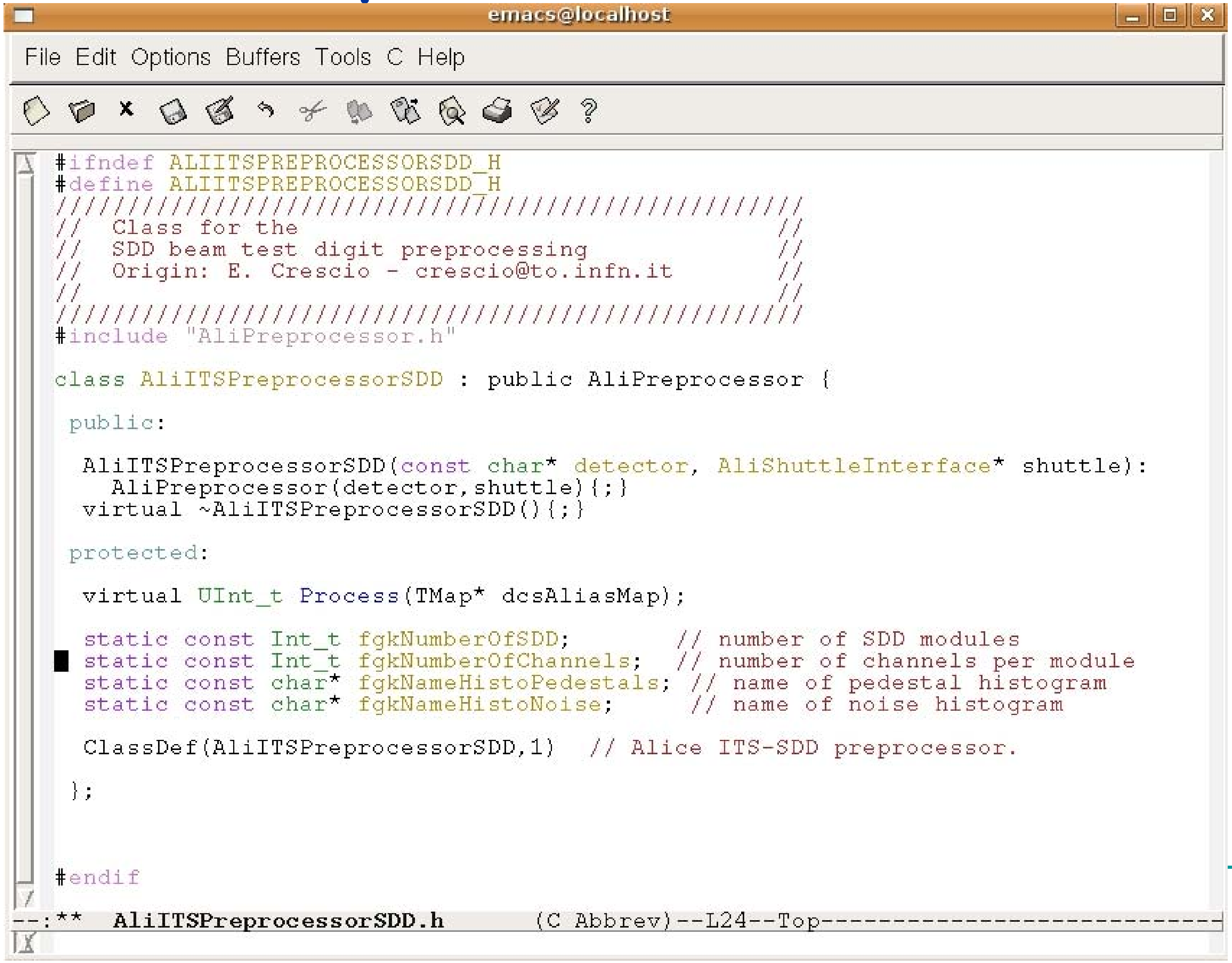
SDD: status of the calibration framework

Elisabetta Crescio, Massimo Masera
INFN & Torino University

SDD Preprocessor (1)

- First prototype of “SHUTTLE” preprocessor for SDD (AliITSPreprocessorSDD) ready
  - The method “Process” reads the calibration data files through SHUTTLE and store them in the OCDB in the proper format (AliITSCalibrationSDD classes)
- Tested with “dummy” calibration files using AliTestShuttle
  - Access to the data: OK
  - Storing to CDB: OK
- Committed (HEAD)

SDD Preprocessor (2)



```
#ifndef ALIITSPREPROCESSORSDD_H
#define ALIITSPREPROCESSORSDD_H
////////////////////////////////////////////////////
//  Class for the                                 //
//  SDD beam test digit preprocessing             //
//  Origin: E. Crescio - crescio@to.infn.it       //
//                                                //
////////////////////////////////////////////////////
#include "AliPreprocessor.h"

class AliITSPreprocessorSDD : public AliPreprocessor {

 public:

  AliITSPreprocessorSDD(const char* detector, AliShuttleInterface* shuttle):
    AliPreprocessor(detector,shuttle){;}
  virtual ~AliITSPreprocessorSDD(){;}

 protected:

  virtual UInt_t Process(TMap* dcsAliasMap);

  static const Int_t fgkNumberOfSDD;       // number of SDD modules
  static const Int_t fgkNumberOfChannels;  // number of channels per module
  static const char* fgkNameHistoPedestals; // name of pedestal histogram
  static const char* fgkNameHistoNoise;     // name of noise histogram

  ClassDef(AliITSPreprocessorSDD,1)  // Alice ITS-SDD preprocessor.

 };


#endif
--:**  AliITSPreprocessorSDD.h      (C Abbrev)--L24--Top----------------------------
```

Online calibration

- All the preprocessing algorithms will run on the online machines
- The SHUTTLE preprocessor will simply store the data in the proper format in the OCDB
- Online preprocessing algorithms implemented for laser test of the SDD modules exist (pedestals, noise,gain, bad channels..)
  - They have to be adapted to the DAQ framework and packages available
  - Take into account different rawdata format
- Still under development: preprocessing algorithm for injector's data

Calibration requirement tables

Parameter	Data format/ size per channel	Data size (Total)		Update freq	Source	Confirmed	Run type / Trigger type	# of required events/ sampling rate	Processing level: sub-event or event	Results: FEE/ Archive	Accessible by offline	Calib. Procedure in AliRoot (see Note 3)	use case #
		in OCDB	reference										
Low and HighThresholds for CARLOS	2x8 bits/ sensor	5,08E+02	not needed	Run	DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	------		Yes	No	5 / Note 1
Baselines	9 bits/ each PASCAL + 6 bits/ anode on AMBRA	9,98E+04	not needed	Run	DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	-----		Yes	No (Base classes)	5 / Note 1
Noisy anodes (masked)	512 bitmap: 512 bits/ sensor		not needed		DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	-----				5 / Note 1
Noise amplitude	4 bits/ anode	6,50E+04	not needed	Run	DAQ (data without 0 suppression - dedicated runs)	yes	Dedicated run w/ o 0 supp.	~1000	sub-event		Yes	No (Base classes)	1
Pedestals	8 bit/ anode	1,33E+05	not needed		DAQ (data without 0 suppression - dedicated runs)	yes	Dedicated run w/ o 0 supp.	~1000	sub-event				1
Gains	8 bits/ anode	1,33E+05	not needed	Run	DAQ (dedicated runs with S/ W triggers)	yes	Dedicated runs / SW trigger	~1000	sub-event		Yes	No (Base classes)	1
Drift velocity	1 word/ anode	1,07E+06	not needed	Run	DAQ (dedicated runs with S/ W triggers during standard running)	yes	Standard runs / special trigger (injectors)	every 10-15 minutes	sub-event		Yes	No (Base classes)	1
Dopant concentration map		3,00E+07	not needed	Once	DCDB	yes					Yes	No	Note 2
External temperature monitor	2x sensors +1 x CARLOS	1,46E+02	needed	Run	DCS	yes	Stabdard runs				No	No	4
10--->8 bit compression		0,125	not needed	Run	DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	-----				5 / Note 1
ADC readout clock (20/ 40 MHz)		0,125	not needed	Run	DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	-----				5 / Note 1
Analogue memory sampling frequencyv (20/ 40 MHz)		0,125	not needed	Run	DAQ (JTAG - It can be written to a run header)	yes	Standard runs	-----	-----				5 / Note 1
		3,14E+07											