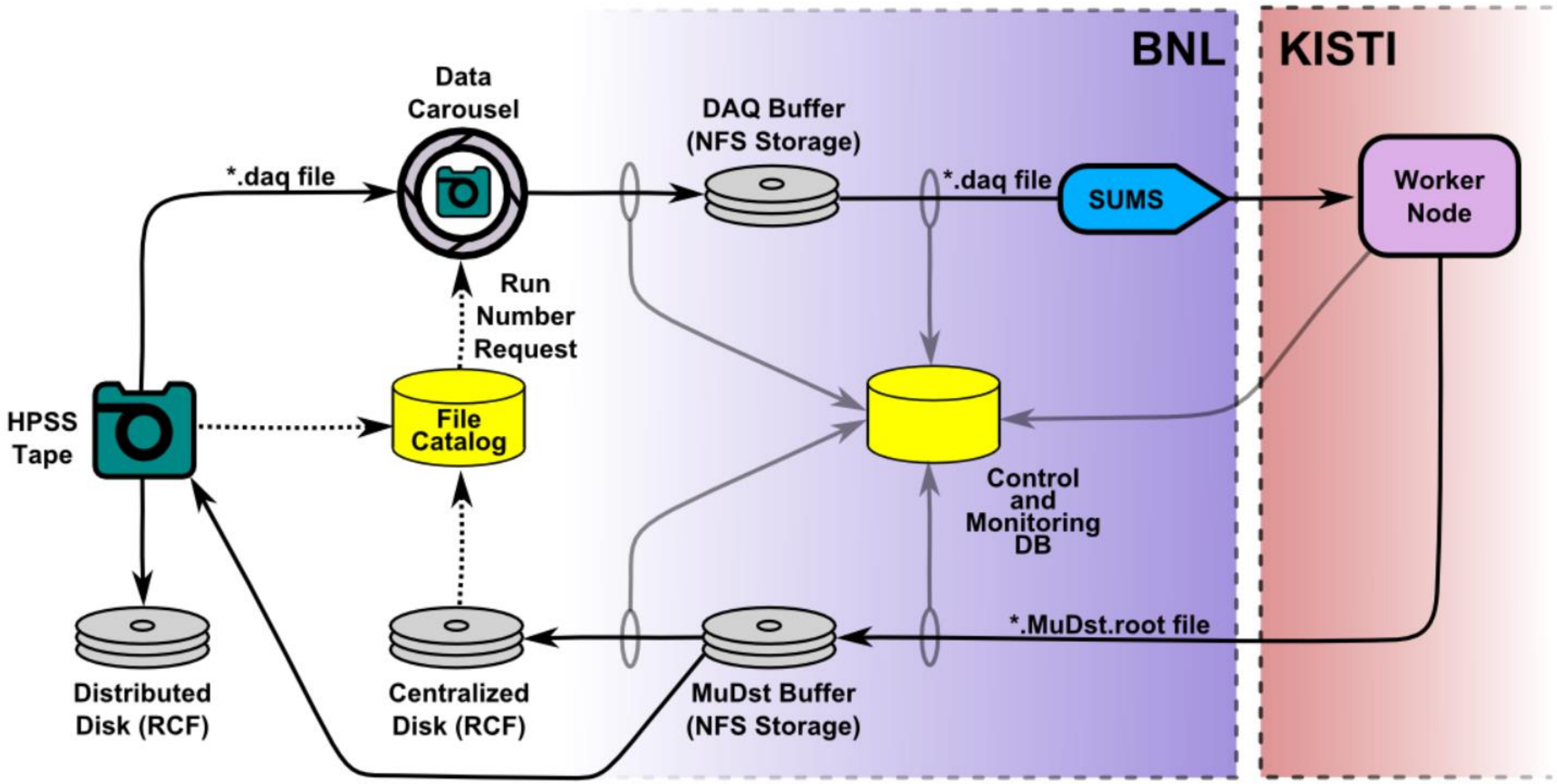
Sink Output

No file on HPSS

`overallJobStates` = "done"
`jobProgress` = "mudst_sunk"

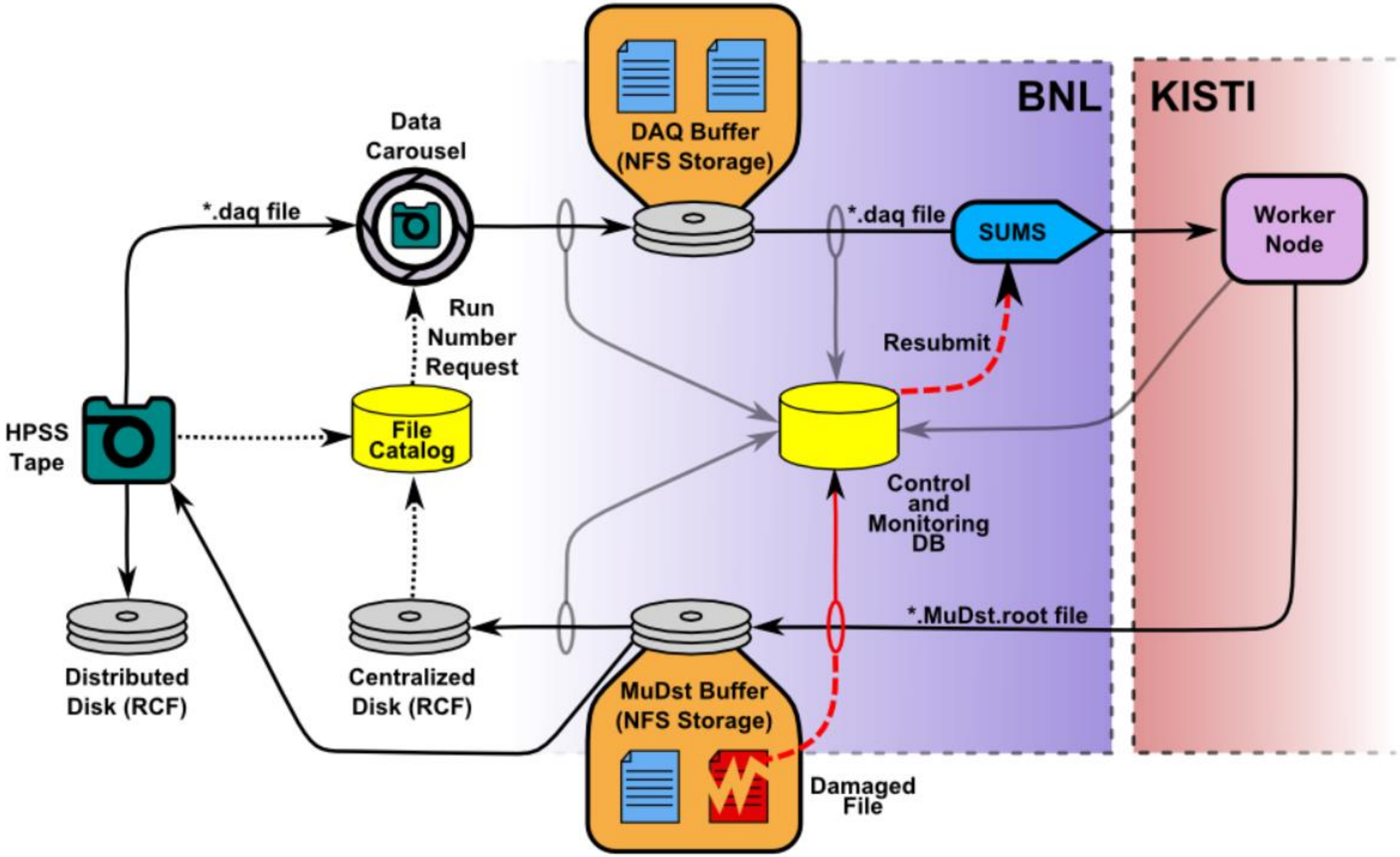
Done

# Production Framework Data Flow



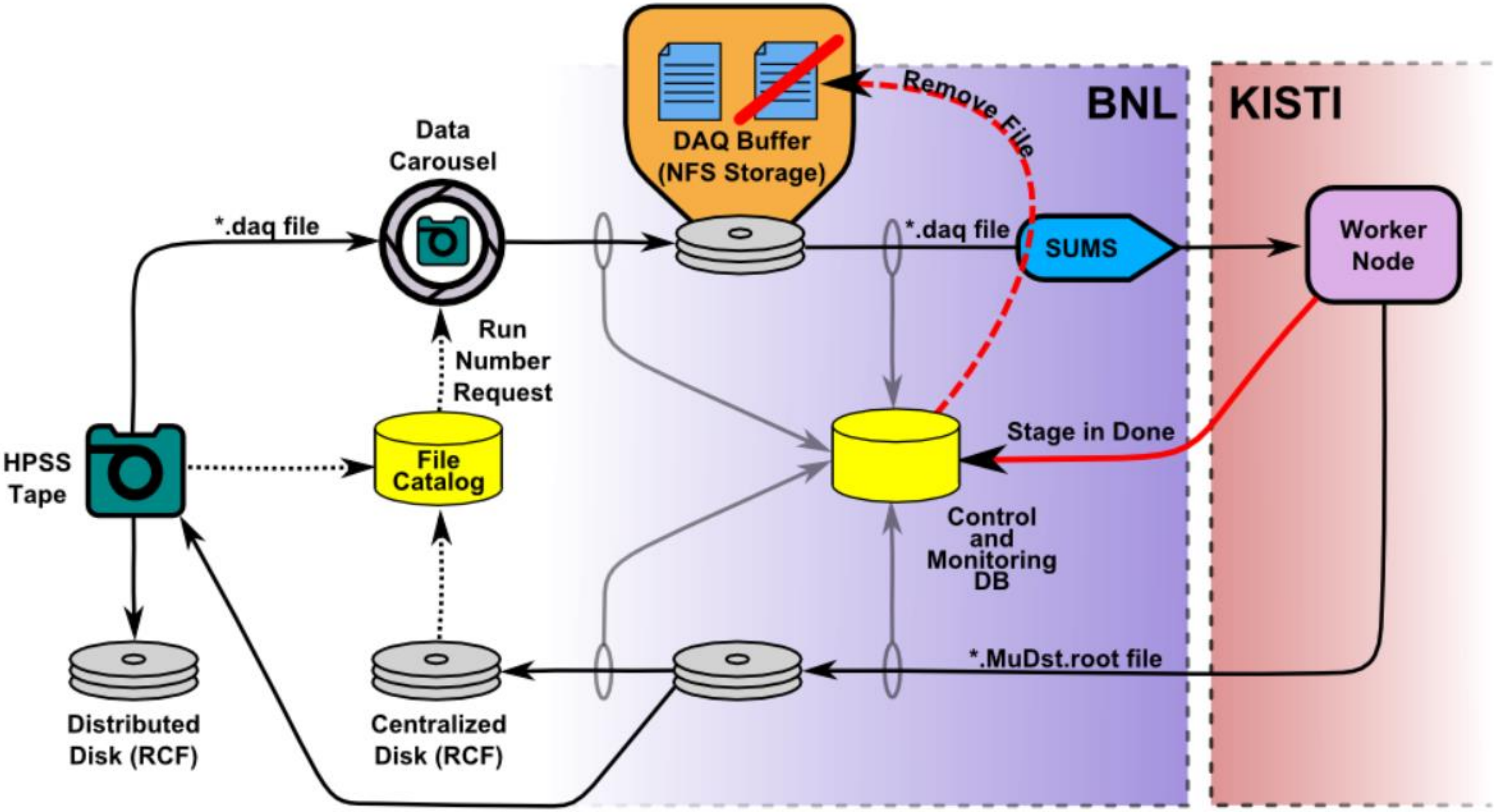
Note: *.Daq files are STAR's raw input files, and *.MuDst files are the reconstructed output

# Production Framework Data Flow
# Two Models of Operation (*1*)



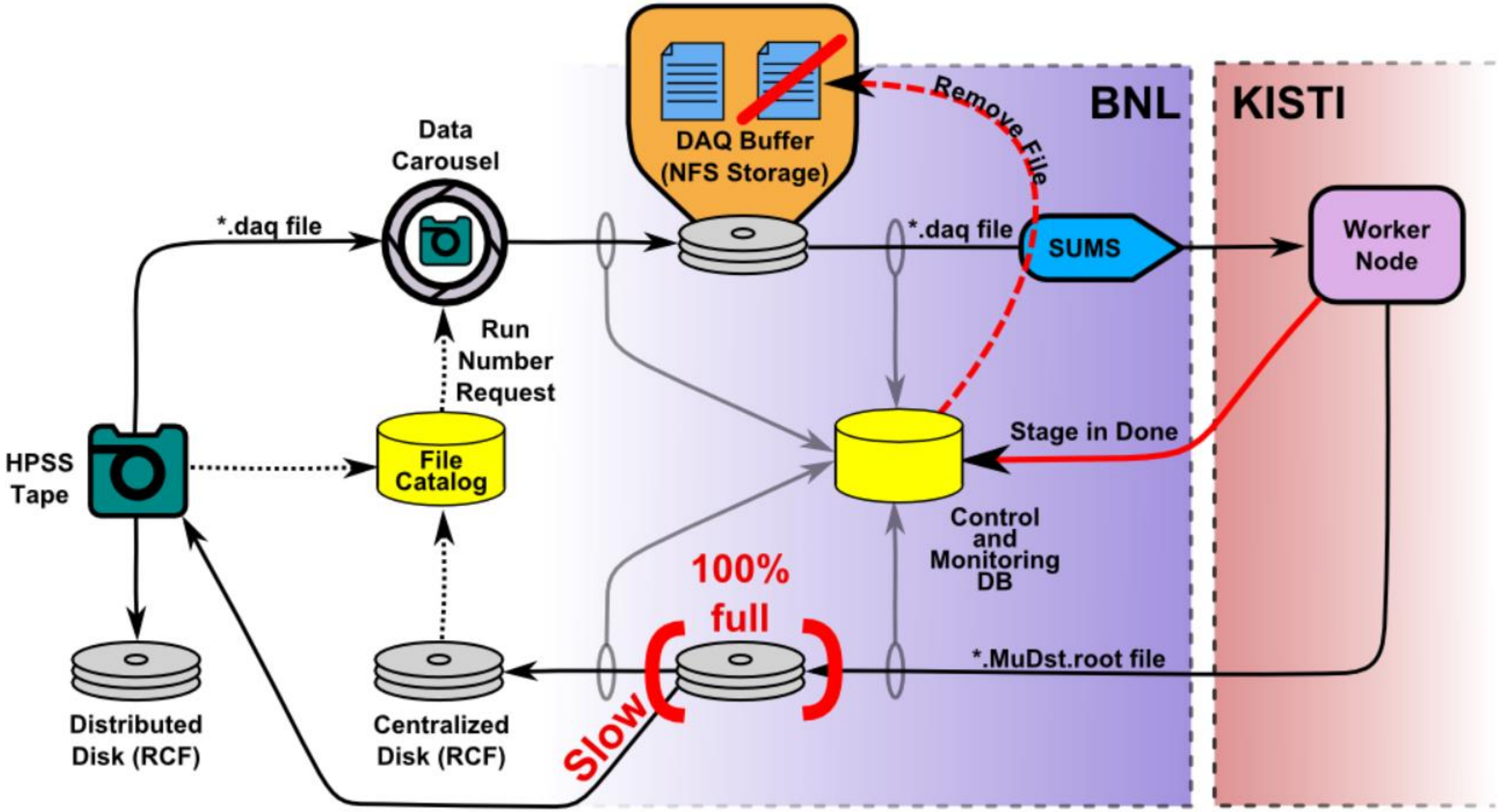
Mode 1 - The buffer holds the input-file until a valid output file is returned. Running jobs limited by buffer size. However resubmission only requires submitting another job.

# Production Framework Data Flow
# Two Models of Operation (2)



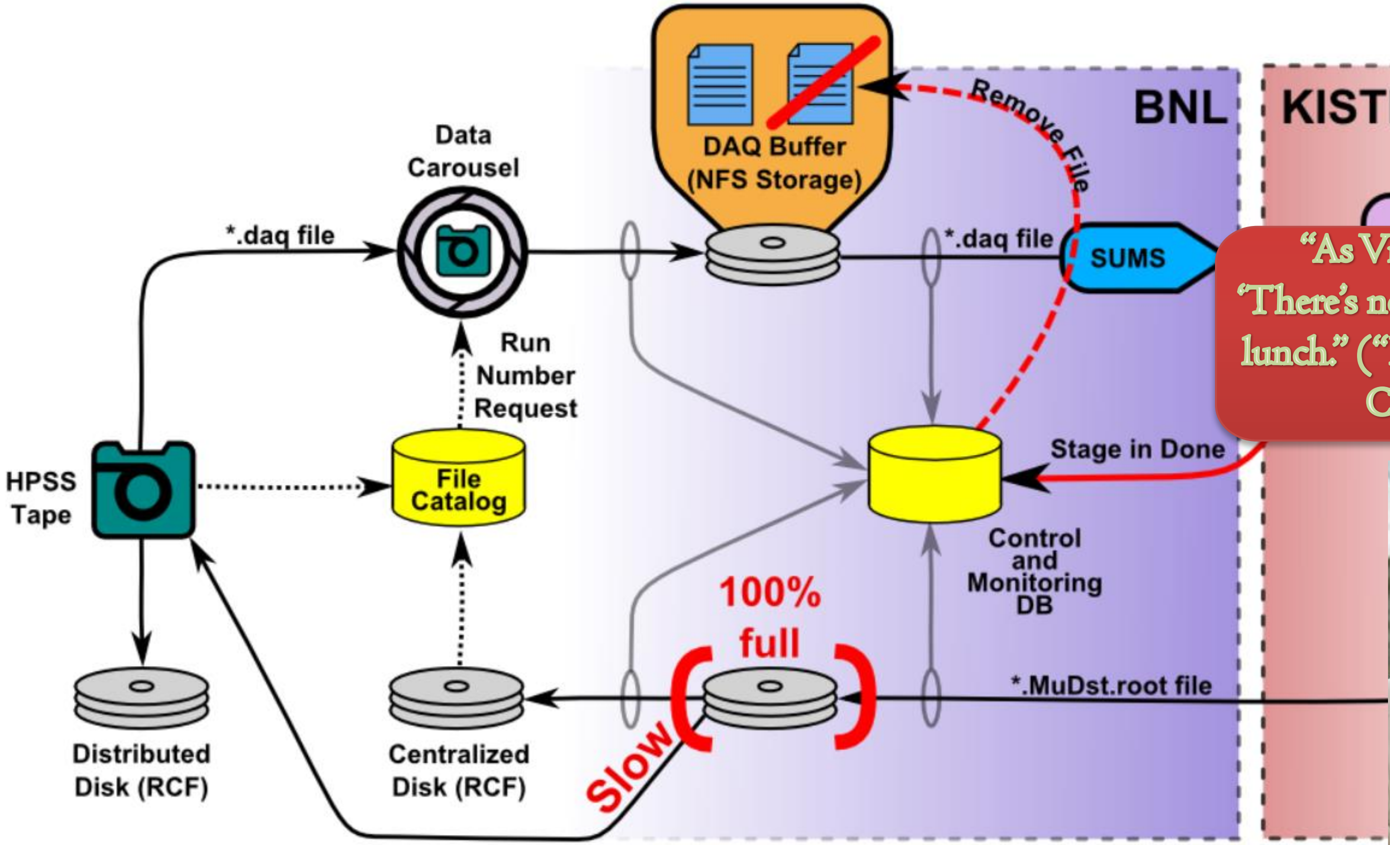
Mode 2 – As soon as job stages in input-file, input buffer is cleared. More running jobs. Idle jobs limited by input buffer size. Resubmission requires restoration of input from tap. Prone to output buffer overruns if HPSS slows down or stops.

# Production Framework Data Flow
# Two Models of Operation (2)



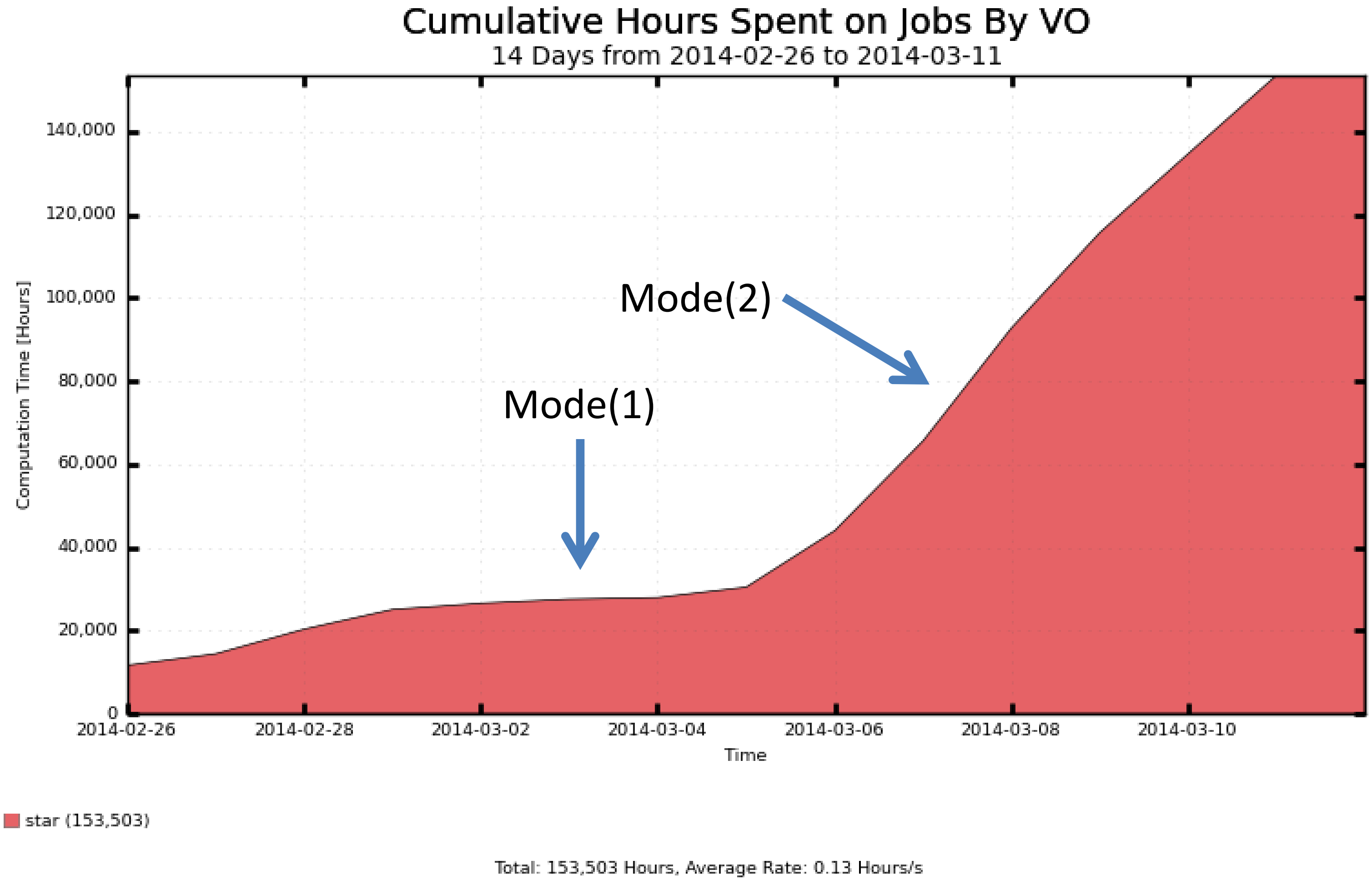
Mode 2 – As soon as job stages in input-file, input buffer is cleared. More running jobs. Idle jobs limited by input buffer size. Resubmission requires restoration of input from tap. Prone to output buffer overruns if HPSS slows down or stops.

# Production Framework Data Flow
# Two Models of Operation (2)



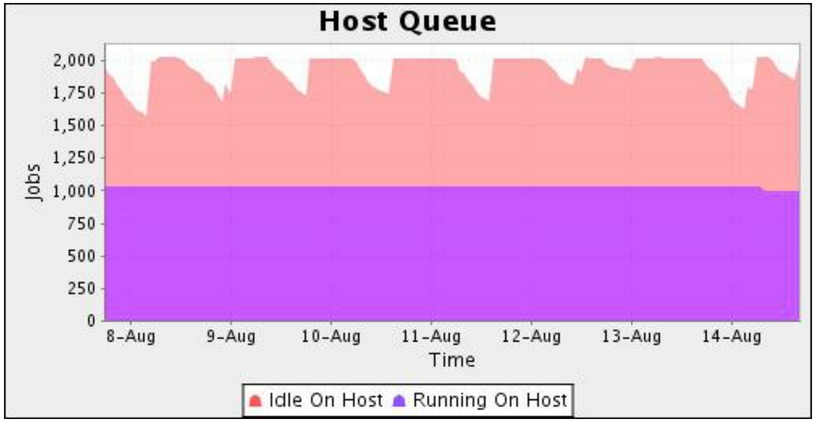
"As Virgil pointed out, 'There's no such thing as a free lunch." ("Inspector Morse" by Colin Dexter)

Mode 2 – As soon as job stages in input-file, input buffer is cleared. More ru s limited by input buffer size. Resubmission requires restoration of input from output buffer overruns if HPSS slows down or stops.

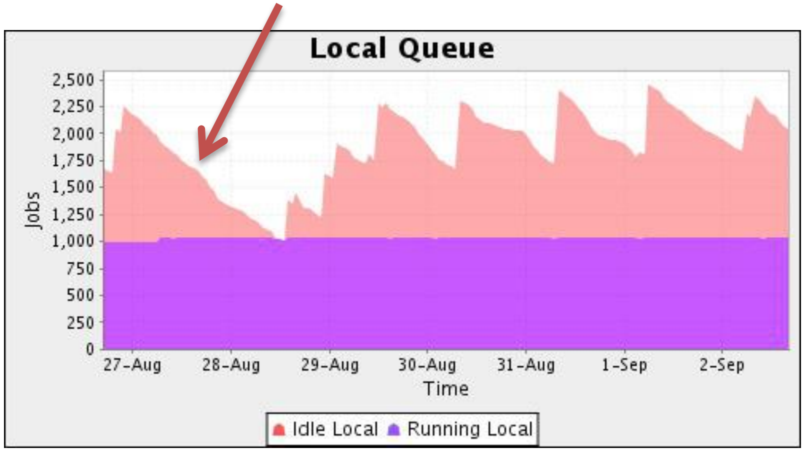

Cumulative Hours Spent on Jobs By VO
14 Days from 2014-02-26 to 2014-03-11
Computation Time [Hours]
0
20,000
40,000
60,000
80,000
100,000
120,000
140,000
2014-02-26
2014-02-28
2014-03-02
2014-03-04
2014-03-06
2014-03-08
2014-03-10
Time
Mode(1)
Mode(2)
🟥 star (153,503)
Total: 153,503 Hours, Average Rate: 0.13 Hours/s

# Running

### Normal Running

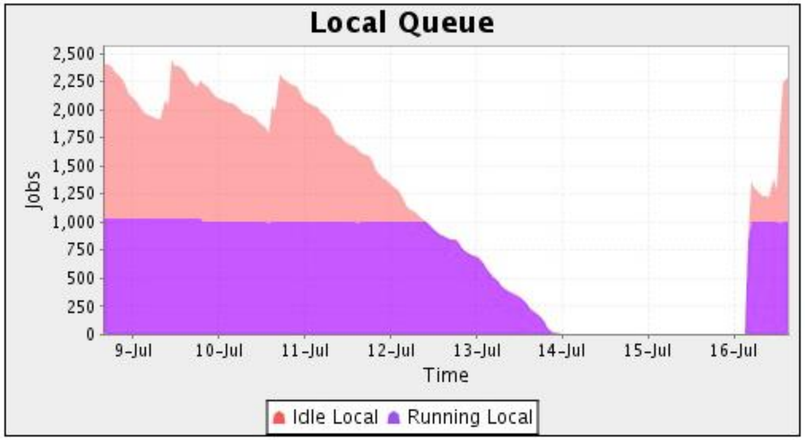


### 35 Hours HPSS Down Time



- Monitoring was setup to easily track queue occupancy
- Statistics indicate good slot utilization
- Able to survive outages in network and HPSS
- Filling the queue is fast

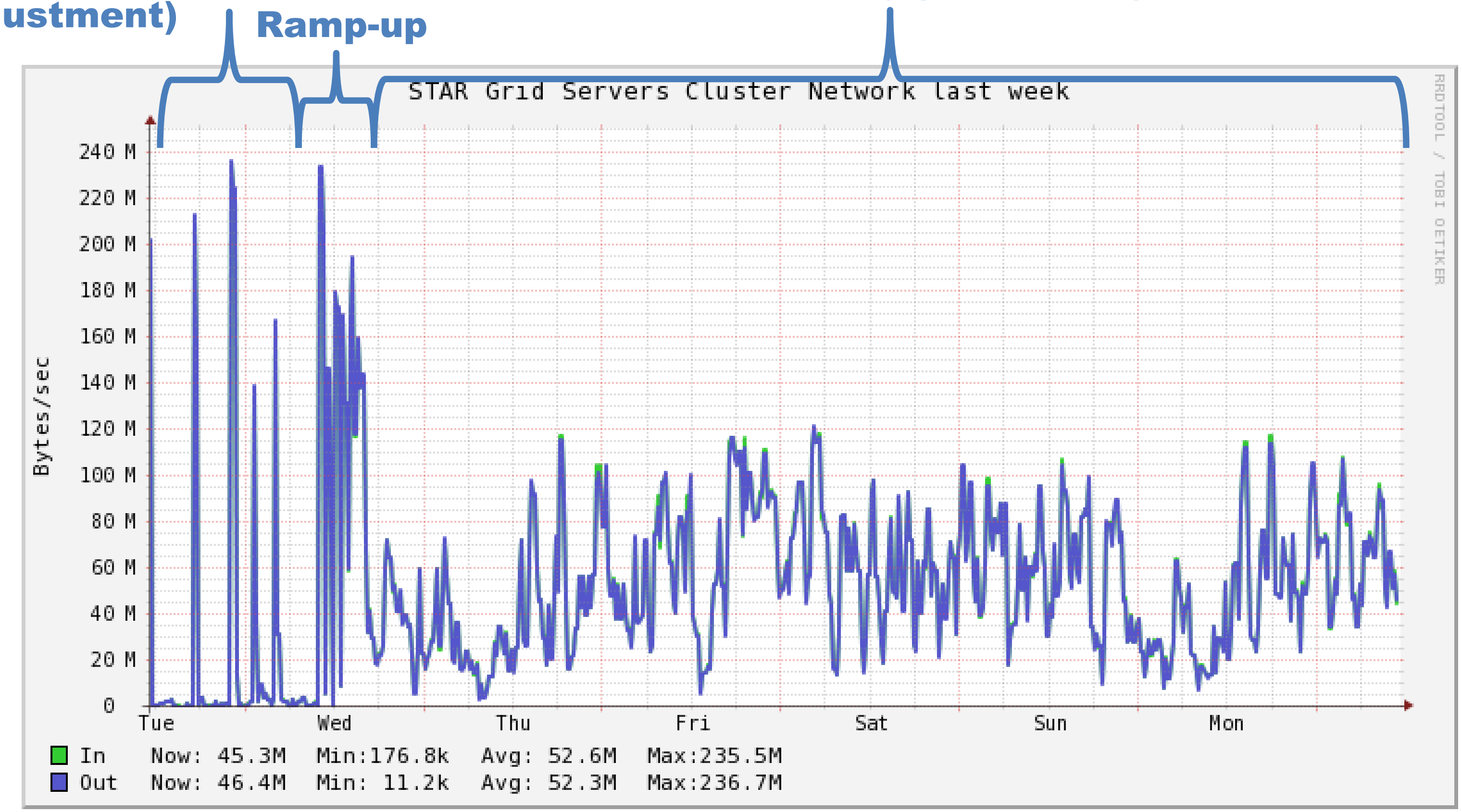
### Restart after Queue Drainage



*(Deliberate Drainage for GK for Patching)*

# Network Bandwidth After Saturation

- Once batch system is saturated, job startup (I/O) is more diffuse.

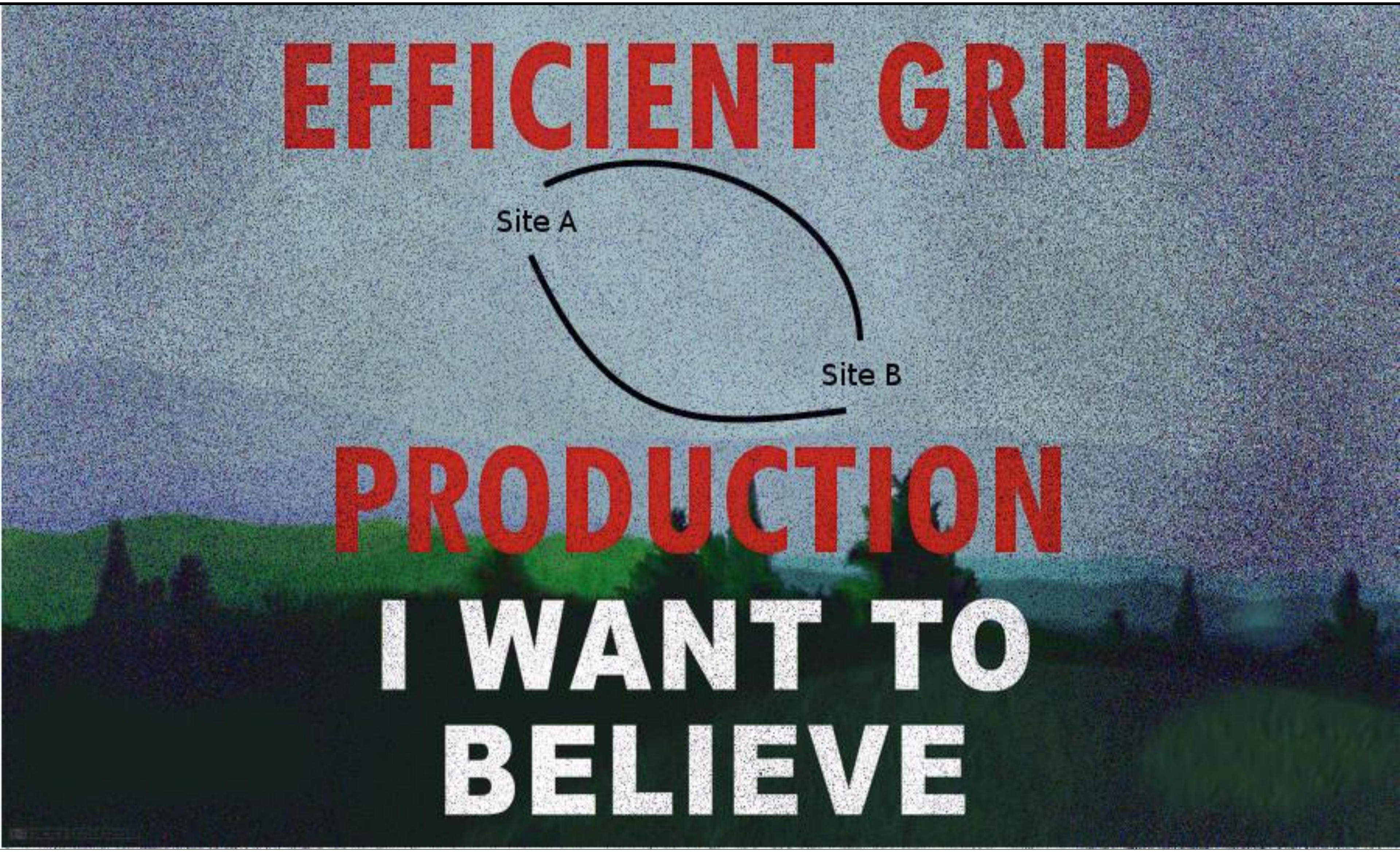


300Jobs(experimentation and adjustment)

Ramp-up

1,000 Jobs Running (Saturation)

# Statistics And Outcomes



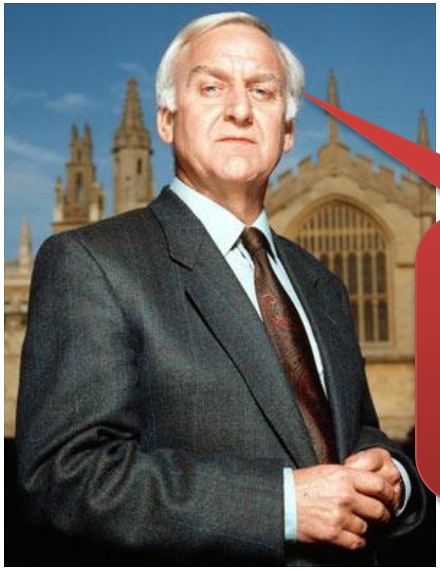
# Statistics And Outcomes

- Dataset: p+p200 GeV consisting of 105,632 files, 397.3 TB of input, 213.7 TB of output
- Production Time:  9 months, with 1025 slots, 6,179,544 wall-CPU-hours
- **99% of jobs returned output and log files to HPSS !!**
  - Contributing factors:
    - Finite State Checking
    - Dedicated resources
      - generous queue time limit to allow jobs to finish gracefully
    - High site availability and stability
    - Input/output Copy Hardening

# Statistics And Outcomes

- Dataset: p+p 200 GeV consisting of 105,632 files, 397.3 TB of input, 213.7 TB of output
- Production Time: 9 months, with 1025 slots, 6,179,544 wall-CPU-hours
- **99% of jobs returned output and log files to HPSS !!**
  - Contributing factors:
    - Finite state checking
    - Dedicated resources
      - generous queue time limit to allow jobs to finish gracefully
    - High site availability and stability
    - Input/output copy hardening

*Could we replicate this at other sites ?*

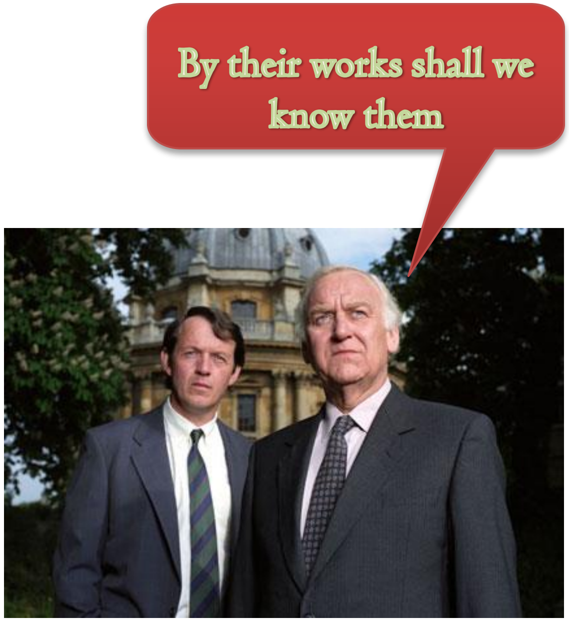


"I never think I only ever imagine." ("Inspector Morse" by Colin Dexter)

# Future Works

- Running on sites with shared resources
  - Target NERSC PDSF (800 slots)
    - Shared by users, Limited by CPU hours consumed
- Binary running:
  - STAR’s online farm(+200slots).
    - Shared by very high priority jobs requiring immediate termination of reconstruction jobs
    - Framework needs to detect and stop submitting
    - Recovering a massive number of jobs will be required
    - However large period of times where there will be NO activities (hence, all slots available)
- Distributing Workload Between Two or More Sites (central planning)
  - Developing algorithms for dynamic prediction of the optimal workload distribution ratio can be a vexing problem due to spontaneous farm load changes.
    - Initial research on this topic was carried out in 2005:
      - https://indico.cern.ch/event/0/session/7/material/paper/1?contribId=393
      - Efstathiadis E. (BNL), Lauret J. (BNL), Legrand I. (Caltech), Hajdu L. (BNL) for the STAR & US-CMS PPDG teams, “Development and use of MonALISA high level monitoring services for Meta-Schedulers”, CHEP04 Paper. September 2004.

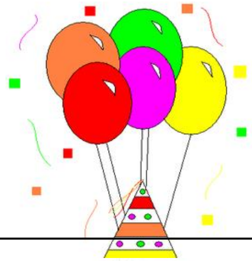
# Future Works



- Running on sites with shared resources
  - Target NERSC PDSF (800 slots)
    - Shared by users, limited by CPU hours consumed
- Binary running:
  - STAR's online farm (200+ slots)
    - Shared by very high priority jobs requiring immediate termination of reconstruction jobs
    - Framework needs to detect and stop submitting
    - Recovering a massive number of jobs will be required
    - However large period of times where there will be NO activities (hence, all slots available)
- Distributing workload between two or more sites (central planning)
  - Developing algorithms for dynamic prediction of the optimal workload distribution ratio can be a vexing problem due to spontaneous farm load changes
    - Initial research on this topic was carried out in 2005:
      - https://indico.cern.ch/event/0/session/7/material/paper/1?contribId=393
      - Efstathiadis E. (BNL), Lauret J. (BNL), Legrand I. (Caltech), Hajdu L. (BNL) for the STAR & US-CMS PPDG teams, "Development and use of MonALISA high level monitoring services for Meta-Schedulers", CHEP04 Paper. September 2004.

# Conclusion

- STAR has vast computing requirements and a history of offloading work to other sites whenever possible
- We presented a Finite State workflow leveraging STAR's KISTI Korean Tier-1 site resources
  - Automated yet simple set of components were used
  - We leveraged past knowledge and took great care in developing this framework
- We achieved an unprecedented job stability – Average efficiency 99% over 9 months is a great success
- The effort shows that job efficiency running on the grid can be as good as local production
- We aim to implement balancing workload between more then one site in our next development, targeting shared resources and infrastructure



# Questions?

Apparently it happened during a colloquium.

What's that when it's at home then?

It's a posh name for a conference.

Inspector Lewis "Wild Justice"