THE OPERATION EVENT LOGGING SYSTEM OF THE SLS

A. Lüdeke, SLS, Paul Scherrer Institute, Villigen, Switzerland

Abstract

Modern 3rd generation synchrotron light sources aim for 100% availability. No single beam interruption is acceptable and every distortion of operation should be investigated: What caused the interruption? Can it be avoided in the future? If it can't be avoided, how can the recovery be accelerated? An automated event recording system has been implemented at the Swiss Light Source (SLS) in order to simplify those investigations. The system identifies operation distortions and records automatically type and duration of the event. Relevant information of the event, like control system archive data or shift protocols, is linked to the event and presented in web pages. Each event will be assigned to a failure cause. Means to filter the events are provided. We will describe the concept and the implementation of the system at the SLS and our experiences with it.

MOTIVATION

A large fraction of the work of an operations manager is dedicated to the analysis of operations data. Each distortion of the beam quality needs to be detected, the reason for the distortion should be identified and a repetition of the distortion should be prevented. The procedure is always in three steps: a) identifying an undesired operation state (called in the following an "event"), b) assigning this "event" to a cause and c) define actions to take. While the first part can be easily automated, the latter two depend on human interpretation of the facts. But the decision making process can be supported by applications: each type of "event" requires a certain set of information to decide on the cause and reliable statistical information on the different "events" are a prerequisite for an efficient planning of resources.

An automation of the event identification does not only save time. It requires a defined metrics for "undesired operation states" and it assures that all events according to this metrics are identified. This provides a reliable base for decision making.

SPECIFICATIONS

The operation event logging system does an automatic recording of defined "events" during operation of the light source. The events are detected from control system variables by start-rules and stop-rules, both depending on the operation mode. A set of references to related information are defined for each event type and is added by the logger for each event.

In order to reference all related data of an event in a useful way, all those data sources need to support references. A simple and efficient way for supporting references is by web interfaces that allow to reference data by unified resource locators (URLs). The main sources of information for operation at the SLS are the EPICS channel archiver, the EPICS alarmhandler and the electronic log book. All of them have been customized in order to provide web interfaces that allow to access data of a time interval by an URL. [1]

One or more causes can be assigned to each event. Each cause is defined by two categories: the affected area, e.g. "Linac", "storage ring", etc., and the category of the failing system, like "RF" or "Magnets". A comment describes the actual failure. If more than one cause is assigned to an event, the total event duration is split up into fractions for each cause.

A web-browser interface to the event database allows to query it and to filter for specific event types, durations and causes.

GENERIC IMPLEMENTATION

Automatic Event Logging

An event is defined by the *event_rules* database table. It has a *start_rule*, a *start_delay*, a *stop_rule* and a *stop_delay*. An event of a certain type can have multiple definitions depending on the operation mode of the light source. Operation modes, as defined in the *operation_category* table, are *"User Operation", "Beamline Tests"*, *"Machine Tests"* and *"Shutdown"*.

If the start-rule of an event is evaluated to true for the current operation mode, a counter is started. If the counter exceeds the start-delay, the event start is recorded. If it evaluated to false before the start-delay has been exceeded, the counter is reset to zero. After the event start the event stop-rule is evaluated until it is true for at least stop-delay seconds. The event stops as well in case of a change of the operation mode. Then the event is recorded to the database. Event-start and event-stop dates are defined as the beginning of the delay interval. E.g. if a stop-delay is one hour, the event entry is written to the database one hour after the event-stop date.

The start of one event can stop events of other types. E.g. a "beamdrop" event, signifying a distortion of top-op operation, ends with the start of a "downtime", that shows that the beam-current dropped below 50 mA. Those event relations are defined in the *event_precedence* table.

Figure 1 shows a state diagram of the event logger for one event type. Each event has it's own state set, but by

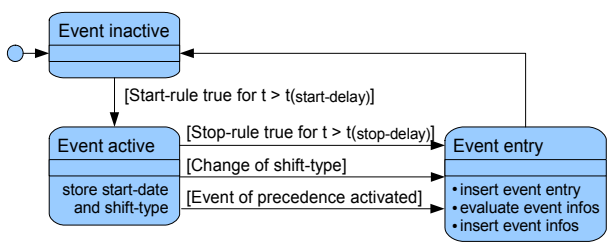

Figure 1: generic state diagram for each event-type

the event precedence each event activation can deactivate other events.

Actual events are recorded in the *event_entry* table. The *event_start* date is the primary key of this table, the other information is *event_stop*, *event_type* and *op_type* in reference to an *operation_category*. The table constraints take care that only defined *event_type, op_type* combinations can be added with valid dates. Each event entry can have many *event_info* entries. Those contain comments for an event defined by its *event_start*. Automatic entries in the *event_info* table are made according to definitions in the *event_links* table. The table defines for each *event_type* a list of commands to be executed. For each command the timestamps for the start and the end of the event are passed as arguments. The output text is then written to the *event_info* table.

It is possible to manually add more infos later on to a specific event. The table *event_info_hist* keeps a history of delete and update transactions for the event_info table.

Event Cause Assignment

The operation experts manually assign one or more causes to the event after its automatic generation. The script “cause” or the graphical user interface “xcause” allow privileged users to assign a cause to an event entry. A cause is defined by the *event_start,* a *cause_area*, a *cause_system,* a *fraction* and a *comment*. All event causes for one event should add up their fraction to 100%. The possible cause areas are defined in the event_area table by a two letter *area_ID* and a description, e.g. 'LI' for 'Linac'. The system categories are defined in the table event_category again by a two letter *sys_ID* and a description, e.g. 'RF' for 'Radio frequency'.

Both applications allow to show those events that do not yet have a cause assigned or where the fractions of all assigned causes do not add up to 100%.

All delete or update transaction to the table *event_cause* are logged in the *event_cause_hist* table, together with the modifying user and the modification date.

Event Browser and Statistics

Web pages, applications and command line tools have been provided to browse and modify the event database. Figure 2 shows a screen shot of the event browser web page. The browser allows to select events of a given time range. It provides filters for the operation mode, the event type, cause area, cause system and event duration. All information to an event is shown in a single row, including the list of causes and their descriptions. The last column “Info” contains the automatic and manual information for the event. The three entries visible for each event in figure 2 are the automatically added links to the web retrieval tools used for the SLS (see next section). The command line tools 'event' allows to add or delete events afterwards, e.g. in case of control system errors. Deletion is restricted to privileged users. The tool 'EVENTDB_info' allows to add additional information as *event_info* entries manually.

A script 'EVENTDB_stat' has been provided for statistical analysis of the event data. It generates for each operation mode and event type an event length distribution, a sum of the event durations for each cause system and for each cause area.

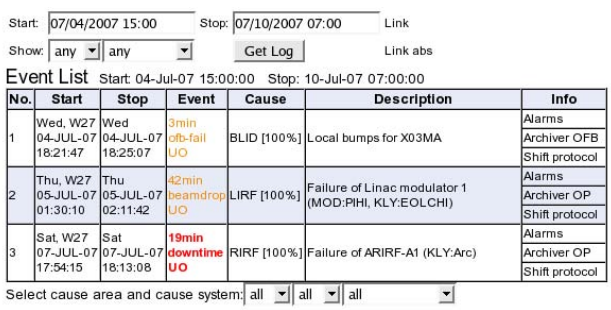

Figure 2: Event browser web page.

SLS SITE SPECIFICS

The generic application has to be configured by rule definitions of the desired events. The currently defined event rules for the SLS are shown in table 1.

Table 1: Event rules defined for the SLS. Rules are defined for the operation modes user operation (UO) and beamline tests (BL)

Event	start stop	rule	op.-modes	delay [s]
downtime	start	beam current $I_{beam} < 50$ mA	UO, BL	1
downtime	stop	top-up current $I_{top\text{-}up}$ has been reached	UO, BL	3600
beamdrop	start	$I_{beam} < I_{top\text{-}up}$	UO BL	150 300
beamdrop	stop	$I_{beam} > I_{top\text{-}up}$	UO, BL	3
ofb-fail	start	orbit feedback switched off	UO BL	10 900
ofb-fail	stop	orbit feedback running	UO BL	20 900
blow-up	start	beam height $\delta_y > 15$ µm or beam width $\delta_x > 70$ µm	UO	3
blow-up	stop	$\delta_y < 15$ µm and $\delta_x < 70$ µm	UO	60

Event precedence is only defined for “downtime” events: they do stop “beamdrop”, “blow-up” and “ofb-fail” events.

All control system variables that are used in the event rules are define in the *event_pv* table. The table just keeps a short variable name to be used in the rule-formulas and the control system process variable name to connect to.

Currently there are three automatic event infos defined for each event type:

- A link to the shift protocols of the event.
- A link to the alarms during the event.
- A link to the archiver.

For “downtime” and “beamdrop” events the link to the archiver shows the beam current. For “ofb-fail” it shows status information of the orbit feedback. For “blow-up” events the horizontal and vertical beam sizes are shown.

We define 20 different systems for event causes and 8 cause areas. While the causes for most events are added by the operation manager, the "ofb-fail" event causes are added by the feedback experts.

EVENT DB EXPERIENCE

The event logger is in operation since August 2006. It has recorded a total of 736 events: 247 downtime events, 136 orbit feedback failures and 150 beamdrop events during user operation and another 142 downtime, 12 ofb-fail and 49 beamdrop events during beamline tests. The blow-up events have been added only recently and no event has yet occurred. The average time between the recording of an event and the assignment of a cause has been two days.

About a third of the events during beamline tests were scheduled for a purpose. This is simply reflected by assigning the cause "scheduled" to those events.

Several SLS applications are using the event information. The events of a shift are automatically added to the shift protocol. A weekly user run overview is generated as a web page and shows the event and cause information marked in plots of the main control system variables, like beam current, beam sizes and beam lifetime (see Fig. 3). Those overviews are used to discuss the operations performance in the biweekly operation meetings and to decide on measures to improve the performance, if necessary.

The script for statistical analysis of the downtime is used in yearly reports. Figure 4 and 5 show some statistics from the event DB. Those reports are used for strategic decisions of upgrade and maintenance plans.

The event browser is used for the daily analysis of faults. It forces the operation manager to document each outage in a timely manner and allows everyone to get a quick update on recent operation distortions. It helps all system experts to answer questions like: has a similar problem occurred earlier? Is a particular klystron arcing more than others? Which events caused the longest beam interruption last year?

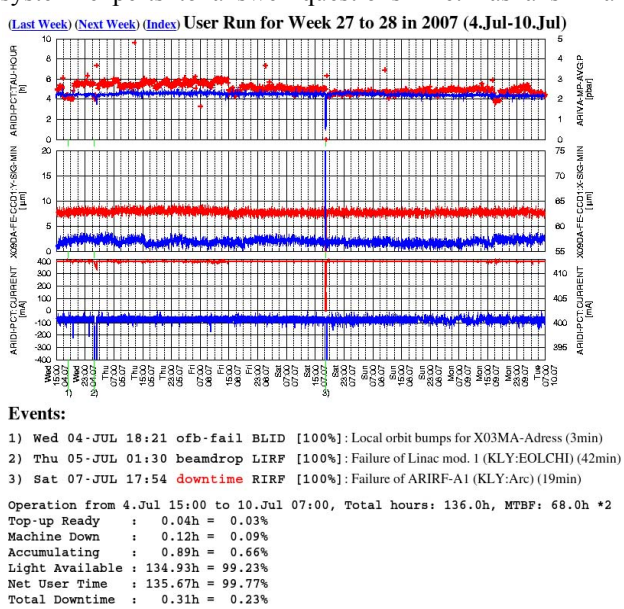

Figure 3: User run overview with events marked

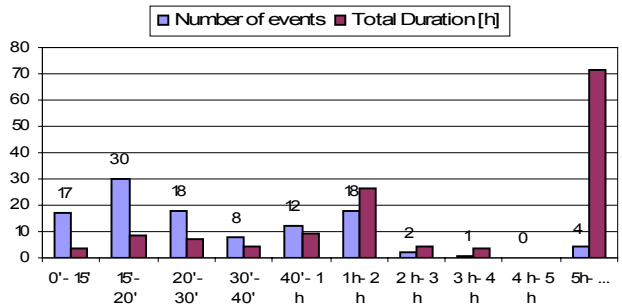

Figure 4: Durations of downtimes in 2007

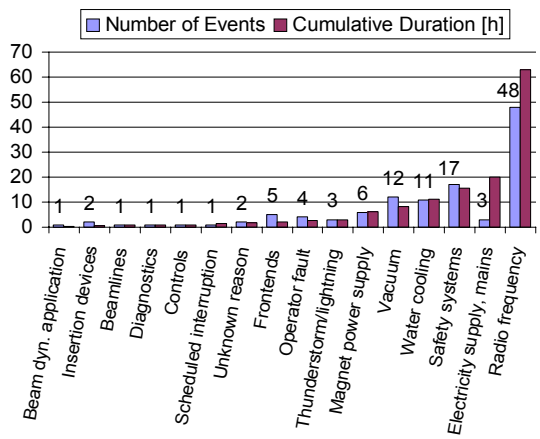

Figure 5: Downtime events and duration per system, generated from the event DB

SUMMARY AND OUTLOOK

The operation event database proved to be an extremely valuable tool for operation management of the Swiss Light Source. It simplifies the tasks of the operation manager and helps to prioritize maintenance and performance upgrade plans.

We plan to extend the event types for outages of the other SLS feedback systems: the filling pattern feedback, the multi-bunch feedback, the tune feedback system and the top-up control. Often the recognition of repeating failures of those systems is not trivial, since the feedback is immediately restarted by the operator.

The alarm log of a downtime event can be used to automatically suggest a cause for the beam loss. E.g. machine protection interlock signals, RF failures and vacuum interlocks can be easily determined by simple filters on the alarms. This extension will help the operators to handle complex failures and further enhance the accuracy of the cause assignments.

REFERENCES

[1] A.Lüdeke, "SLS Operation Management: Methods and Tools", EPAC'06, Edinburgh, UK, June 2006, p. 2715 (2006)