



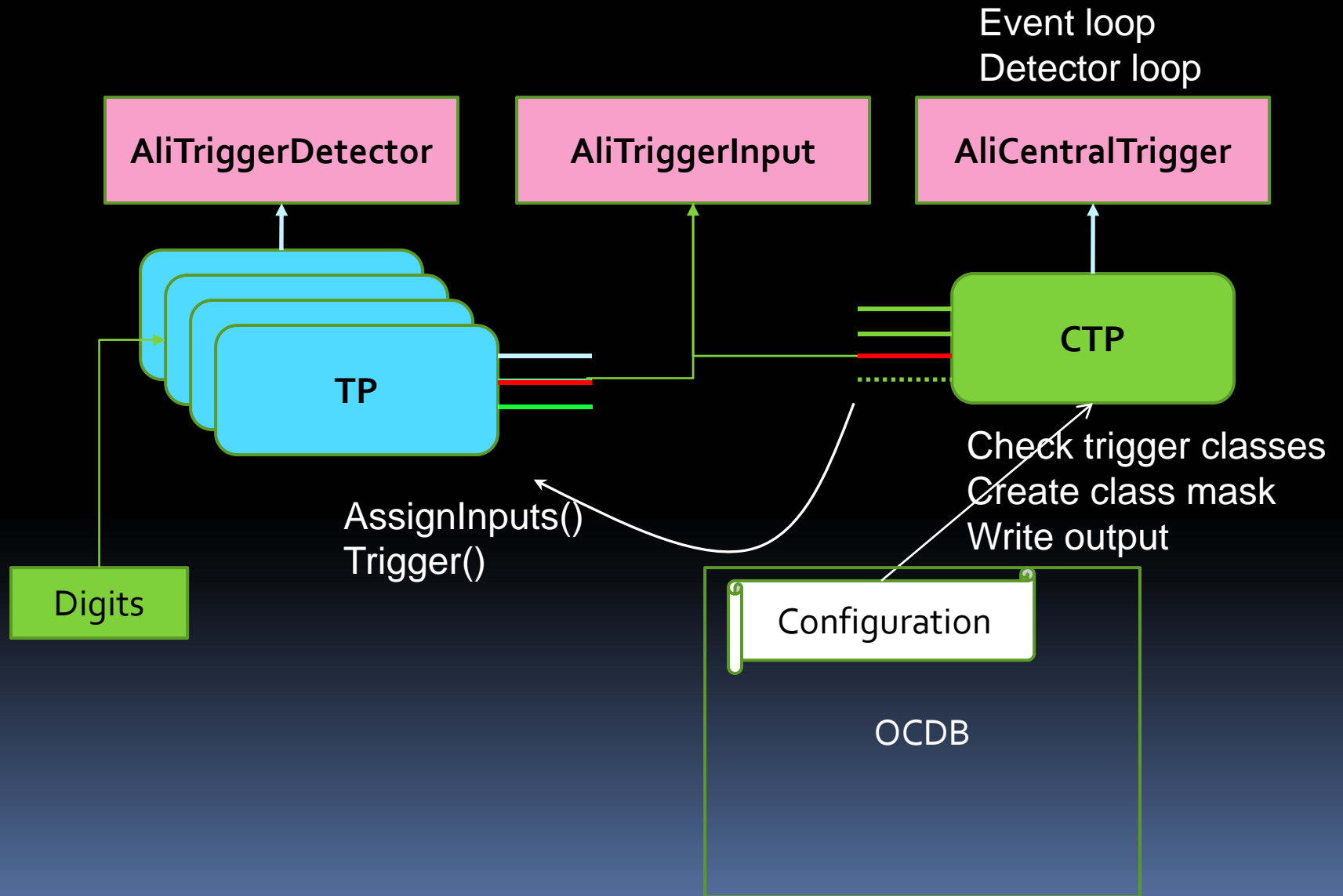
A.Gheata, ALICE offline week 22 June 2009

# SKETCH FOR A NEW TRIGGER SIMULATION FRAMEWORK

# Purpose

- Emulate decision of the whole trigger chain starting from real or simulated inputs
  - Handle dependencies and interaction between different components (inputs, outputs, configuration)
  - Handle time-dependent behavior of the trigger (trigger levels, BUSY time, latencies, ...)
- Be able to replay and validate trigger decision from raw data or digits information
  - Globally or for individual trigger processors
- Foresee the trigger response for custom trigger settings (trigger studies)

# What exist already



# What the framework will do

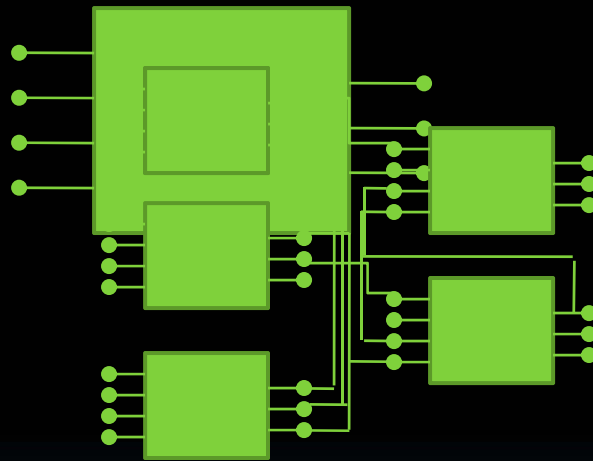
- Provide a template to simulate a generic trigger processor (TP) and its response function
  - Integrate existing algorithms
- Provide a way to handle and propagate signals between connected TPs
  - Steer main event and detector loops
- Schedule the replay of the response functions for all connected TPs according to the input scenario.
- Provide access to data and related services

# Device

$$O_i = f(I_1, \dots, I_n, \langle \text{config} \rangle)$$



# Device



# Trigger processors simulation

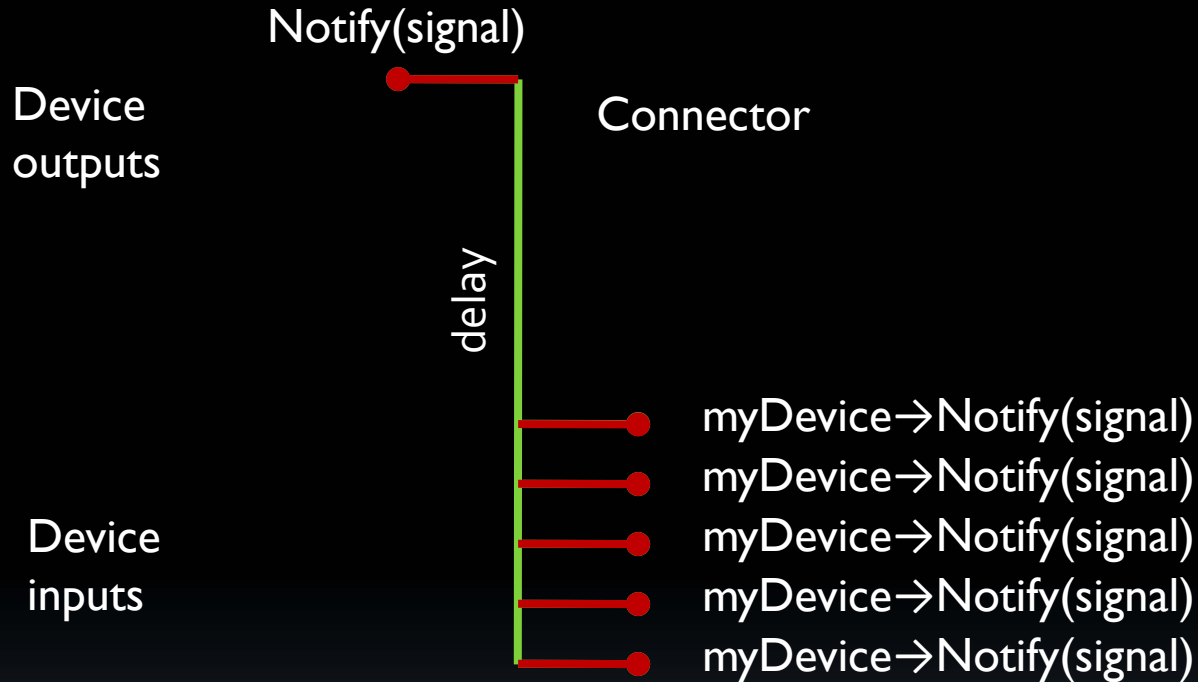
- Any trigger processor is a 'device'
  - Including CTP
- Device may be subdivided in smaller blocks
  - Such decomposition may not be important but providing such possibility may give extra flexibility in some cases
- Trigger experts should provide the response functions for their TP
  - Or port them from existing trigger detector class
- Devices are connected via 'slots'
  - Flow controlled by the framework

# Slots and connectors

- Slots are defined and belong to a device
  - Two types: input and output
- Slots receive or generate signals
  - When receiving signals, they notify the device they belong to (or keep the state until the device response fct. is called)
  - When generating signals, they propagate them to other devices via 'connectors'
- Slots are ignored unless plugged into a 'connector'
- A connector have a feeder slot and possibly a set of consumers
  - Current signal notifies all consumers
- Connectors may simulate also delays if needed



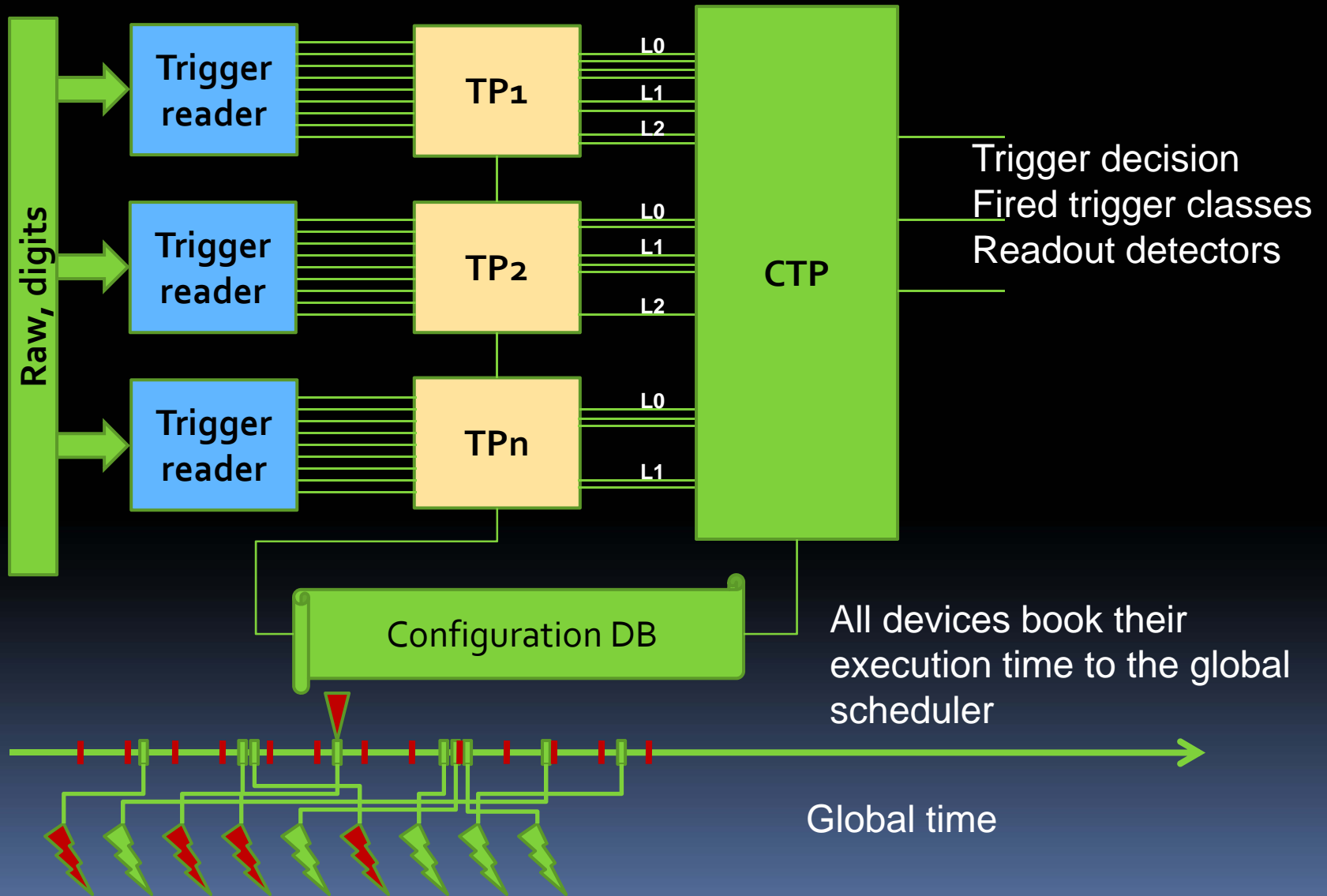
# Slots and connectors



# Signals


- Generated by device outputs, propagate the information between them
  - Name, type, start time
  - May embed data (messages)
- Once produced, signals are changing the state of the slot(s) they are being sent to.
  - A general RESET signal used to clear these states in the event loop
- Signals can be standalone, synchronized or grouped in sequences
  - A device is woke-up to compute its response functions only after the required input sequence is completed

# Flow





# Event scheduler

- Books execution of a device response function in a time slot
    - Advances the global time pointer to next slot and fires the booked signals to the connectors linked to the producer slot
  - Handle possible delays and busy time
- 



# To do

- Further clarify requirements and main use cases (with CTP & TP's)
  - Design and implement the steering class, event scheduler, base class for TP devices.
  - Convert the existing implementation of CTP to the new framework
  - Provide services for data access
- 