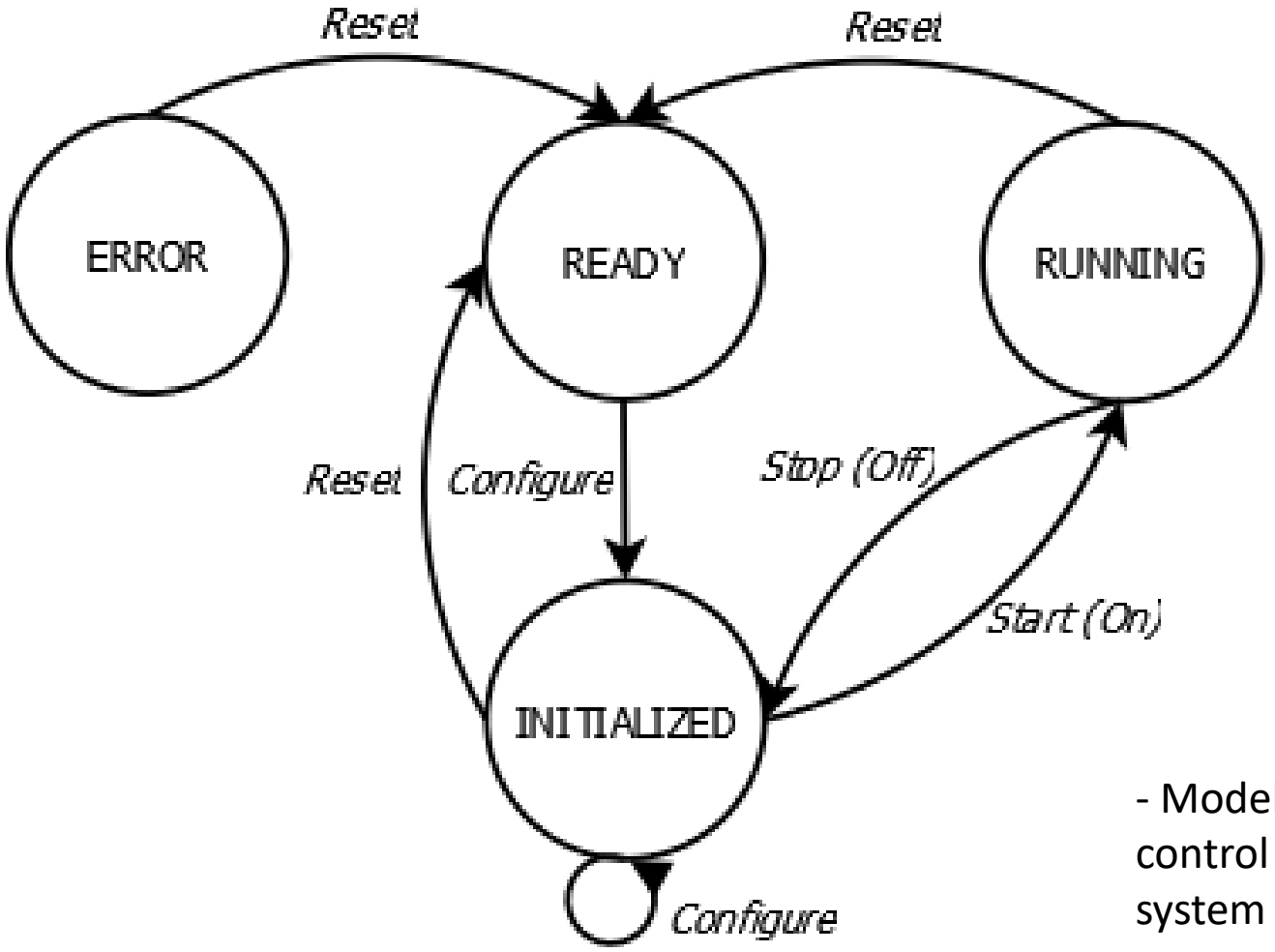
DCS configuration

DCS Finite state machine model



- Model implemented at the DCS control level by each PANDA sub-system

Device configuration (initialization) (I)

- Sub-detector configuration data represents the collection of all preset values existing on every single online device when the device is in INITIALIZED state:
  - Voltage, currents, temperature, pressure;
  - ASIC register values;
  - Etc.
- Preferably the configuration data for each device should reside in the device non-volatile memory.
- Once applied, the configuration data should not change (except reset) until next configuration cycle.

Device configuration (initialization) (II)

- Use Extensible Markup Language (XML) format for the configuration files

Advantages: - plain text: understandable and easy to modify;
- self-explainable and extensible;
- plain-text data is software and hardware independent;
- deserialization tools available for many programming languages.

(Deserialization: conversion of xml data into structured data (array, structure, classes,..))

In the case of PANDA Controls the importing of process variables into database engines is done via xml configuration files => the same method can be applied to configure DCS Field devices

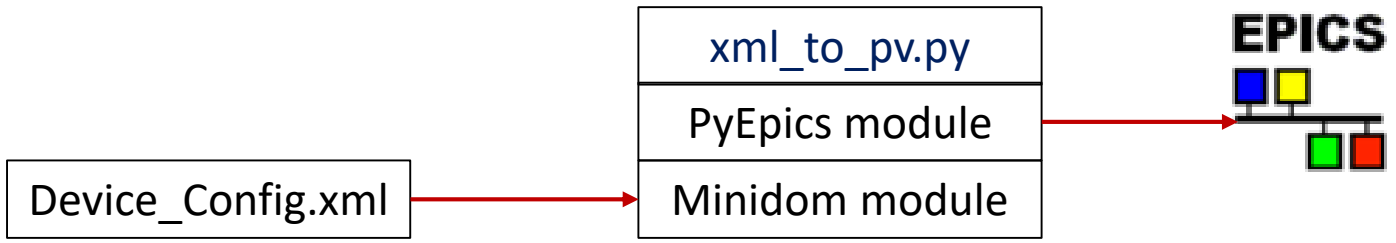
Device configuration implementation (I)

Example: xml configuration file for STT High voltage preset values

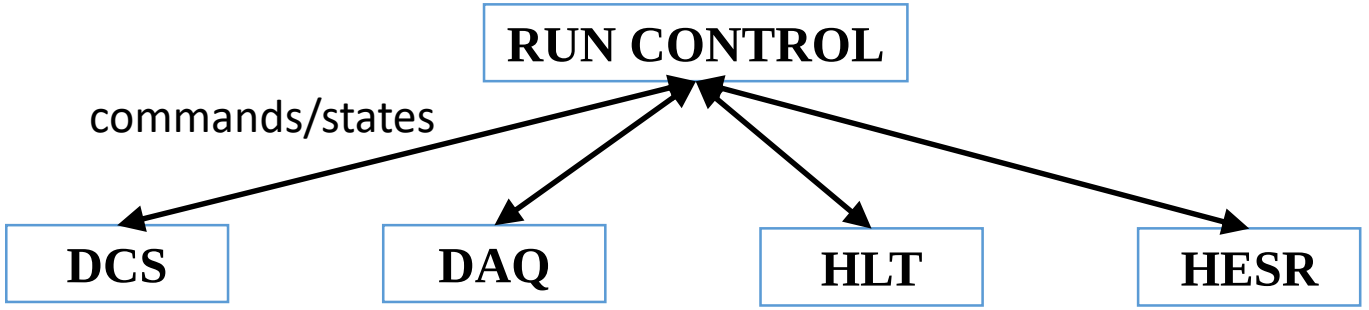
```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE document SYSTEM "channel.dtd">

<sub_system Name = "STT">
   <out_channel>
      <name>SET_HV_000</name>
      <value>1799.00</value>
   </out_channel>
   <out_channel>
      <name>SET_HV_001</name>
      <value>1799.50</value>
   </out_channel>
</sub_system>
```

Device configuration implementation (II)

- For each configuration xml is very much convenient to define a corresponding Document Type Definition (DTD);
- Before Initializing the device a verification of xml syntax should be performed;
- In linux **xmllint** program can be used to parse the xml and check the syntax;
- If xmllint returns 0 error code >>> a python script is used to deserialize the xml data and update the Epics Process Variables (channels)



PANDA Experiment (Run) Control



- EPICS environment;
- Common FSM model for the systems;
- Concept to be described in the Controls TDR;
- Implementation will be given later (after approval of TDRs) in a joint PANDA Internal Note.