# **ValDb**: an aggregation platform to collect reports on the validation of CMS software and calibrations

Antanas Norkus, Vilnius University (LT) antanas.norkus@cern.ch ***on behalf of CMS collaboration***

## PdmV group and ValDb service

Physics Data And Monte-Carlo Validation (PdmV) is a group of PPD inside the CMS experiment. PdmV tasks include evaluation of physics performance and validation of Monte Carlo and datasets samples, prompt reconstruction, new algorithms for physics object ID, analysis recipes and physics definitions, performance of physics algorithms for pile-up subtraction in data, global event description, tuning of the pile-up description in the MC, data/MC tuning and Run dependent MC, event generators and their tunings, effect of new Alca constants on Physics analyses. The physics objects, detector performance, analysis and trigger groups collaborate with PdmV providing monitoring code and people, responsible to inspect its output.

Validation campaign gets validated by comparing target and reference validation samples produced for that purpose. Automatic sample comparison is done with RelMon* and validation contacts needs to check the RelMon report and fill their report into **ValDb**. Each release added in **ValDb** can be linked to RelMon report, having all needed ingredients for validation in one place.

**ValDb** is a service for bookkeeping reports from all validation experts, meaning there is no need to remove a release or a single report. Each validation contact is responsible for ones specific area (Tracker,bTag,Tau etc.) allowing one to change status only of ones responsible area. To organize user permissions automatically, **ValDb** synchronizes user database each hour taking information from validation contacts e-groups page (user name, responsible area, e-mail for announcements) and HyperNews (e-mail in case its different from one specified in e-groups). E-groups is e-mail group list with specific details about each participant (usernam , email, comments) and HyperNews is a web-based threaded forum system with a two-way email gateway developed by SLAC.

**ValDb** has three types of users:

- guest user - can only read reports,
- validation contact - has permissions to read data and edit only their area in release report,
- admin – is able to read all data, add a new release and new users, modify and put validation reports.

Each software release cycle is organized in campaign to to ease following the changes cycle and keep all information in single place. Users can check all changes in detailed information for single area and see how validation evolved from *'NOT YET DONE'* to current state.

Web interface with the help of JavaScript and HTML5 capabilities is bookmarkable and can be shared by different people or saved for reference in documentation Twiki pages.

**ValDb** has a build it announcement capability to send e-mails to concerned HyperNews forums building up a threads of email on a given validation campaign. Further discussion with expert can then happen directly in HyperNews to solve issues. Sub-threads are created by sub-categories. This lets users to be informed directly to their emails in clean way for each validation campaign while comments and discussions and be a replies to the message by **ValDb**.

[*] RelMon is a tool to automatically compare two sets of samples and produce a report. For further information see: http://iopscience.iop.org/1742-6596/368/1/012008

## Used tools

AngularJS is JavaScript framework useful with two way data binding and dynamic data representation. Making each HTML page to be few hundreds lines, creating reports tables dynamically.

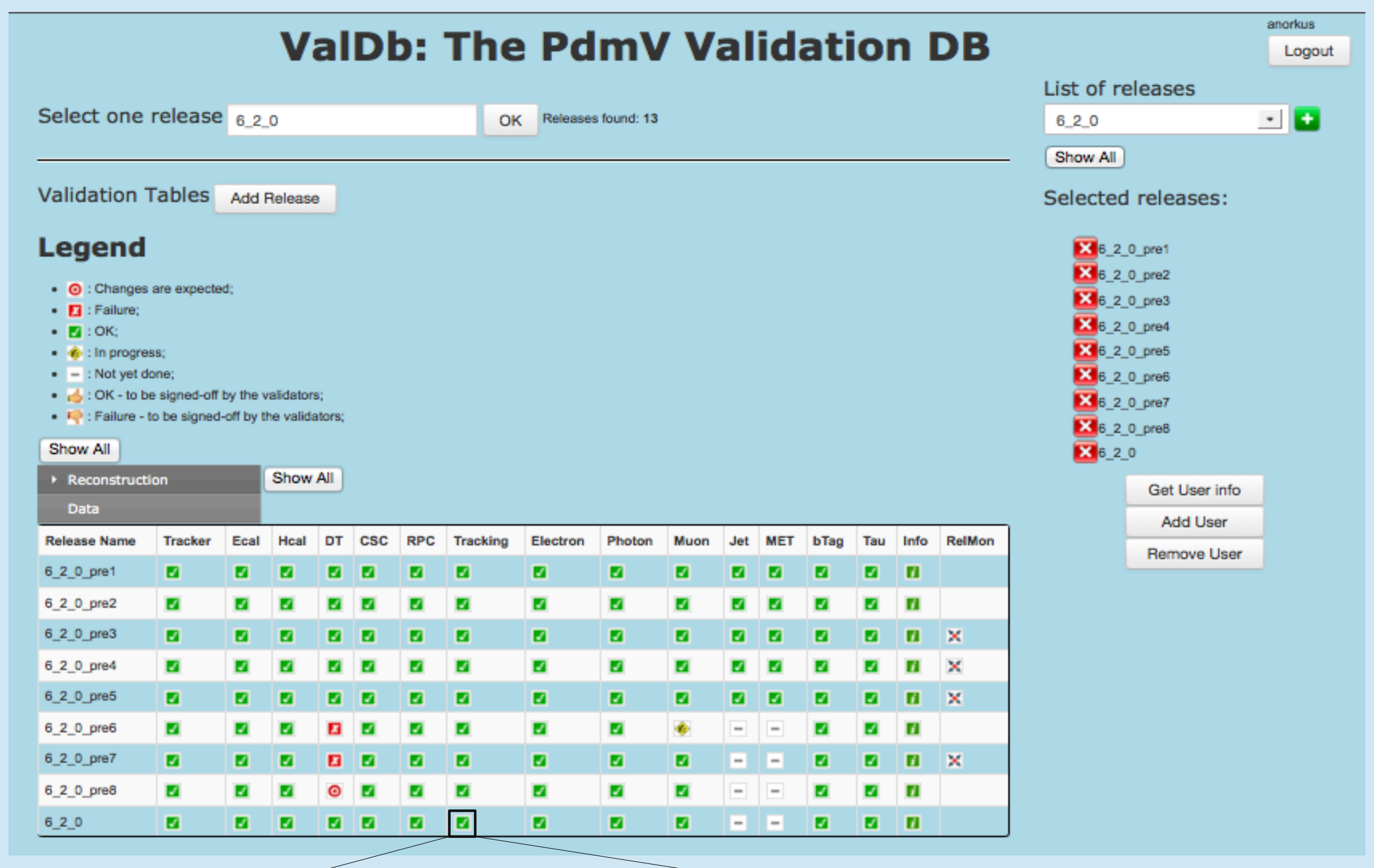
Twitter Bootstrap's HTML and CSS templates allow to create tidy and responsive web pages in almost no time. ValDb interface uses its CSS and modal pop-up to display detailed information.

CherryPy is Python web framework. Serving HTML files and managing user requests. Simple way to access HTTP request headers and data.

Python is a high-level programming language. Used to parse user requests and return data in JSON format, send emails over SMTP protocol, connect to database and logging.

Oracle database used to save user reports. DB schema allows to save newest reports and snapshot of previous report points

## ValDb interface



anorkus
Logout

**ValDb: The PdmV Validation DB**

Select one release 6_2_0 OK Releases found: 13

List of releases 6_2_0 Show All

Selected releases:

- 6_2_0_pre1
- 6_2_0_pre2
- 6_2_0_pre3
- 6_2_0_pre4
- 6_2_0_pre5
- 6_2_0_pre6
- 6_2_0_pre7
- 6_2_0_pre8
- 6_2_0

Get User info
Add User
Remove User

Validation Tables Add Release

**Legend**

- ⊙ : Changes are expected;
- ☒ : Failure;
- ☑ : OK;
- ✱ : In progress;
- − : Not yet done;
- 👍 : OK - to be signed-off by the validators;
- 👎 : Failure - to be signed-off by the validators;

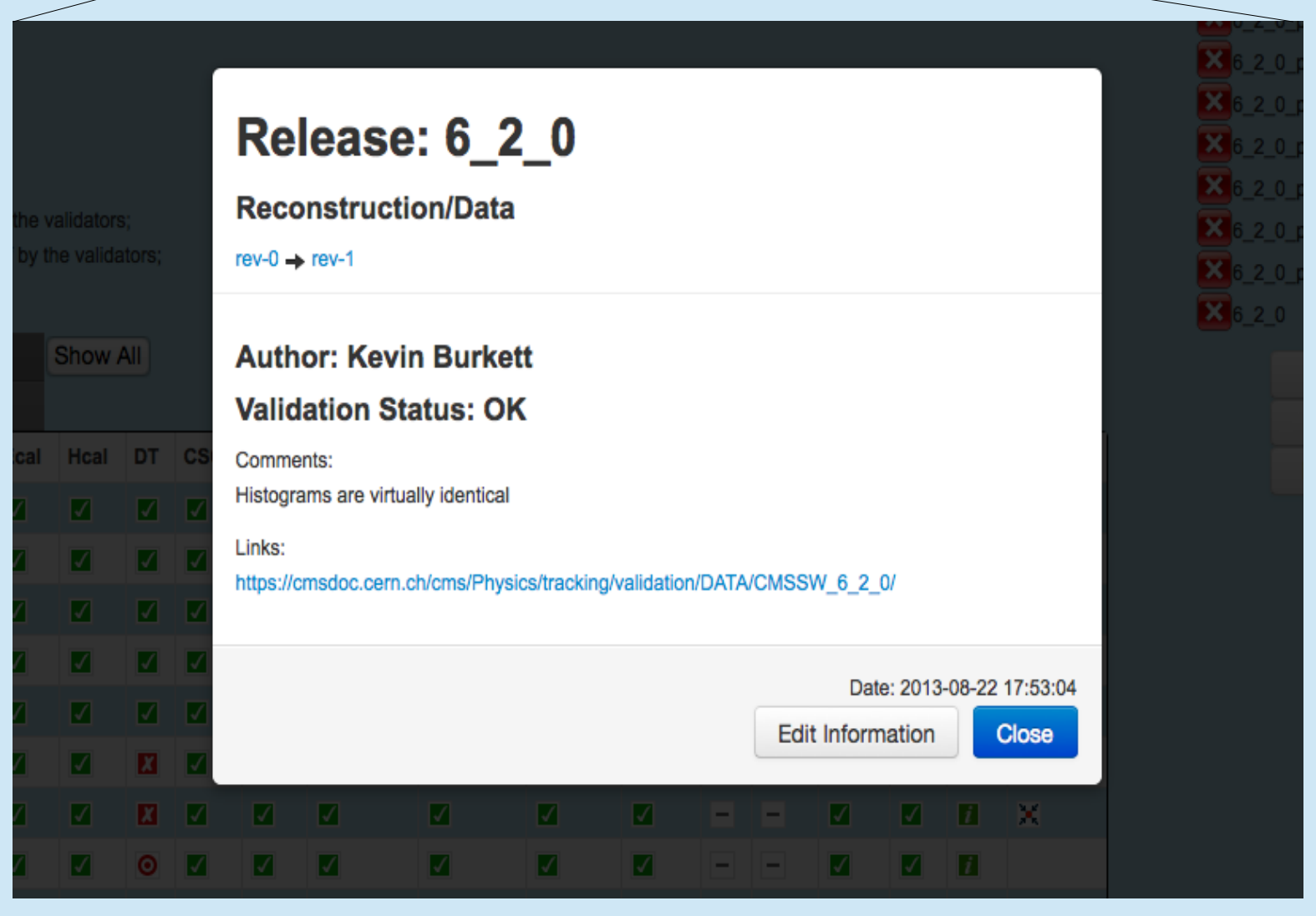
Show All

Reconstruction Show All

Data

| Release Name | Tracker | Ecal | Hcal | DT | CSC | RPC | Tracking | Electron | Photon | Muon | Jet | MET | bTag | Tau | Info | RelMon |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6_2_0_pre1 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | i |  |
| 6_2_0_pre2 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | i |  |
| 6_2_0_pre3 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | i | ✗ |
| 6_2_0_pre4 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | i | ✗ |
| 6_2_0_pre5 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | i | ✗ |
| 6_2_0_pre6 | ☑ | ☑ | ☑ | ☒ | ☑ | ☑ | ☑ | ☑ | ☑ | ✱ | − | − | ☑ | ☑ | i |  |
| 6_2_0_pre7 | ☑ | ☑ | ☑ | ☒ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | − | − | ☑ | ☑ | i | ✗ |
| 6_2_0_pre8 | ☑ | ☑ | ☑ | ⊙ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | − | − | ☑ | ☑ | i |  |
| 6_2_0 | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | ☑ | − | − | ☑ | ☑ | i |  |

Admin interface showing CMSSW_6_2_0 release cycle. Making it easy to draw overall conclusion on the outcome of validation campaign. Displaying all functions: add new release, add new user, find users, remove user.



**Release: 6_2_0**

**Reconstruction/Data**

rev-0 → rev-1

**Author: Kevin Burkett**

**Validation Status: OK**

Comments:

Histograms are virtually identical

Links:

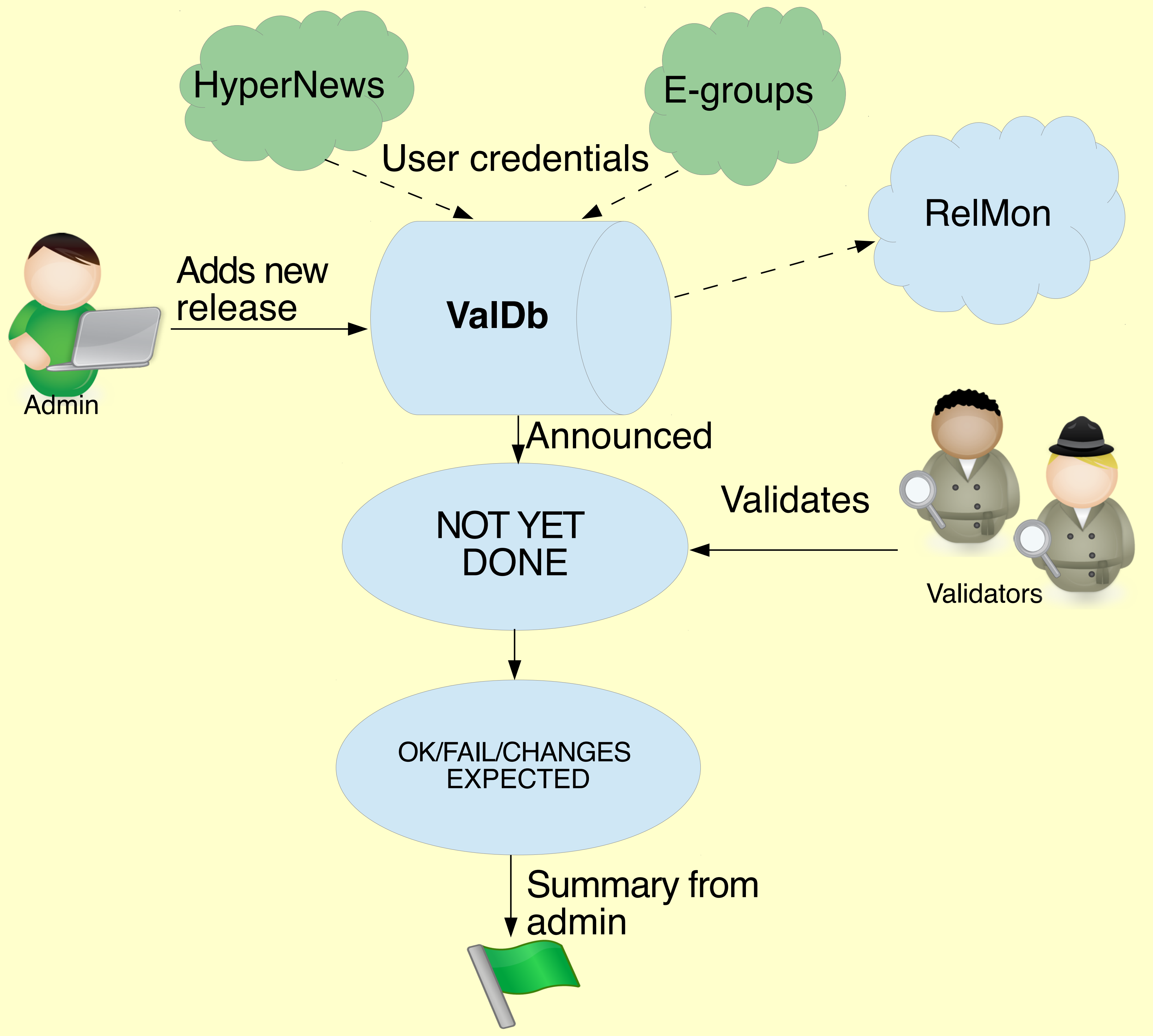
https://cmsdoc.cern.ch/cms/Physics/tracking/validation/DATA/CMSSW_6_2_0/

Date: 2013-08-22 17:53:04

Edit Information Close

By click on icon, detailed information is displayed in Twitter Bootstrap's modal. Showing user comments and links to validation plots, date when validation contact put ones report. The evolution of report can be check by clicking on older revision.

## ValDb schema



HyperNews
E-groups
User credentials
RelMon
Adds new release
ValDb
Admin
Announced
NOT YET DONE
Validates
Validators
OK/FAIL/CHANGES EXPECTED
Summary from admin

### ValDb operations

- User database is synchronized every hour by a python cron job. It takes user credentials (username, email) from validators e-groups and HyperNews.
- Admin adds new release to **ValDb** for validation. Entry can be linked to external RelMon report.
- **ValDb** announces new release via SMTP protocol to validation HyperNews. The announcement message is an starting message for threading and any discussions about validation.
- The validation status of newly added release is by default set to *'NOT YET DONE'.*
- Validation contacts puts their validation report and changes status of validation. **ValDb** sends a reply message to HyperNews, using *'In-Reply-To' header* inside e-mail to keep threaded structure inside HyperNews forum.
- All reports are filled and admin is able to summarize all information and perform a wrap-up.

### Possible upgrades

- Automatization of new release addition with minimal human interaction.
- Better integration with RelMon.
- Possibility to edit the link to RelMon report.
- Adding sub-categories dynamically.
- RESTful interface to get information about validation campaign.