Improve performance of subscription to events. Look into Cumbia and see how it parallelized subscription!

Participants

- Nicolas, Reynald (ESRF)
- Giacomo, Alessandro (Elettra)
- Sergi, Zibi (ALBA)
- Gwen (SOLEIL)

Minutes

- Zibi explained the problems in Taurus:
    - Taurus GUI subscribes to hundreds of attributes ATTR_CONF_EVENT and CHANGE_EVENT events in the main thread - in case of RF plant GUI this takes approx. 7 s.
- Reynal explained that subscriptions to one device server are serialised by the admin device serialisation monitor.
- Nicolas explained that subscription id is generated by the tango core locally on the client side.
- Reynald analysed the code and explained that there are two locks on the client side:
    - Per callback monitor lock
    - Lock for the event map
- GIacomo explained QTango and Cumbia solutions:
    - QTango uses one thread per device.
    - Cumbia uses threads but load balances with a pool of threads.
    - Cumbia is reading the configuration and not subscribing to ATTR_CONF_EVENT.
- Nicolas and Reynald had to leave after the first hour. We agreed on next actions:
    - Zibi will share the tests platform
    - Kernel experts will look into the code separately and meet afterwards:
        - Gwen will check the Java client side.
        - Reynald and Nicolas will check the C++ client side.
    - Giacomo will provide startup time results of 2000 attributes app
- Giacomo showed that push_event() proceeding from the subscription (first thread) is executed in the current thread and not the event consumer thread.
- Sergi made tests with stateless=True subscriptions. Asynchronous subscription is a very similar concept - we return immediately the event id and then subscribe gradually in the background thread (keep alive thread).