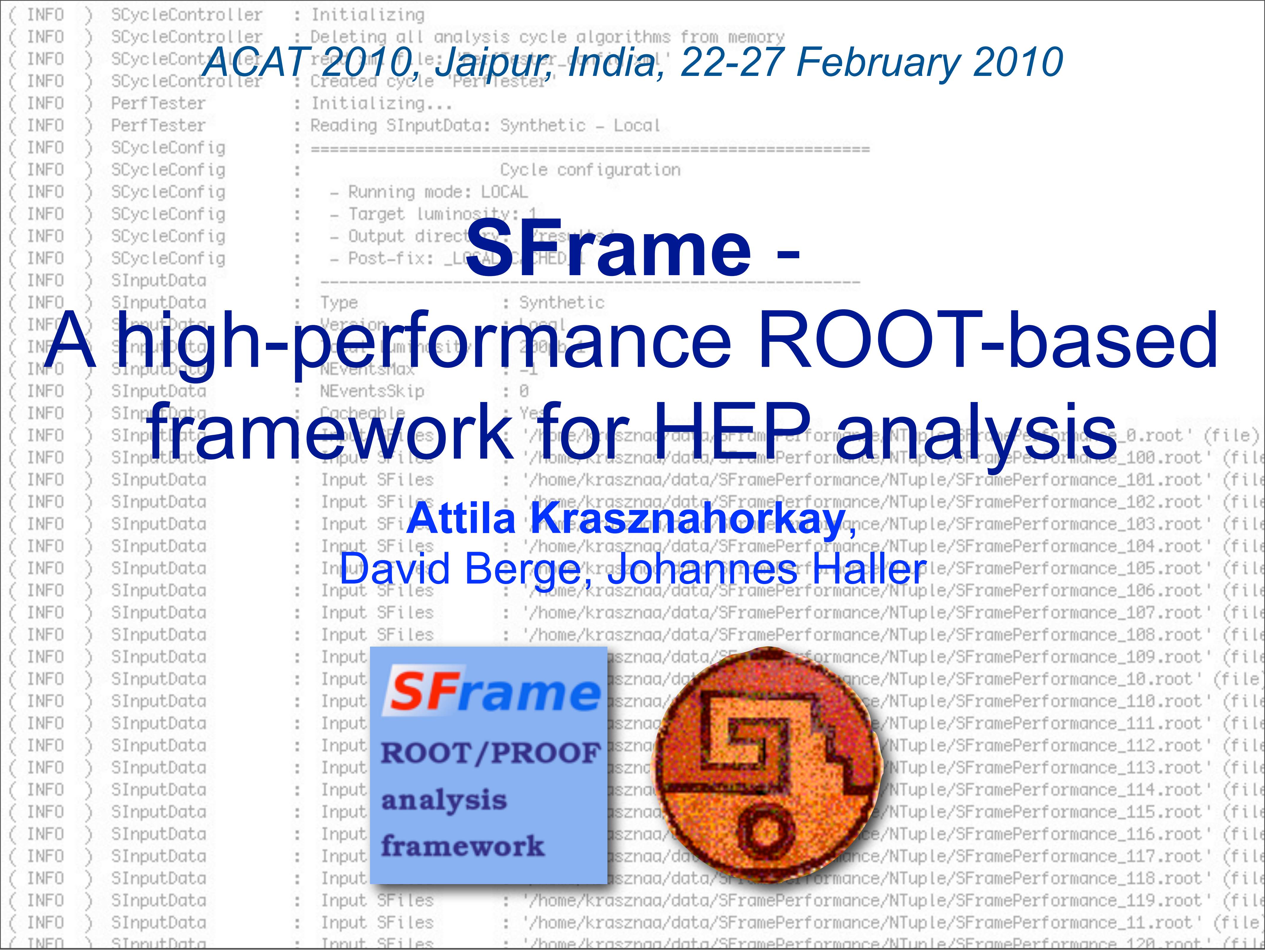
ACAT 2010, Jaipur, India, 22-27 February 2010

SFrame - A high-performance ROOT-based framework for HEP analysis

Attila Krasznahorkay,
David Berge, Johannes Haller



```
( INFO  )  SCycleController   : Initializing
( INFO  )  SCycleController   : Deleting all analysis cycle algorithms from memory
( INFO  )  SCycleCont[illegible]
( INFO  )  SCycleController   : Created cycle 'PerfTester'
( INFO  )  PerfTester         : Initializing...
( INFO  )  PerfTester         : Reading SInputData: Synthetic - Local
( INFO  )  SCycleConfig       : ==========================================================
( INFO  )  SCycleConfig       :                     Cycle configuration
( INFO  )  SCycleConfig       :    - Running mode: LOCAL
( INFO  )  SCycleConfig       :    - Target luminosity: 1
( INFO  )  SCycleConfig       :    - Output direct[illegible]
( INFO  )  SCycleConfig       :    - Post-fix: _L[illegible]
( INFO  )  SInputData         :   -------------------------------------------------------
( INFO  )  SInputData         :   Type                : Synthetic
( INFO  )  SInputData         :   Version             : Local
( INFO  )  SInputData         :   [illegible]
( INFO  )  SInputData         :   NEventsMax          : -1
( INFO  )  SInputData         :   NEventsSkip         : 0
( INFO  )  SInputData         :   Cacheable           : Yes
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa/data/[illegible]formance/NTuple/SFramePerformance_109.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa/da[illegible]nce/NTuple/SFramePerformance_10.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa[illegible]e/NTuple/SFramePerformance_110.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa[illegible]e/NTuple/SFramePerformance_111.root' (file
( INFO  )  SInputData         :   Input [illegible]NTuple/SFramePerformance_112.root' (file
( INFO  )  SInputData         :   Input [illegible]NTuple/SFramePerformance_113.root' (file
( INFO  )  SInputData         :   Input [illegible]NTuple/SFramePerformance_114.root' (file
( INFO  )  SInputData         :   Input [illegible]/NTuple/SFramePerformance_115.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa/[illegible]ce/NTuple/SFramePerformance_116.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa/da[illegible]ance/NTuple/SFramePerformance_117.root' (file
( INFO  )  SInputData         :   Input [illegible]sznaa/data/S[illegible]formance/NTuple/SFramePerformance_118.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file
( INFO  )  SInputData         :   Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (file
```

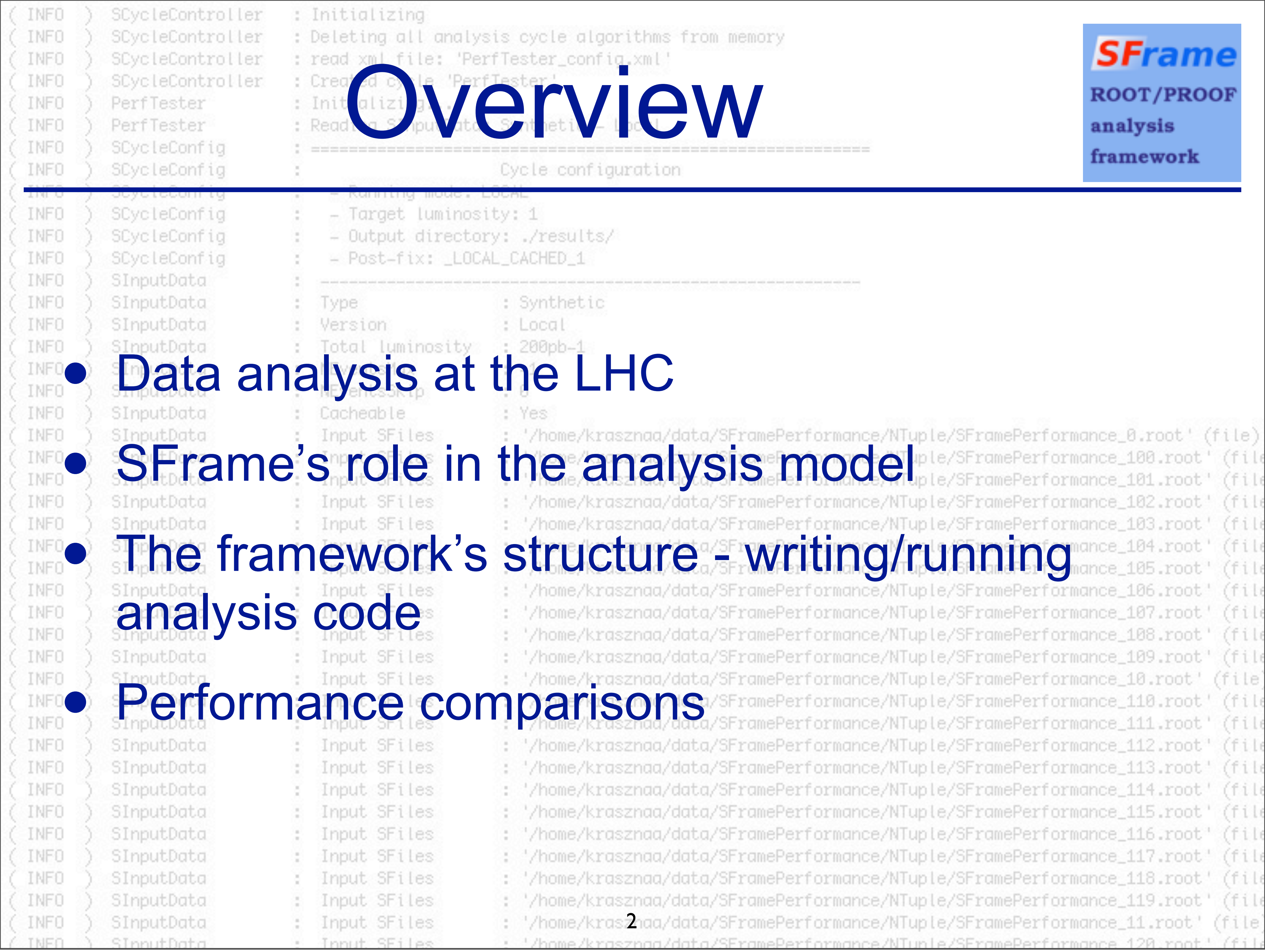
SFrame

ROOT/PROOF analysis framework

Overview

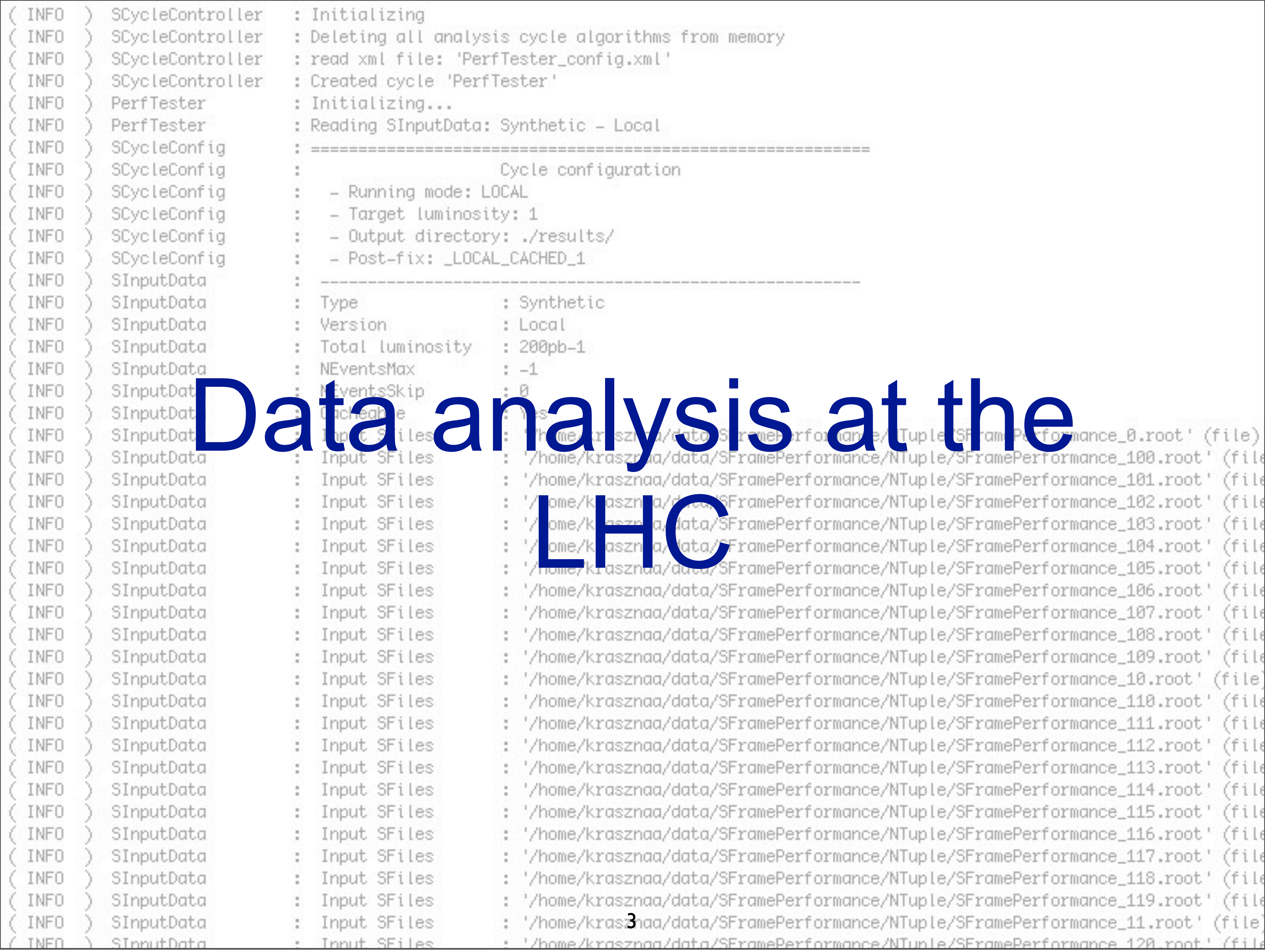
- Data analysis at the LHC
- SFrame's role in the analysis model
- The framework's structure - writing/running analysis code
- Performance comparisons

SFrame ROOT/PROOF analysis framework



```
( INFO  ) SCycleController    : Initializing
( INFO  ) SCycleController    : Deleting all analysis cycle algorithms from memory
( INFO  ) SCycleController    : read xml file: 'PerfTester_config.xml'
( INFO  ) SCycleController    : Created cycle 'PerfTester'
( INFO  ) PerfTester          : Initializing ...
( INFO  ) PerfTester          : Reading SInputData: Synthetic - Local
( INFO  ) SCycleConfig        : ================================================================
( INFO  ) SCycleConfig        :                          Cycle configuration
( INFO  ) SCycleConfig        :   - Running mode: LOCAL
( INFO  ) SCycleConfig        :   - Target luminosity: 1
( INFO  ) SCycleConfig        :   - Output directory: ./results/
( INFO  ) SCycleConfig        :   - Post-fix: _LOCAL_CACHED_1
( INFO  ) SInputData          :  ---------------------------------------------------------------
( INFO  ) SInputData          :  Type                   : Synthetic
( INFO  ) SInputData          :  Version                : Local
( INFO  ) SInputData          :  Total luminosity       : 200pb-1
( INFO  ) SInputData          :  NEventsMax             : -1
( INFO  ) SInputData          :  NEventsSkip            : 0
( INFO  ) SInputData          :  Cacheable              : Yes
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_109.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_10.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_110.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_111.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_112.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_113.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_114.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_115.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_116.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_117.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_118.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file
( INFO  ) SInputData          :  Input SFiles           :  '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (fil
```

Data analysis at the LHC



```
( INFO  ) SCycleController   : Initializing
( INFO  ) SCycleController   : Deleting all analysis cycle algorithms from memory
( INFO  ) SCycleController   : read xml file: 'PerfTester_config.xml'
( INFO  ) SCycleController   : Created cycle 'PerfTester'
( INFO  ) PerfTester         : Initializing...
( INFO  ) PerfTester         : Reading SInputData: Synthetic - Local
( INFO  ) SCycleConfig       : ==============================================================
( INFO  ) SCycleConfig       :                     Cycle configuration
( INFO  ) SCycleConfig       :   - Running mode: LOCAL
( INFO  ) SCycleConfig       :   - Target luminosity: 1
( INFO  ) SCycleConfig       :   - Output directory: ./results/
( INFO  ) SCycleConfig       :   - Post-fix: _LOCAL_CACHED_1
( INFO  ) SInputData         :  ------------------------------------------------------------
( INFO  ) SInputData         :  Type                 : Synthetic
( INFO  ) SInputData         :  Version              : Local
( INFO  ) SInputData         :  Total luminosity     : 200pb-1
( INFO  ) SInputData         :  NEventsMax           : -1
( INFO  ) SInputData         :  NEventsSkip          : 0
( INFO  ) SInputData         :  Cacheable            : Yes
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_109.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_10.root' (file)
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_110.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_111.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_112.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_113.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_114.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_115.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_116.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_117.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_118.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file
( INFO  ) SInputData         :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (fil
```

LHC's data processing

SFrame
ROOT/PROOF analysis framework

- ~300 MB/s from the experiments during data taking
- Prompt reconstruction and data distribution -> Reconstructed files get copied worldwide
- Reco. files are processed on the LHC GRID
- Turn-around times are on the order of 1 day (when lucky) -> Not adequate for developing/ tuning an analysis
- -> Create skimmed data samples that can be analysed on local clusters

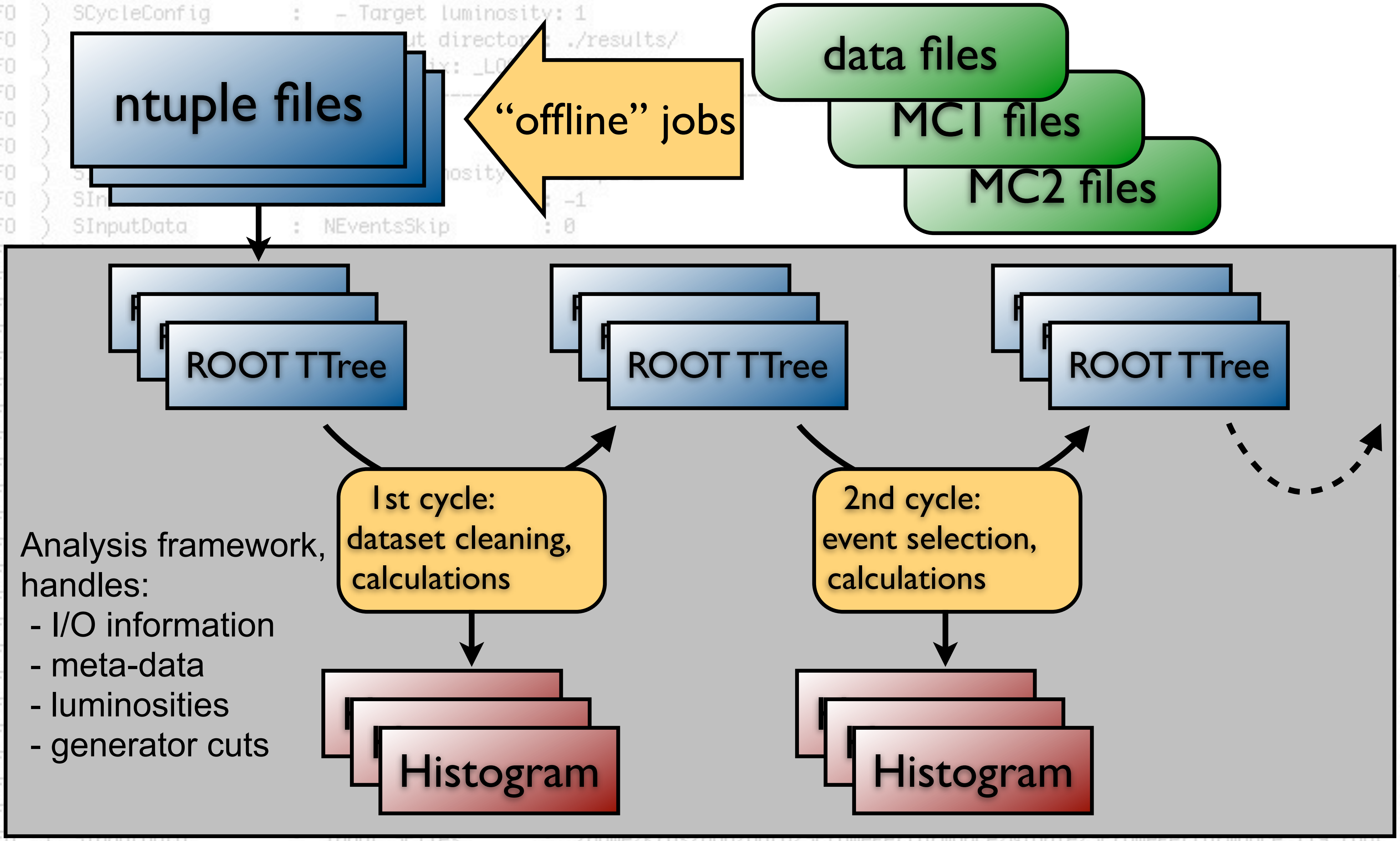
Requirements

SFrame
ROOT/PROOF analysis framework

- Read data from ROOT ntuples, and create event level output data (ROOT TTree), result histograms or other objects.
- Has to be easy to develop/debug code locally, and then be able to send the analysis to a cluster of machines
- The framework has to provide some functionality often needed in HEP analyses
- Should be easily configurable (without having to re-compile the user code)
- The code has to be as fast as possible

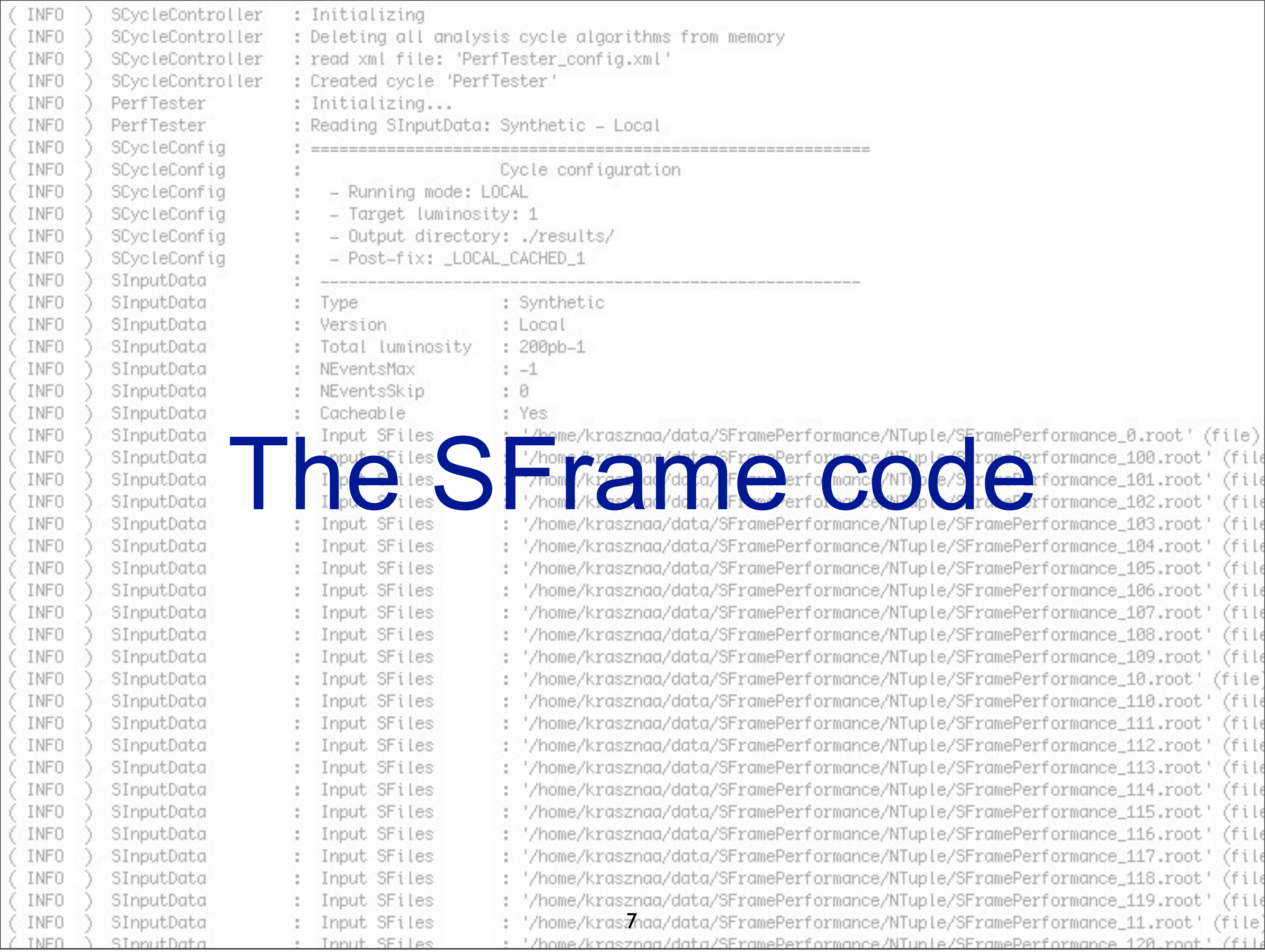
Analysis flow

SFrame
ROOT/PROOF analysis framework



Analysis framework, handles:
- I/O information
- meta-data
- luminosities
- generator cuts

The SFrame code



```
( INFO  )  SCycleController   : Initializing
( INFO  )  SCycleController   : Deleting all analysis cycle algorithms from memory
( INFO  )  SCycleController   : read xml file: 'PerfTester_config.xml'
( INFO  )  SCycleController   : Created cycle 'PerfTester'
( INFO  )  PerfTester         : Initializing...
( INFO  )  PerfTester         : Reading SInputData: Synthetic - Local
( INFO  )  SCycleConfig       : ==============================================================
( INFO  )  SCycleConfig       :                      Cycle configuration
( INFO  )  SCycleConfig       :    - Running mode: LOCAL
( INFO  )  SCycleConfig       :    - Target luminosity: 1
( INFO  )  SCycleConfig       :    - Output directory: ./results/
( INFO  )  SCycleConfig       :    - Post-fix: _LOCAL_CACHED_1
( INFO  )  SInputData         :  ------------------------------------------------------------
( INFO  )  SInputData         :  Type                : Synthetic
( INFO  )  SInputData         :  Version             : Local
( INFO  )  SInputData         :  Total luminosity    : 200pb-1
( INFO  )  SInputData         :  NEventsMax          : -1
( INFO  )  SInputData         :  NEventsSkip         : 0
( INFO  )  SInputData         :  Cacheable           : Yes
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_109.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_10.root' (file)
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_110.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_111.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_112.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_113.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_114.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_115.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_116.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_117.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_118.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (file
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file)
( INFO  )  SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (file
```

SFrame

SFrame
ROOT/PROOF
analysis
framework

- Started out as a framework for ATLAS analyses done outside of the ATLAS offline software
- Lives on SourceForge: http://sourceforge.net/projects/sframe/
- Most of the documentation is on the SF Wiki: http://sourceforge.net/apps/mediawiki/sframe/

SFrame features (1)

SFrame
ROOT/PROOF analysis framework

- If the user follows some simple rules on writing an analysis package, the package:
  - Compiles a shared library
  - Creates a .par file that can be used by PROOF
- This library and .par package can then be declared in an XML configuration file so that SFrame uses them
- Makes code sharing within a group very simple -> Can share “analysis cycles” for common tasks (dataset cleaning, etc.)

SFrame features (2)

SFrame
ROOT/PROOF
analysis
framework

- Arranges input files into InputData blocks (blocks that should be handled homogeneously)
- Keeps track of the integrated luminosity of the InputData blocks
- Provides a way of scaling the Monte Carlo to the data's integrated luminosity
- Easy-to-use functions for reading/writing TTrees
- Simple interface for writing various TObjects into the output file.
- Some additional classes, providing convenience functionalities and speed increases

SFrame code structure

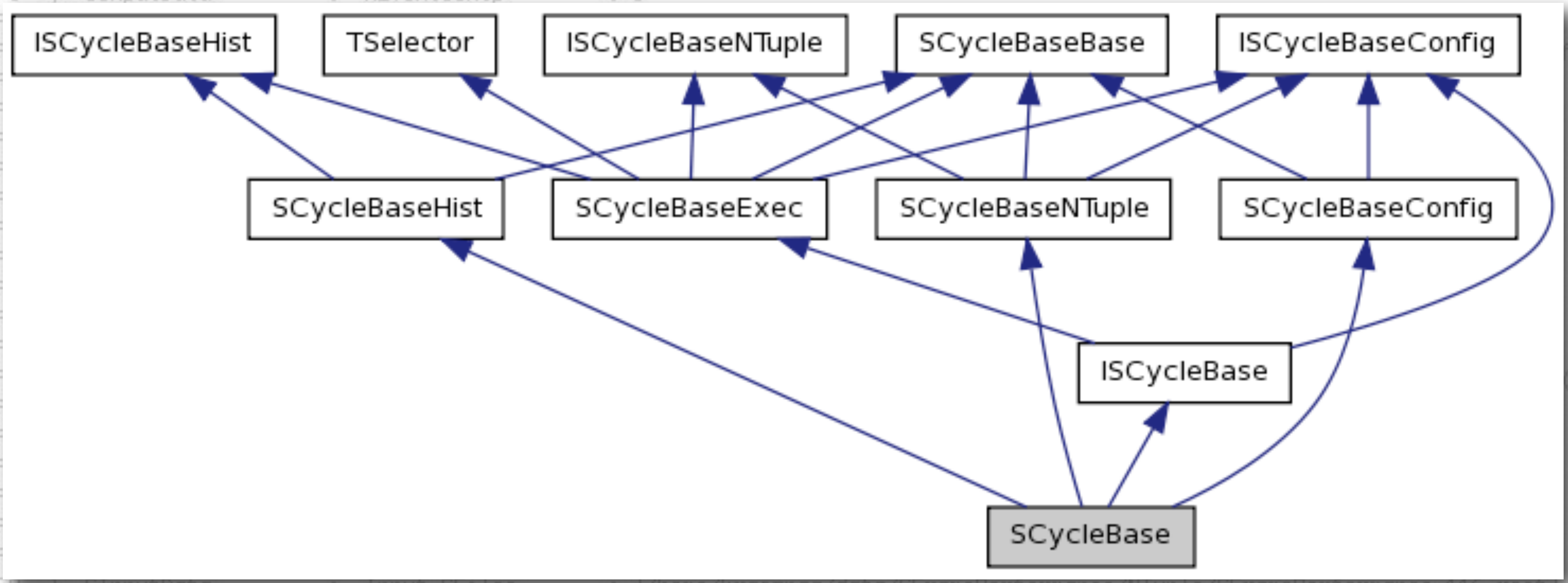
SFrame ROOT/PROOF analysis framework

- Compiles 3 shared libraries by default
  - **`libSFrameCore`**: The core classes that all SFrame jobs need
  - **`libSFramePlugIns`**: Classes providing convenience functionality
  - **`libSFrameUser`**: Library with some user code examples
- Compiles an executable (**`sframe_main`**) which is only linked against **`libSFrameCore`**. Other libraries -> loaded at runtime.

SCycleBase (1)

SFrame
ROOT/PROOF analysis framework

- The user writes classes inheriting from SCycleBase. Example can be found here.



SCycleBase (2)

SFrame
ROOT/PROOF analysis framework

- The base class is highly modular -> It is possible to create new base classes with extended capabilities.
  - Reading something else than ROOT TTree-s as input (like HepMC events for instance)
  - Writing new kinds of outputs
- The framework executes cycles through the **`ISCycleBase`** interface. As long as the new base class implements this interface, it works with the framework. -> Can overwrite functions from **`SCycleBaseExec`**.

Creating a package/cycle

SFrame
ROOT/PROOF
analysis
framework

- With only a few commands one can arrive at compilable code, starting from scratch.

```
# cd a_good_directory_for_the_analysis_code/
# svn co https://sframe.svn.sourceforge.net/svnroot/sframe/SFrame/trunk SFrame
# cd SFrame
# source ./setup.[c]sh
# make
# cd ../
# sframe_new_package.sh MyAnalysis
# cd MyAnalysis/
# sframe_create_cycle.py --name My::AnalysisCycle \
    --linkdef include/SFrameMyAnalysis_LinkDef.h
```

Edit the created AnalysisCycle.h and AnalysisCycle.cxx code skeletons, adding your analysis code.

```
# cd MyAnalysis/
# make
```

A simple cycle (1)

SFrame
ROOT/PROOF
analysis
framework

```
namespace My {
  class AnalysisCycle : public SCycleBase {
  public:
    AnalysisCycle();

    virtual void BeginCycle() throw( SError );
    virtual void EndCycle() throw( SError );

    virtual void BeginInputData( const SInputData& id ) throw( SError );
    virtual void EndInputData( const SInputData& id ) throw( SError );

    virtual void BeginInputFile( const SInputData& id ) throw( SError );

    virtual void ExecuteEvent( const SInputData& id, Double_t weight ) throw( SError );

  private:
    std::vector< float >* m_inputVar1;
    std::vector< float >* m_inputVar2;
    std::vector< double > m_outputVar1;
    std::vector< double > m_outputVar2;

    ClassDef( My::AnalysisCycle, 0 )
  };
}
```

Inherits from common base class

Functions declared in the base class

Pointers connected to the input variables

Objects to be written to the output

Declare the class to ROOT

A simple cycle (2)

SFrame
ROOT/PROOF analysis framework

```
void AnalysisCycle::BeginInputData( const SInputData& ) throw( SError ) {
  // Create histograms in the output file:
  Book( TH1F( “FirstHistogram”, “First histogram”, 100, 0.0, 100.0 ) );
  Book( TH1F( “SecondHistogram”, “Second histogram”, 50, 0.0, 50.0 ), “control” );
  // Put the variables in the output TTree:
  DeclareVariable( m_outputVar1, “OutputVar1” );
  DeclareVariable( m_outputVar2, “OutputVar2” );
}

void AnalysisCycle::BeginInputFile( const SInputData& ) throw( SError ) {
  // Connect the pointers to the input variables:
  ConnectVariable( “CollectionTree”, “InputVar1”, m_inputVar1 );
  ConnectVariable( “CollectionTree”, “InputVar2”, m_inputVar2 );
}

void AnalysisCycle::ExecuteEvent( const SInputData&, Double_t weight ) throw( SError ) {
  m_outputVar1.clear(); m_outputVar2.clear(); // Reset the output variables
  // Do an event selection. Events failing the selection will not be written out:
  if( m_inputVar1->size() < 2 ) throw SError( SError::SkipEvent );
  // Fill the previously booked histograms:
  Hist( “FirstHistogram” )->Fill( ( *m_inputVar1 )[ 0 ], weight );
  Hist( “SecondHistogram”, “control” )->Fill( ( *m_inputVar1 )[ 1 ], weight );

  // Now fill the output variables using some code
}
```

Histogram put into “control” directory in output

A full job configuration

SFrame
ROOT/PROOF analysis framework

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE JobConfiguration PUBLIC "" "JobConfig.dtd">

<JobConfiguration JobName="MyAnalysisJob" OutputLevel="INFO">

   <Library Name="libSFrameMyAnalysis" />
   <Package Name="SFrameCore.par" />
   <Package Name="SFrameMyAnalysis.par" />

   <Cycle Name="My::AnalysisCycle" OutputDirectory="./results/" PostFix="_test"
          TargetLumi="123.4" RunMode="PROOF" ProofServer="lite" ProofWorkDir="" >

      <InputData Type=”MC” Version=”MyPhysicsProcess” NEventsMax=”-1” NEventsSkip=”0” >
         <In FileName=”MyInputFile.root” Lumi=”43.2” />
         <InputTree Name=”CollectionTree” />
         <OutputTree Name=”SFrameTree” />
      </InputData>

      <UserConfig>
         <Item Name="InputTreeName" Value="CollectionTree" />
         <Item Name=”OutputTreeName” Value=”SFrameTree” />
      </UserConfig>
   </Cycle>
</JobConfiguration>
```

Libraries and packages used by the job

Declaration of what should run and where

Declaration of inputs and outputs

Settings of the user properties

Handling different inputs

SFrame ROOT/PROOF analysis framework

```
<InputData Type="data" Version="run123456" NEventsMax="-1" NEventsSkip="0" >
   <In FileName="Data_run123456_001.root" Lumi="5.5" />
   <In FileName="Data_run123456_002.root" Lumi="2.3" />
   <In FileName="Data_run123456_003.root" Lumi="6.4" />
   <InputTree Name="CollectionTree" />
   <OutputTree Name="SFrameTree" />
</InputData>

<InputData Type="MC" Version="ttbar" NEventsMax="-1" NEventsSkip="0" >
   <In FileName="TTbar_001.root" Lumi="100.0" />
   <In FileName="TTbar_002.root" Lumi="100.0" />
   <In FileName="TTbar_003.root" Lumi="100.0" />
   <InputTree Name="CollectionTree" />
   <OutputTree Name="SFrameTree" />
</InputData>

<InputData Type="MC" Version="Zee" NEventsMax="-1" NEventsSkip="0" >
   <In FileName="Zee_001.root" Lumi="50.0" />
   <In FileName="Zee_002.root" Lumi="50.0" />
   <In FileName="Zee_003.root" Lumi="50.0" />
   <InputTree Name="CollectionTree" />
   <OutputTree Name="SFrameTree" />
</InputData>
```

Treated in a special way

If one sets the **`TargetLumi`** option to the sum of these, the MC gets weighted to the data

Performance comparisons

```
( INFO  ) SCycleController   : Initializing
( INFO  ) SCycleController   : Deleting all analysis cycle algorithms from memory
( INFO  ) SCycleController   : read xml file: 'PerfTester_config.xml'
( INFO  ) SCycleController   : Created cycle 'PerfTester'
( INFO  ) PerfTester         : Initializing...
( INFO  ) PerfTester         : Reading SInputData: Synthetic - Local
( INFO  ) SCycleConfig       : ============================================================
( INFO  ) SCycleConfig       :                     Cycle configuration
( INFO  ) SCycleConfig       :   - Running mode: LOCAL
( INFO  ) SCycleConfig       :   - Target luminosity: 1
( INFO  ) SCycleConfig       :   - Output directory: ./results/
( INFO  ) SCycleConfig       :   - Post-fix: _LOCAL_CACHED_1
( INFO  ) SInputData         :  -----------------------------------------------------------
( INFO  ) SInputData         :  Type                : Synthetic
( INFO  ) SInputData         :  Version             : Local
( INFO  ) SInputData         :  Total luminosity    : 200pb-1
( INFO  ) SInputData         :  NEventsMax          : -1
( INFO  ) SInputData         :  NEventsSkip         : 0
( INFO  ) SInputData         :  Cacheable           : Yes
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_109.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_10.root' (file)
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_110.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_111.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_112.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_113.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_114.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_115.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_116.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_117.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_118.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (file
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file)
( INFO  ) SInputData         :  Input SFiles        : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (file
```

The analysis

SFrame
ROOT/PROOF analysis framework

- Generated 10M events distributed in 200 files using the ATLAS offline software
- Each event contains 10 particles with flat distributions in $|\eta| < 3.0$, $5.0 < p_T < 50.0$ GeV, $-\pi < \Phi < \pi$
- Stored the generated events in ATLAS's MC event format
- Using the offline software, "translated" the events into "flat" ROOT TTree-s to serve as input to non-ATLAS analysis code.
- Wrote a simple but CPU intensive code analysing these events using multiple languages/frameworks

The analysis software

SFrame
ROOT/PROOF analysis framework

- **Athena:** An AthAlgorithm that does only the analysis tasks, in a minimalistic ATLAS offline job
- **ACLiC:** The analysis code put into a class created by TTree::MakeClass(...), compiled and run from ROOT
- **CINT:** The analysis put into a single function, and run from ROOT in interactive mode
- **PyROOT:** A stand-alone Python analysis script using the ROOT bindings
- **SFrame:** A cycle implementing this analysis
    - LOCAL mode: Running the cycle on one processor core
    - PROOF-Lite mode: Running the cycle on all 4 cores of one PC
    - PROOF mode: Running the cycle on all 8 cores of two PCs

Processing speeds

SFrame
ROOT/PROOF analysis framework

From 5 consecutive runnings.

Input location		Local	XRootD
Athena		1.77±0.02 kHz	N/A
ACLiC		3.85±0.03 kHz	3.77±0.04 kHz
CINT		259.0±2.2 Hz	N/A
PyROOT		127.2±2.2 Hz	123.1±1.6 Hz
SFrame	LOCAL	4.04±0.02 kHz	4.02±0.03 kHz
	PROOF-Lite	15.92±0.15 kHz	15.81±0.13 kHz
	PROOF	N/A	29.53±0.17 kHz

Comparison conclusions

SFrame ROOT/PROOF analysis framework

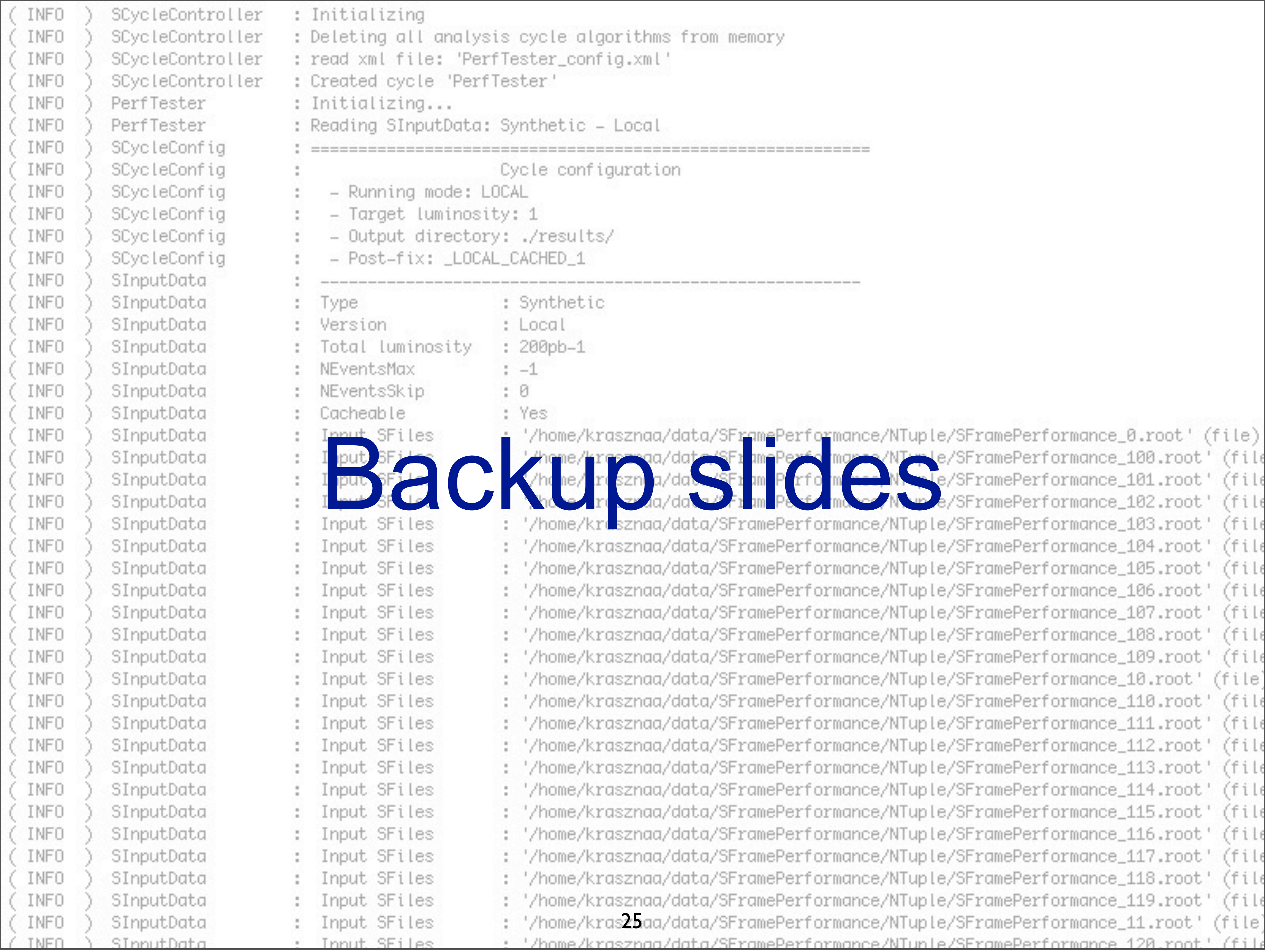
- ATLAS offline software performing very well with small MC events. -> Full reconstructed events with a real analysis perform usually ~100 Hz.
- Dedicated ACLiC code very fast. Speed increase in SFrame's "LITE mode" only due to different histogram handling. Realistic analyses -> O(1-2 kHz)
- The code was just too complicated for PyROOT to process quickly. However code development can be very quick.
- Same holds for interactive CINT, however coding is similar in difficulty to using ACLiC.
- As long as I/O is not a limiting factor, SFrame scales well with CPU count. (Relationship more complicated when writing TTree-s.) Realistic analyses do O(1-2 kHz) per CPU core.

Summary

SFrame
ROOT/PROOF analysis framework

- SFrame is used by many groups in ATLAS by now
- Offers good speed increase even over compiled ROOT code without much effort
- Can run the same code locally or using PROOF by just changing a configuration option
- Development continuing: Adding new convenience classes for using PROOF, integrating ATLAS file reading into the code (as an external library), and anything else that comes up while performing the first ATLAS analyses with real data.

Backup slides



```
( INFO  )  SCycleController    : Initializing
( INFO  )  SCycleController    : Deleting all analysis cycle algorithms from memory
( INFO  )  SCycleController    : read xml file: 'PerfTester_config.xml'
( INFO  )  SCycleController    : Created cycle 'PerfTester'
( INFO  )  PerfTester          : Initializing...
( INFO  )  PerfTester          : Reading SInputData: Synthetic - Local
( INFO  )  SCycleConfig        : ============================================================
( INFO  )  SCycleConfig        :                     Cycle configuration
( INFO  )  SCycleConfig        :   - Running mode: LOCAL
( INFO  )  SCycleConfig        :   - Target luminosity: 1
( INFO  )  SCycleConfig        :   - Output directory: ./results/
( INFO  )  SCycleConfig        :   - Post-fix: _LOCAL_CACHED_1
( INFO  )  SInputData          :  ----------------------------------------------------------
( INFO  )  SInputData          :  Type                 : Synthetic
( INFO  )  SInputData          :  Version              : Local
( INFO  )  SInputData          :  Total luminosity     : 200pb-1
( INFO  )  SInputData          :  NEventsMax           : -1
( INFO  )  SInputData          :  NEventsSkip          : 0
( INFO  )  SInputData          :  Cacheable            : Yes
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_0.root' (file)
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_100.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_101.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_102.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_103.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_104.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_105.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_106.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_107.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_108.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_109.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_10.root' (file
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_110.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_111.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_112.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_113.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_114.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_115.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_116.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_117.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_118.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_119.root' (fil
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_11.root' (file
( INFO  )  SInputData          :  Input SFiles         : '/home/krasznaa/data/SFramePerformance/NTuple/SFramePerformance_120.root' (fil
```

Analysis code used in the performance comparison

SFrame
ROOT/PROOF analysis framework

- Create TLorentzVector-s from the generated particles
- Plot the properties of the generated particles, including the total (transverse) energy of the particles
- Calculate the invariant mass of all 2, 3, 4 and 5 particle combinations
- Calculate the distance in pseudorapidity, azimuthal angle and ΔR between all 2-particle combinations
- Total number of created histograms: 14

Hardware used for the performance comparison

SFrame
ROOT/PROOF analysis framework

- 2 “standard” CERN PCs running SLC5 64-bit
  - Quad-core AMD Phenom™ 9600B CPU
  - 4/8 GB RAM
  - 160 GB local disk
- 4 TB LaCie 4big Quadra on one PC serving files over XRootD
- ROOT 5.22/00d
- Python 2.5
- ATLAS offline software 15.6.4