AUTOMATED AVAILABILITY STATISTICS

Don’t do it yourself !
Let the control system do it …

P. Duval, M. Lomperski, H. Ehrlichmann, DESY, Hamburg, Germany
J. Bobnar, Cosylab, Ljubljana, Slovenia

Abstract

The availability of a particle accelerator or any large machine with users is not only of paramount importance but is also, at the end of the day, an oft quoted number (0 to 100%) which represents (or is taken to represent) the overall health of the facility in question. When a single number can somehow reflect on the maintenance, operation, and engineering of the machine, it is important to know how this number was obtained. In almost all cases, the officially quoted availability is generated by hand by a machine coordinator, who peruses the operation statistics over the period in question. And when humans are involved in such a calculation there might be a latent tendency to avoid the stigma of low availability. So, not only might there be scepticism at 'impossibly high' availability, but comparing quoted availability from one machine with another might turn out to be virtually meaningless.
We present a method for calculating the machine availability automatically, based on the known (and archived) machine states and the known (and archived) alarm states of the machine. Ideally this is sufficient for a completely automatic determination of the availability. This requires, however, a perfect representation of all possible machine states and a perfect representation of all possible fatal alarms (those leading to down time). As achieving perfection is ever an ongoing affair, the ability for a human to 'post-correct' the automated statistics is also described.

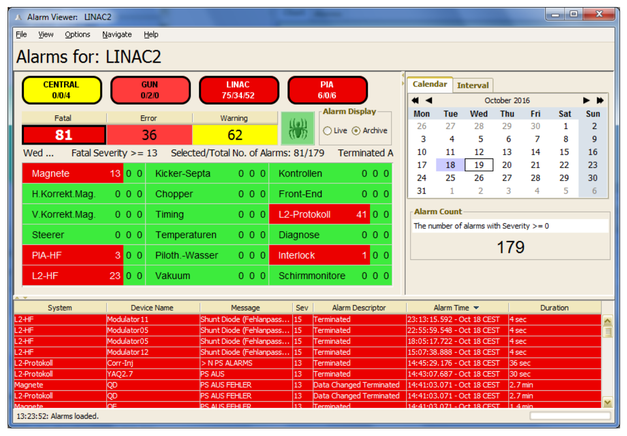
Ansatz

“if the machine is not available then there must be at least one fatal alarm in one of its subsystems.”

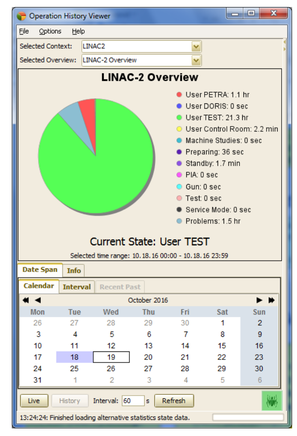
Motivation

- Remove human bias
- Reduce workload on machine coordinator
- Monitor Availability on-line

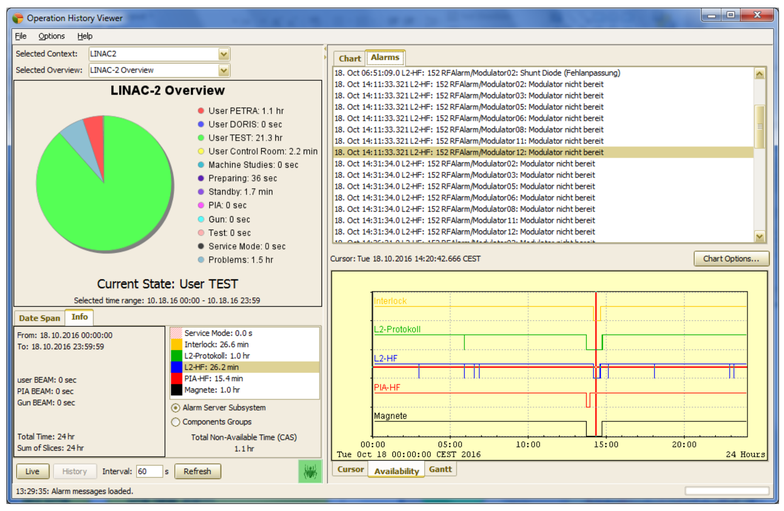
Alarms



States



Use the Alarm system and the Operation State history to deduce and assign blame for non-availability …



Required Services

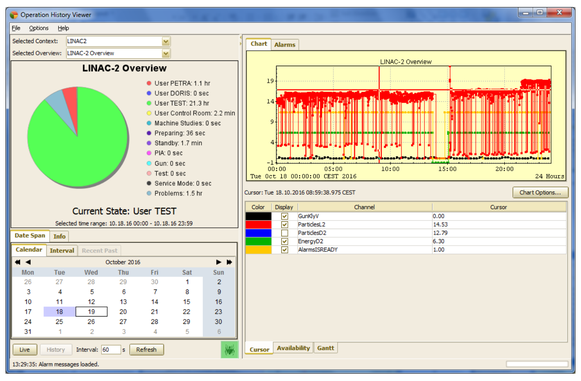
State Server
- Well-defined list of possible machine states
- Declared externally (by operator)
- Status of declared state
  - with/without beam (running)
  - with/without fatal alarms (problems)
- Bean counting: integrated time spent in any one state

Alarm Server
- Well-defined alarm subsystems
- Fatal alarm statistics
- Archive of all alarms
- ‘is ready’ = absence of any fatal alarm (availability)
- Bean counting: integrated time where a subsystem has at least one fatal alarm

bean counting … (how many seconds spent in Machine Studies?) →

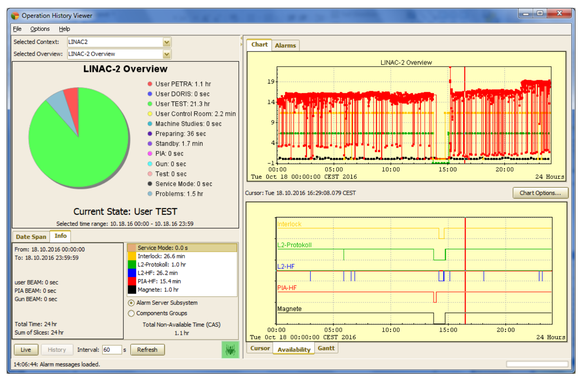
Archive Server
- Archive the bean counts for each state
- Archive the bean counts for the ‘is ready’ information

A Day in the Life of the PETRA Linac



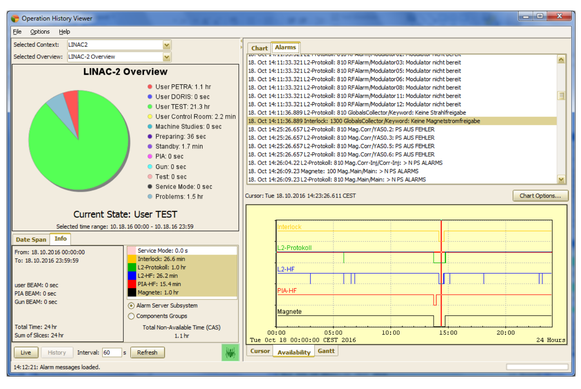
24-hour history of selected machine parameters and “AlarmsISREADY” (a boolean value):
1 => machine is available
0 => machine is not available

We can select any time range we like either with the calendar or by zooming on the trend chart …



Why was the machine not available? Which subsystems were responsible?

Only those subsystem with at least one fatal alarm are displayed in the ‘Info’ and ‘Availability’ tabs …

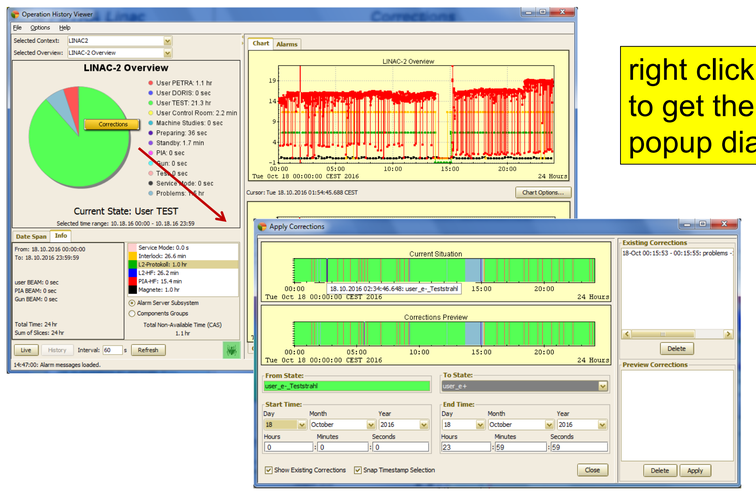


Where and what were the fatal alarms?

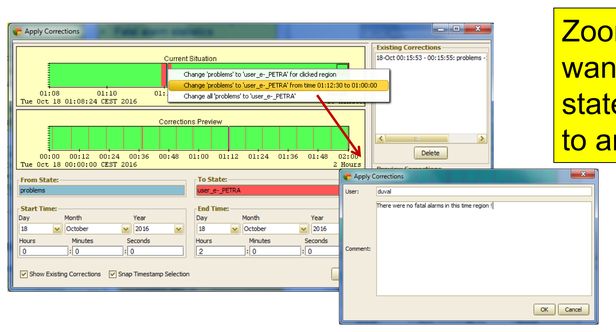
Philosophical Question:
is *problems* a declared state?
or is *problems* an attribute ?

time spent in *problems* and total non-available time are not equal !
Why ?

Corrections



right click on the pie-chart to get the corrections popup dialog …



Zoom in as much as you want and right click on the state you wish to assign to another state …

but: you have to fill out a ‘form’ !

Notes: this is an on-going project (maybe forever …)
- Alarms (Why were they there? Why weren’t they there?) are where most of the *action* is …
- Corrections are currently necessary in the trail-and-error phase !
- Can also correct the availability with another dialog popup !

To do:
- Consistency checks
- Automatic notification to responsible parties if there’s a correction

http://tine.desy.de

it doesn’t always have to have a *slick* web site to be good …