Saftlib d-bus replacement and API changes

D-bus Performance issues

- ## Late RT-Actions

  - 2 ms to FESA action, up to 5ms

- ## Late Function Generator Feeding

  - FG not armed at trigger time

  - Still running when armed

  - PowerSupply Abort / Flush

- ## Unexpected Interrupts

  - If Abort / Flush interrupts are not received saftd will reset its state

  - LM32 / saftd state mismatch

Improvements with no API change

- ## Saftbus

  - FESA / FESL work with no code changes

- ## Interrupt handler

  - IRQ handling uses the glib mainloop

  - FG state updates sometimes too late causing unexpected IRQ

  - Separate IRQ handling and X-bus request handling

- Property updates

    - Status changes broadcast d-bus messages to update proxies so that clients can read status locally rather than wait for a dbus call. It generates heavy traffic.

    - Replace properties with get methods (driver interface definition)

    - Change properties to on-demand (redesign proxy)

    - Stop updating certain properties

        - ActionCount causes a dbus signal before the timing event

        - Error counters

Improvements requiring API change

- ## Signals

  - Reduce the number of signals generated. Signals are always sent to dbus even if no-one is listening.

- ## Add blocking wait-for-event call

  - Simpler for clients than callback / wait / signal

  - Matches saftbus pipe read model

- ## Glibmm

  - Limit the available types

  - Remove conversions esp. Variant

  - Remove a dependency

Improvements requiring API change

- ## Saftlib internal events

  - A driver that can trigger actions within saftd

  - ActionSink / Condition that do not generate signals

  - e.g. SaftlibActionSink

    - Register for timing event

    - Extract BeamProcess from eventID

    - Load/Arm Function Generator for BeamProcess

- ## Shared Memory

  - Currently being integrated into PowerSupply

  - Data copied during Set instead of RT-Action

  - Loading LM32 takes most time