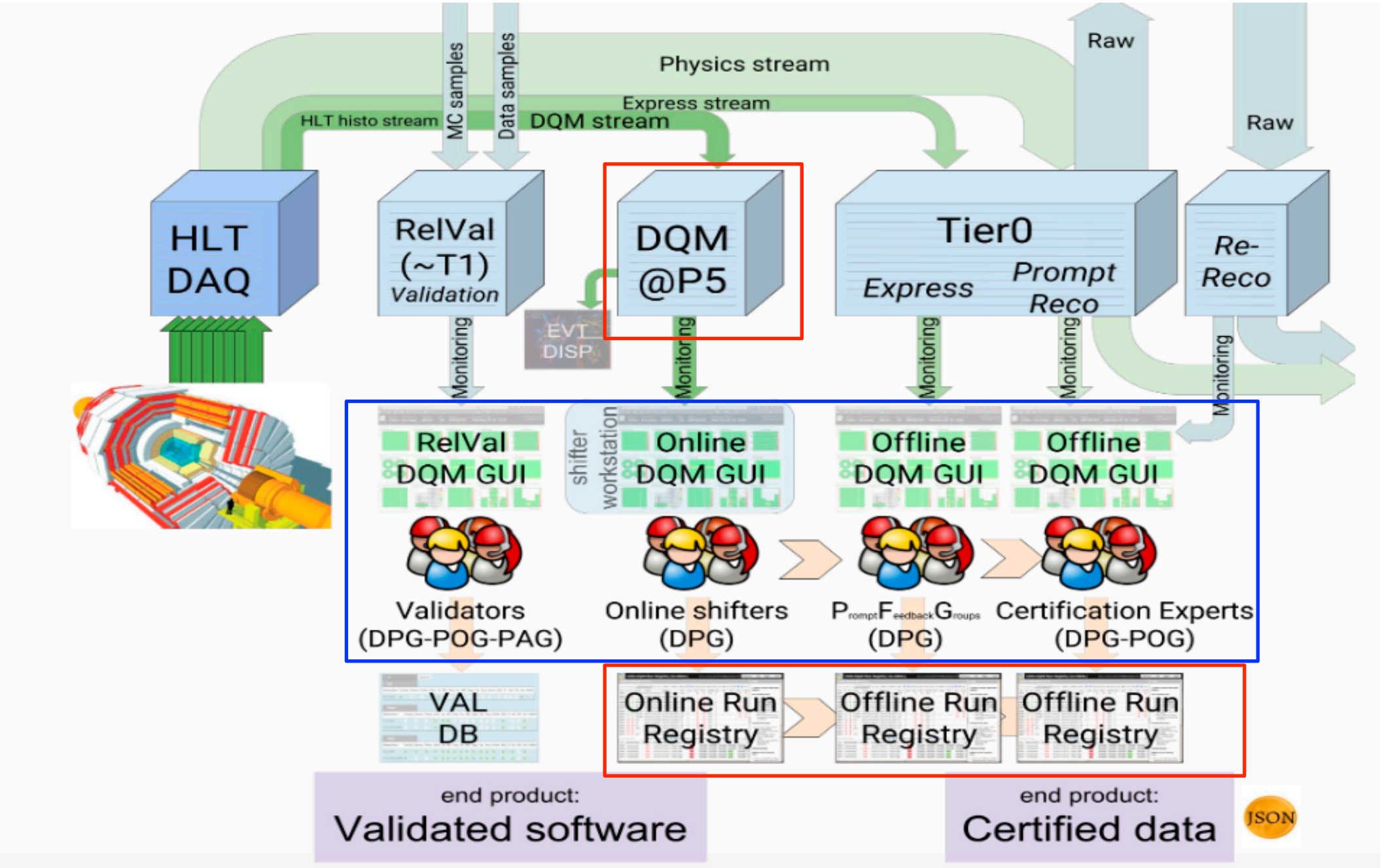
DQM system at point point 5



Development projects

- DQM Legacy GUI running well (from ~10 years) with all four flavors
  - ✓ **Developed new DQM GUI** with the newest technologies (easier maintenance, added more complex features, improve user experience)
  - ✓ Technologies: python, web tools, light C++ development
  - **Pending tasks:**
    - Integration of sound alarms in online GUI
    - Migrate to Kubernetes
    - Migrate relval GUI to new software, full integration with computing of offline GUI
    - Few cosmetics: Implement tinyurl, display server name
- **Infrastructure development for computational tasks (ML)**
  - ✓ A major goal of DQM is to increase the level of automation in data quality monitoring and data certification. There are several tools in their infancy that should be evaluated, and then a robust solution should be developed for our needs.
  - ✓ Technologies: python, web tools, ML
  - ✓ Evaluating existing statistical software from subsystems, developing design proposals
  - ✓ Development of a computational engine for fast statistical comparisons of histograms.
  - ✓ Integration of computational engines with DQM GUI (statistical tests, regressions, machine learning)

Operations and Maintenance

- **Maintenance of existing softwares**
  - ✓ Historical Data-Quality Monitoring (HDQM) tool (https://cms-hdqm.web.cern.ch/)
  - ✓ The Run Registry (https://cmsrunregistry.web.cern.ch/)
  - ✓ Pull Request Inspector tool (https://prinspector.web.cern.ch/)
    - Follow and Resolve CMSSW/cmsdist/deployment **issues**
- Some tools that require migration to kubernetes
- **Support/involvement in online DQM operations (data taking)**
  - ✓ To keep DQM machines up and running to make sure all DQM services are reaching to its users
- Other Offline duties (not in any precise order, random indeed)
  - ✓ **PR review** and interplay/synch with Online DQM PR merging (github)
  - ✓ Watch and answer/reply **queries and requests** to DQM group