

Automated Validation System for CEPC

Teng Li

Shandong University

2021.4

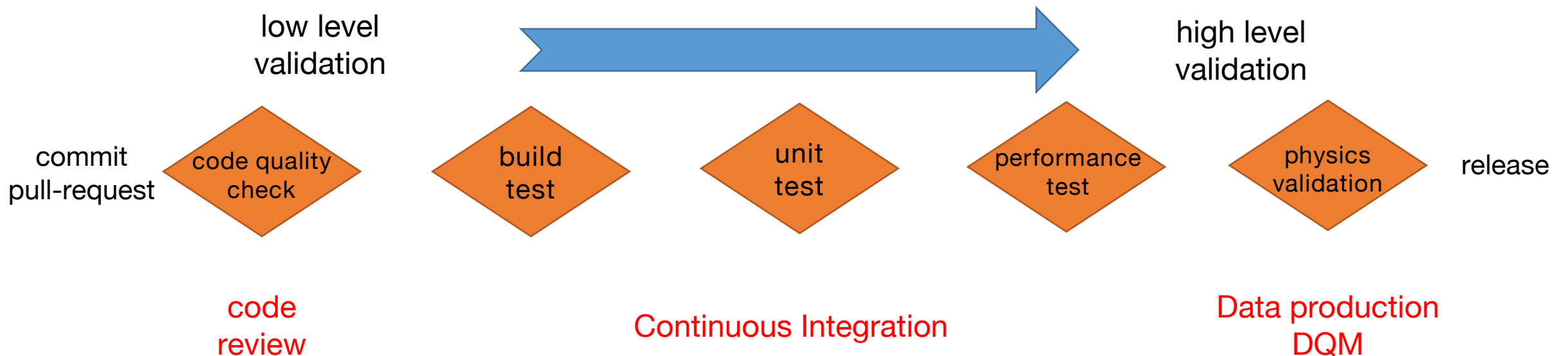
Outline

- HEP software validation toolkit
- Modern CI/CD system
- Proposed validation system for CEPC

Software Validation

Software validation

- Software validation is critical for HEP experiments
 - Long lifecycle, complex, radical changing software
 - Local issues timely and automatically
- HEP software validation is sophisticated



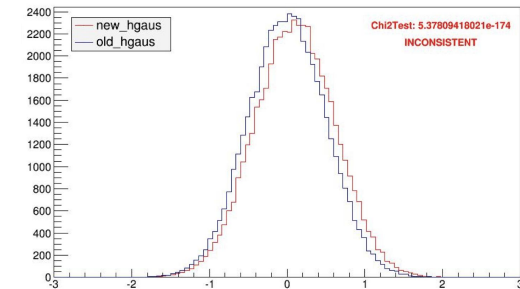
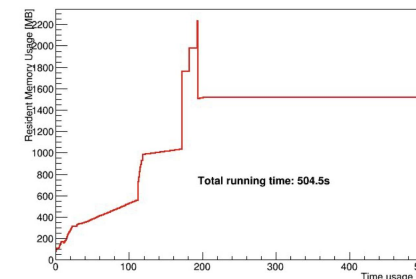
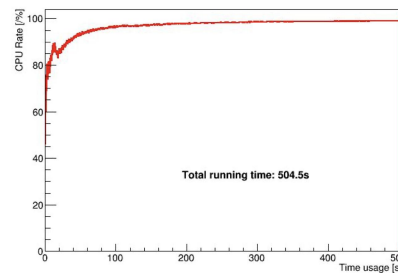
Development of Validation Toolkit

- Large experiments usually develop their own validation toolkits
 - ATLAS (ATN, RTT, FCT); CMS; LHCb (SimDQ)
 - Similar functionalities, different focus
- A powerful toolkit was developed for JUNO for software validation and data production
 - Provide interfaces to define and run unit tests:

```
1  #!/usr/bin/python
2
3  from JunoTest import *
4
5  mytest = UnitTest()
6
7  mytest.addCase("detsim", "python tut_detsim.py --extmax 300 gun")
8  mytest.addCase("histtest", "root -q draw.C", plotRef="oldPlot/histos.root")
9
10 mytest.run()
```

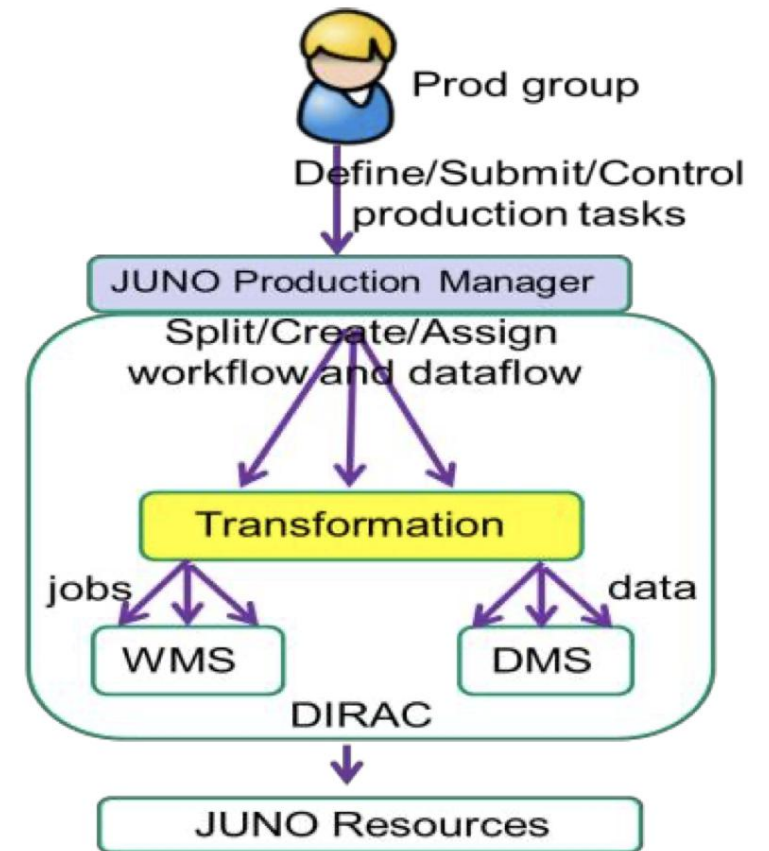
```
1  $ ./junotest UnitTest all
2  $ ./junotest UnitTest help
3  $ ./junotest UnitTest {package name}
```

- Various detectable failures
- Performance profiling
- Validation on physics distributions



Development of Validation Toolkit

- Physics validation based on massive data production
 - Flexible production definition based on steering files (.ini)
 - Chained production workflow
 - Automatic job generation, monitoring and physics validation
 - Integrated with the distributed computing system (DIRAC transformation system)



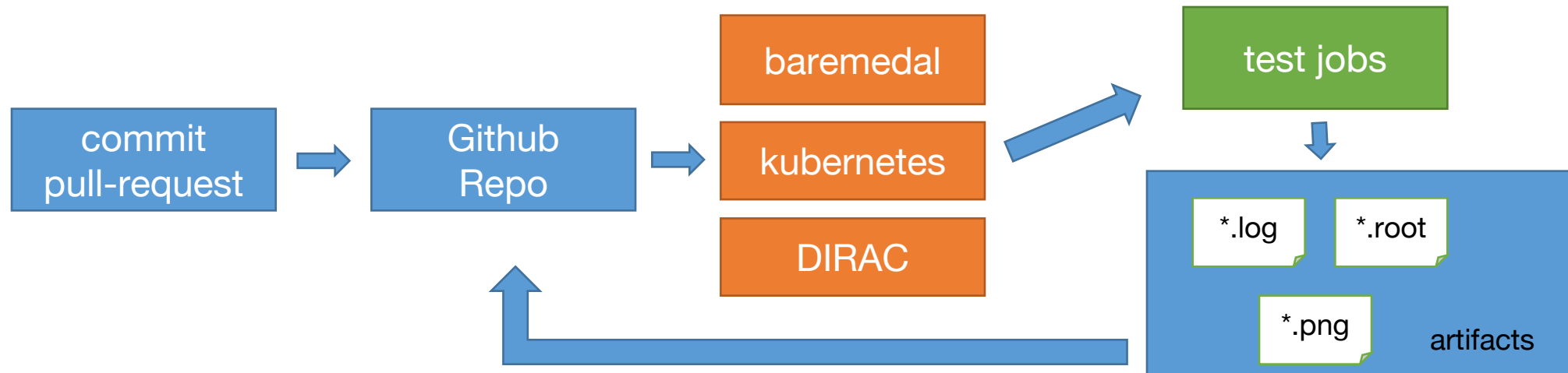
Future Development Plan

- Unit test and performance test: CMake and Catch2
 - Catch2 is a modern unit testing framework
 - Support various features to build flexible unit tests, performance benchmarking etc.
 - Could be easily integrated with CI system
- Data production and physics validation
 - Design workflow definition tool based on yaml files
 - Better integration with DIRAC transformation system
 - Intergrate with modern CI system

Automated Continuous Integration

Github Action System

- Github Action is a modern CI (Continuous Integration)/ CD (Continuous Deployment) system released in 2019
 - Continuously build and test triggered by commit/pull request
 - Defined in yaml files as workflow:
checkout --> build --> unit test --> performance test
 - Test jobs are executed on Github, or self-hosted runner machines



An example: LHCb validation system

- LHCb software validation consists of:
 - A nightly test system
 - Test whether the code could be built on different platforms
 - Standard/Customized Gitlab CI tests
 - Check for code formatting and copyright notices
 - Unit tests for all core software packages
 - Analysis jobs
- See slides from Chris Burr
 - <https://indico.ihep.ac.cn/event/11444/session/12/contribution/173/material/slides/0.pdf>

LHCb nightly test

- Run nightly builds and tests for all Physics applications
 - Script based system built on CouchDB and CouchAPP
 - Test large matrix of projects, platforms and branches
 - Thousands of jobs are ran each day

lhc-head - build: 2750 (2020-10-26)

head of everything against Gaudi/master and LCG_97a
available on: cvmfs

Compare with previous build Compare with other slot Rebuild Deploy on CVMFS Browse files

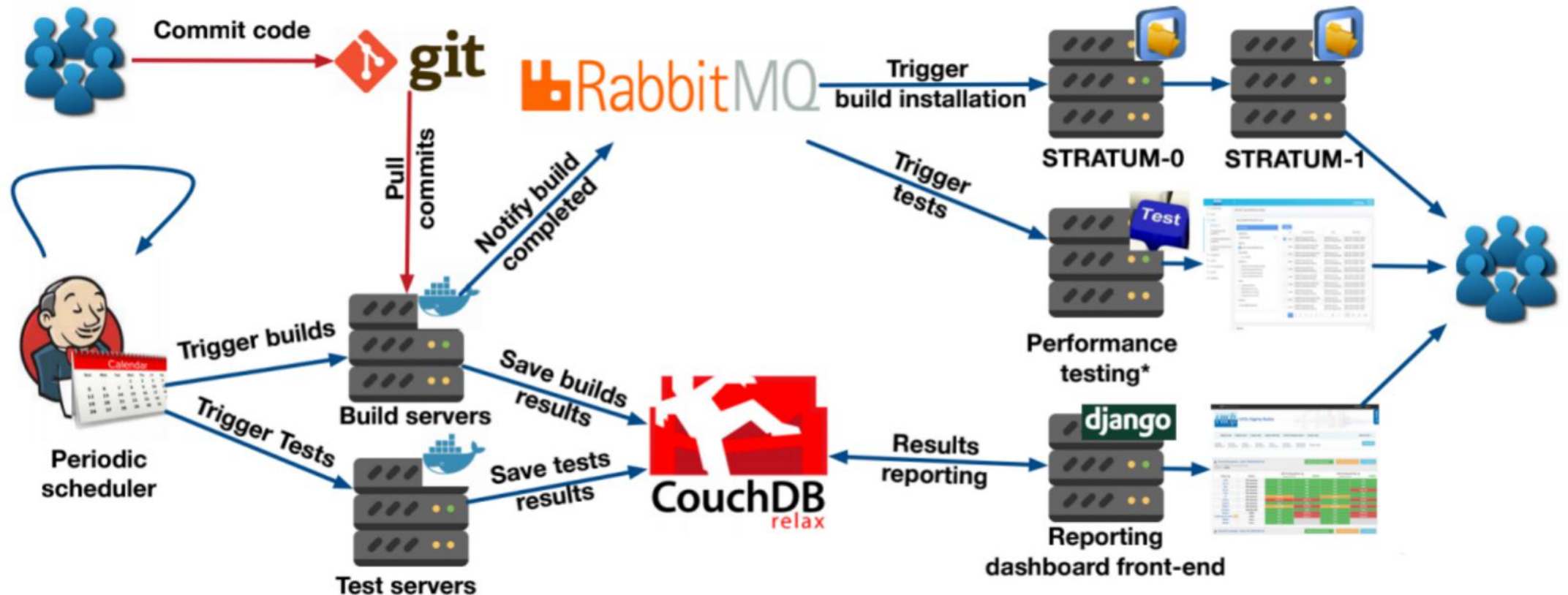
Project	Version	x86_64-centos7-gcc9-opt	x86_64-centos7-gcc9-dbg	x86_64-centos7-clang8-opt	x86_64-centos7-clang8-dbg	x86_64-centos7-gcc9-do0	x86_64-avx2+fm-centos7-gcc9-opt	skylake_avx512+vecwid256-centos7-gcc9-opt	x86_64-centos7-gcc9-py3-opt	x86_64-centos7-gcc9-py3-dbg
PARAM	None	build	build	build	build	build	build	build	build	build
DBASE	None	build	build	build	build	build	build	build	build	build
LCG	97a									
Gaudi	master	build (4)	tests	build (4)	tests	build	tests	build (4)	tests	build (4)
Online	HEAD	build	tests	build	tests	build	tests (1)	build	tests (1)	build
Detector	v0-patches	build (1)	tests (0)	build (1)	tests (0)	build (1)	tests (0)	build (1)	tests (0)	build (1)
LHCb	HEAD	build	tests	build	tests	build	tests (1)	build	tests	build
Lbcom	HEAD	build	tests	build	tests	build	tests	build	tests	build
Boole	PR	HEAD	tests (10)	build	tests (10)	build	tests (10)	build	tests (5)	build
Rec	HEAD	build	tests	build	tests	build	tests (1)	build	tests	build
Brunel	PR	HEAD	tests (8)	build	tests (8)	build	tests (8)	build	tests (8)	build
Phys	HEAD	build	tests	build	tests	build	tests	build	tests	build
Allen	HEAD	build	tests	build	tests	build	tests	build	tests	build
Moore	PR	HEAD	tests (14)	build	tests (14)	build	tests (14)	build	tests (14)	build
Analysis	HEAD	build	tests	build	tests	build	tests	build	tests	build
Davinci	HEAD	build	tests	build	tests	build	tests (2)	build	tests (1)	build
Panoramix	HEAD	build	tests	build	tests	build	tests	build	tests	build
Bender	HEAD	build	tests (71)	build	tests (72)	build	tests (71)	build	tests (71)	build
MooreOnline	HEAD	build (4)	tests (1)	build (4)	tests (1)	build (8)	tests (1)	build (4)	tests (1)	build (4)
Panoptes	HEAD	build (5)	tests	build (5)	tests	build (5)	tests	build (5)	tests	build (5)
Alignment	HEAD	build	tests (1)	build	tests (4)	build (4)	tests (1)	build	tests (1)	build
Geant4	HEAD	build	tests (0)	build	tests (0)	build (8)	tests (0)	build	tests (0)	build
Gauss	HEAD	build (1)	tests (66)	build (4)	tests (65)	build (8)	tests	build (47)	tests (54)	build (48)
Urania	HEAD	build	tests	build (1)	tests	build (2)	tests	build (1)	tests	build
Castelao	HEAD	build	tests	build	tests	build (37)	tests	build (37)	tests	build
Noether	HEAD	build (2)	tests (5)	build (2)	tests (5)	build (4)	tests (5)	build (2)	tests (5)	build (2)
Kepler	HEAD	build	tests (2)	build	tests (2)	build (82)	tests (2)	build	tests (2)	build
AlignmentOnline	HEAD	build	tests (1)	build	tests (1)	build	tests (1)	build	tests (1)	build
Lovell	HEAD	build (6)	tests	build (6)	tests	build (6)	tests	build (6)	tests	build (6)
Orwell	HEAD	build (15)	tests	build (15)	tests	build (15)	tests	build (15)	tests	build (15)
LHCIntegrationTests	HEAD	build	tests	build	tests	build	tests	build	tests	build
MooreAnalysis	HEAD	build	tests	build	tests	build	tests (5)	build	tests (3)	build

List of slots

lhc-2016-patches / 1447	lhc-2017-patches / 1065	lhc-2018-patches / 924
lhc-coverity / 530	lhc-dd4hep / 671	lhc-digi14-patches / 692
lhc-g4-dev / 629	lhc-gaudi-head / 2764	lhc-gauss-conf / 4
lhc-gauss-dev / 2555	lhc-gauss-fast / 113	lhc-gauss-gen-dev / 539
lhc-gauss-gen2-dev / 111	lhc-gauss-lamarr / 168	lhc-gaussino / 826
lhc-head / 2750	lhc-head-2 / 48	lhc-hit2011-patches / 622
lhc-hit2012-patches / 626	lhc-hit2016-patches / 444	lhc-log-dev3 / 1431
lhc-lcg-dev4 / 1445	lhc-master / 1235	lhc-master-mr / 1477
lhc-master-ref / 972	lhc-new-cmake / 207	lhc-prerelease / 2572
lhc-reco14-patches / 1089	lhc-reco15-patches / 573	lhc-run2-gaudi-head / 12
lhc-run2-patches / 849	lhc-run2-patches-dev4 / 492	lhc-run2-prerelease / 64
lhc-run3-cleanup / 51	lhc-sanitizers / 705	lhc-sim09 / 1424
lhc-sim09-cmake / 1153	lhc-sim09-upgrade / 1354	lhc-sim10 / 584
lhc-stripping21-firstpass-patch	lhc-stripping21-patches / 1029	lhc-stripping24-patches / 895
lhc-stripping24r2-28r2-patches / 193		

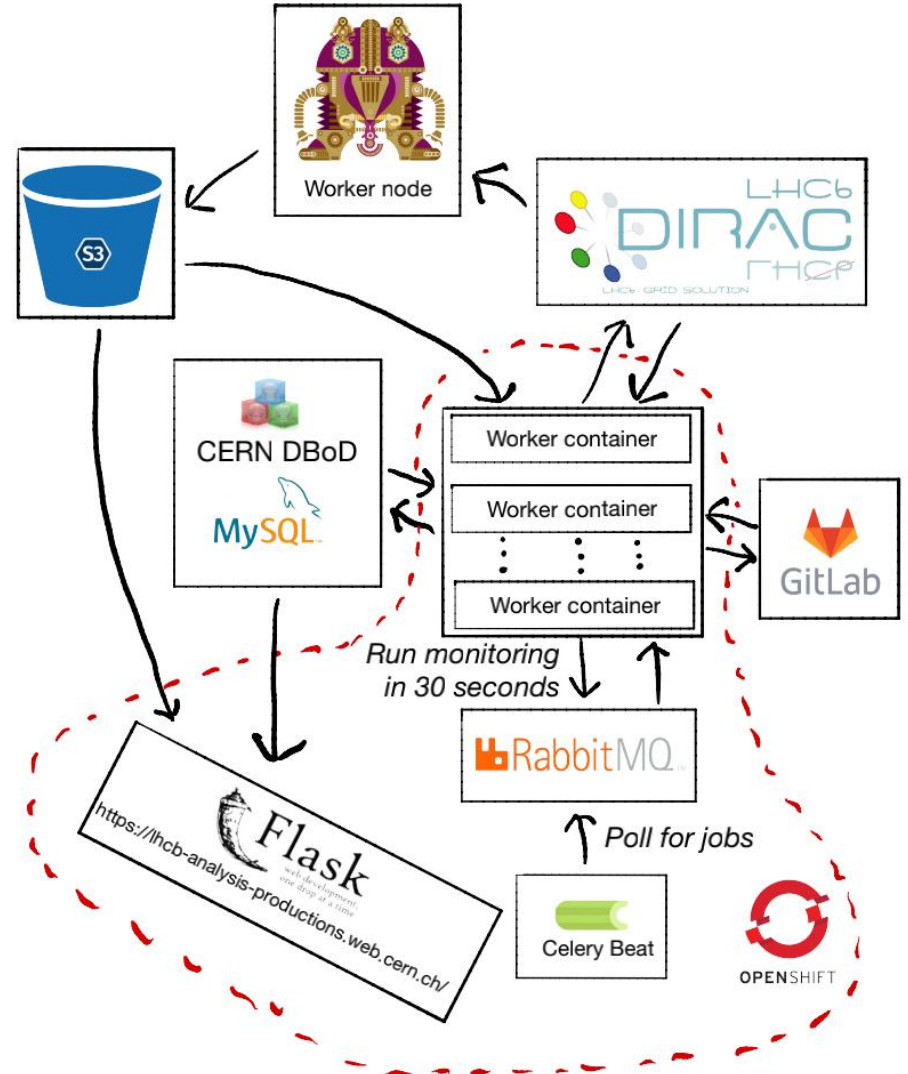
LHCb nightly test

- LHCb nightly test is moving a new solution



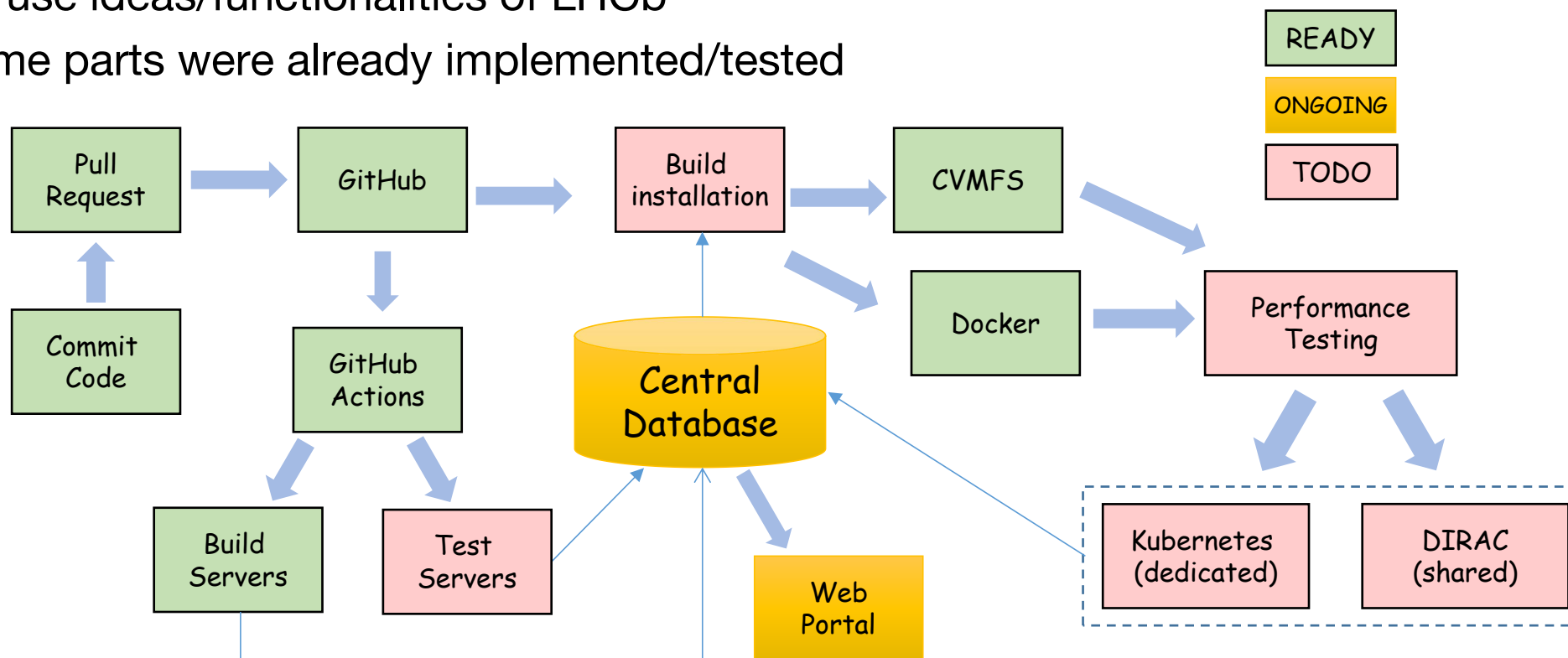
Customized CI for physics validation

- For 'Analysis Productions', use customized Gitlab CI to send jobs to DIRAC with the help of transformation system
 - Generate CI jobs based on job options
 - CI jobs call DIRAC to submit jobs to the grid
- Use Celery and RabbitMQ for managing long-running jobs
- Summary is sent to the GitLab CI log
- Simple flask front-end for exposing detailed results



Proposed validation system

- Validation system proposed for CEPC
 - Based on the Github Action system
 - Re-use ideas/functionalities of LHCb
 - Some parts were already implemented/tested



Main features

- Could be easily integrated with the software validation system
 - Include build/unit tests, performance test and data production
 - Testing log and key distribution uploaded as test artifacts
 - Tests could be triggered on demand or regularly
- Task definition within yaml files
 - Test workflow: check out --> external libs --> build --> tests --> ...
 - What kind of tests should run on a daily/weekly/monthly basis
 - More flexible way of defining data production

Main features

- Tests are run on self-hosted runners:
 - kubernetes (within containers), DIRAC (grid resource)
 - Be able to support test matrix (various platforms)
- Messaging components for long jobs
 - Break with the Github limits for analysis jobs
- Automatic deployment:
 - CVMFS
 - Container
 - Web,

Main features

- CI test dashboard is being developed for easy monitoring



CI Tests ☆ 🔗

📊 📄 ⚙️ 🖨️ 🕒 Last 7 days 🔍 ↻

Daily Build Status

repository_name 🔽	conclusion ↑ 🔽
"rucio"	success
"STCF-Soft-CDR"	success
"k4FWCore"	success
"rucio-moniotring"	queued
"irisdb-kayobe-config"	failed
"virtual-pipelines-eventselection"	time out

Summary

- Validation toolkit for JUNO is introduced
 - Planning to redesign with new technology based on CEPC's needs
- Recent CI/CD ideas were inspired by LHCb
- A CI/CD system is proposed for CEPC
 - Most are raw ideas, feel free to comment
- Timeline:
 - Prototype this year
 - Fully functioning in 2022

Thanks for listening