

# Introduction

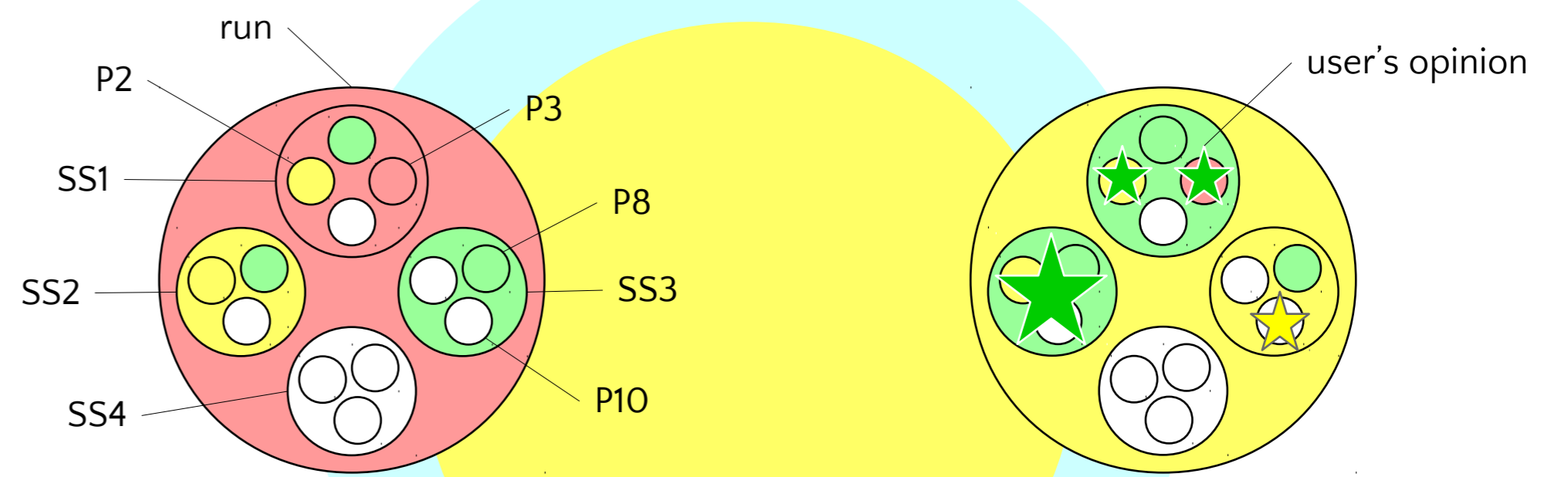
There was no SND DQM system until 2018. The team could make use of detailed events/errors log and thousands of histograms generated for every run (approx. 30m-2h) to unveil problems in time. Everything would have been fine if the team had been a bunch of robots. An operator could miss something while reading endless logs, histograms had to be viewed one by one. The only runs quality information was in a paper log. The data processing involved collecting this information and redoing some checks. The problem was lack of a user-friendly interface and a unified data quality database.

# Requirements

- Show run state, parameters and histograms.
- Assign parameter quality by software and/or by user.
- Group parameters by subsystems, subsystems by run.
- Provide multiple parameter sets for different usages (for operators/experts/debug/new conditions)
- Select runs quality information by various criteria.

# Estimating Run Quality

- There are some parameters (histograms and numeric values).
- The parameters are put in sets (a set for operators, a set for calorimeter experts, etc.) grouped by SND subsystems.
- Scripts and users estimate parameters quality.
- A user can override automatic decisions for a parameter or a subsystem.
- A subsystem quality is the worst quality of its parameters.
- A run quality is the worst quality among the subsystems.



**Example.** Parameters have "blank" (no quality estimate), "good", "doubt" (an expert should decide) and "bad" quality estimates. In the case illustrated parameter 3 of subsystem 1 is "bad". So it does not matter if other SS2 parameters aren't "bad". SS2 quality is "good". Similarly, parameter 8 being "good" defines SS3 quality while it is ignored that P10 is "blank". The run quality is "bad" because SS1 is "bad".

A user reestimated qualities of parameters 2, 3, 10 and subsystem 2. It resulted in changing SS1, SS2, SS3 and run qualities. SS1 and SS2 became "good", SS3 became "doubt". SS3 defined the run quality ("doubt") being the subsystem performing the worst.

Monthly view for the run coordinator to ensure operators and scripts work fine.

EMC

Полное имя ЕММ для событий по FLTI

Operator view for easy checking results.

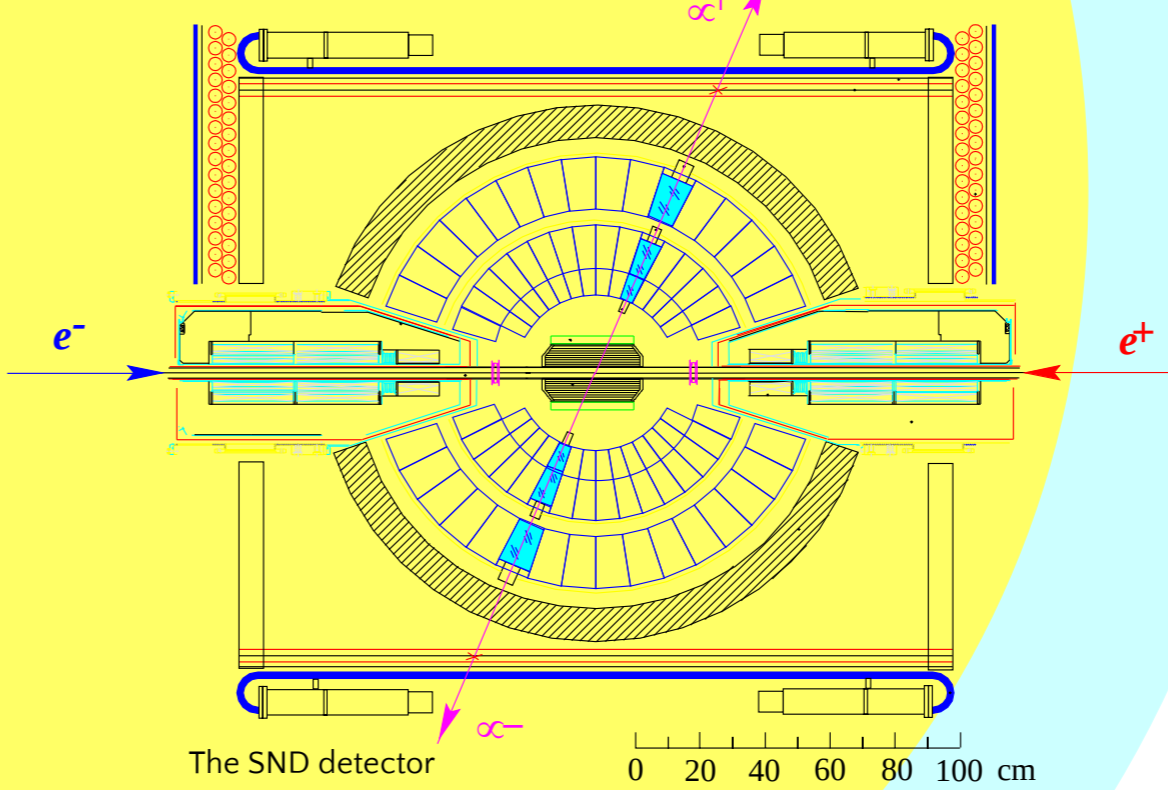
Expert view for displaying parameters in detail.

# The DQM system for the SND detector

Konstantin Pugachev<sup>1,2</sup> (K.V.Pugachev@inp.nsk.su)

<sup>1</sup> Budker Institute of Nuclear Physics, Russia

<sup>2</sup> Novosibirsk State University, Russia



see more: AIP Conference Proceedings 1735, 030009 (2016); <https://doi.org/10.1063/1.4949392>

Пугачев К.В. | 11.02.2019 09:00:00 (день)

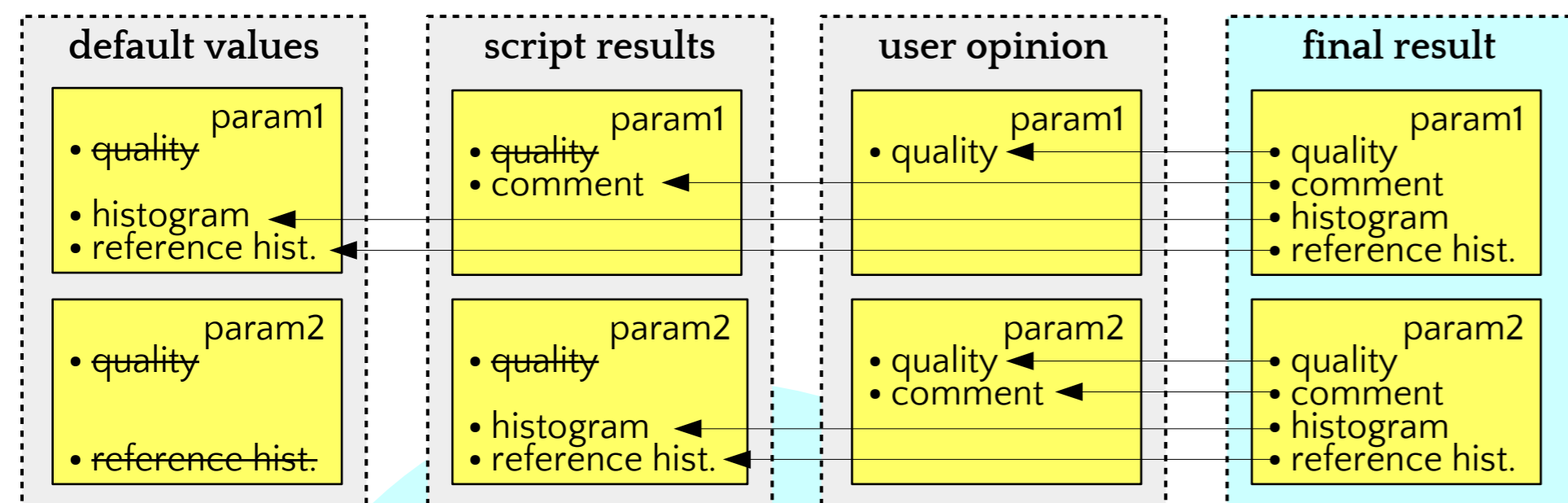
предыдущая смена | следующая смена | текущая смена | список смен | состояние эксперимента | чеклист (100%)

ШАПКА		РЕЖИМ										БУХГАЛТЕРИЯ					КОММЕНТАРИИ				
Время	Номер	$\epsilon$ , МэВ	$I \times L$ , мА	$\gamma_{ee}$ , мкб <sup>-1</sup> с <sup>-1</sup>	$N_e$ , Гц	$\% T_{\text{длв}}$	$f_{\text{LDC}}$ , кгЦ	$I_{ee}$ , нб <sup>-1</sup>	$N_{\text{REC}}$	$N/L$ , мб	Причина остановки	стадия	качество	тип	комментарий	проверил					
04:50-05:25	39221	625	69 × 72	11.1	520	92	73	21.4	1.0M	47	# событий	обработан	хороший	experiment		Пугачев К.В.					
05:26-06:01	39222	625	67 × 65	10.2	505	92	70	20.2	1.0M	50	# событий	обработан	хороший	experiment		Пугачев К.В.					
06:06-06:41	39223	625	68 × 68	10.8	520	92	73	20.8	1.0M	48	# событий	обработан	хороший	experiment		Пугачев К.В.					
06:41-07:18	39224	625	66 × 65	10.2	495	92	69	20.7	1.0M	48	# событий	обработан	хороший	experiment		Пугачев К.В.					
07:18-07:37	39225	625	60 × 59	8.4	455	93	61	9.0	482K	54	оператор	обработан	хороший	experiment		Пугачев К.В.					
14:12-14:24	39226	625	30 × 1	0.12	229	96	15	0.079	156K	1974	оператор	обработан	плохой	experiment	# Тестовый заход	СНД					
18:49-18:56	39227	625	55 × 56	7.7	459	91	60	2.9	172K	60	оператор	обработан	хороший	experiment	# В конце I=0	Пугачев К.В.					
19:02-19:23	39228	625	63 × 62	9.5	513	91	71	10.6	575K	54	оператор	обработан	хороший	experiment	# В конце I=0	Белобородов К.И.					

Quality data in the runs table.

# DQM Scripts

A DQM script is a ROOT macro that analyses subsystem data and yields run parameter values and/or quality data.



```
void script_example( // script arguments are specified in the DQM configuration
int run, // this run number
const char * hists // histograms ROOT file path or NULL
) {
if(hists == NULL) {
parameter("param1").quality(QBAD).comment("No histograms!");
parameter("param2").quality(QBLANK).valueNull().refNull(); // unset values
} else {
// check the histograms somehow (e.g. by rolling dice)
bool p1IsGood = Integer(2), p2IsGood = gRandom->Integer(2);

parameter("param1")
.quality(p1IsGood ? QGOOD : QBAD)
.comment("Checked using lazy Monte-Carlo method.");

parameter("param2")
.quality(p2IsGood ? QGOOD : QBAD)
.valueHist("CL/h29")
.refHist(32200, "CL/h29");
}

flush_parameters();
}
```

# Implementation Details

The DQM is included in the SND information system that is a Node.js application using the following tools: MySQL, ROOT, JSROOT.

