technische universität dortmund

FACT

SFB 876 Providing Information by Resource-Constrained Data Analysis

FACT-Tools

Processing High-Volume Telescope Data

Jens Buß,
Christian Bockermann,
Kai Brügge,
Maximilian Nöthe

for the FACT Collaboration

Federal Ministry of Education and Research

Outline

- FACT telescope and challenges
- Analysis requirements
- `Fact-Tools` and `Streams`
- Big Data scalability

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT - First G-APD Cherenkov Telescope

Type: IACT with 1440 Pixels

Location: La Palma, 2200m a.s.l.

First Light: October 2011

Specialty: Semiconductor based Pixels (SiPM)

Goal: Monitoring of bright TeV Blazars

FACT - First G-APD Cherenkov Telescope

Type: IACT with 1440 Pixels

Location: La Palma, 2200m a.s.l.

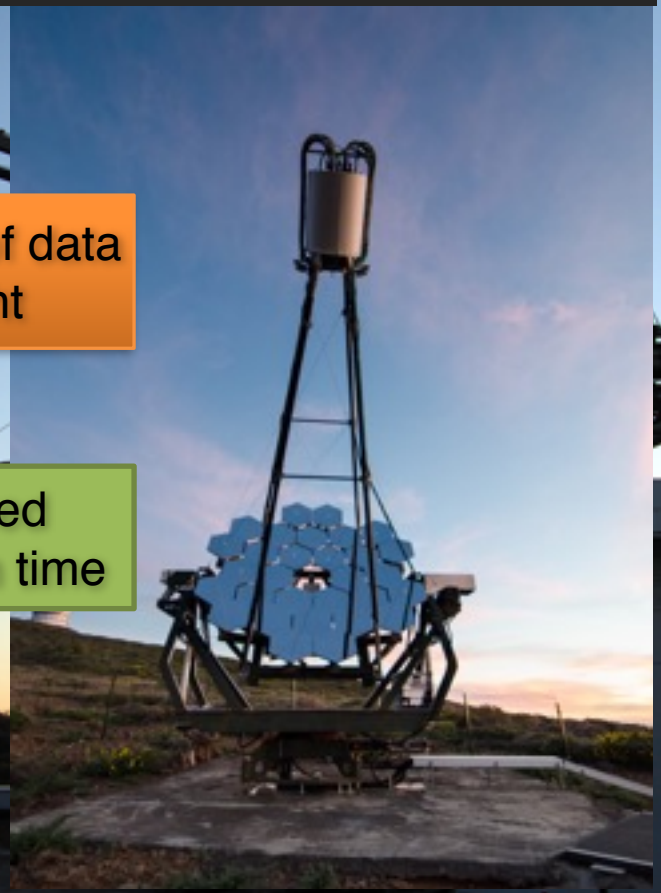
First Light: October 2011

Specialty: Semiconductor based Pixels (SiPM)

Goal: Monitoring of bright TeV Blazars

up to 1 TB of data per night

Maximized observation time



FACT - First G-APD Cherenkov Telescope

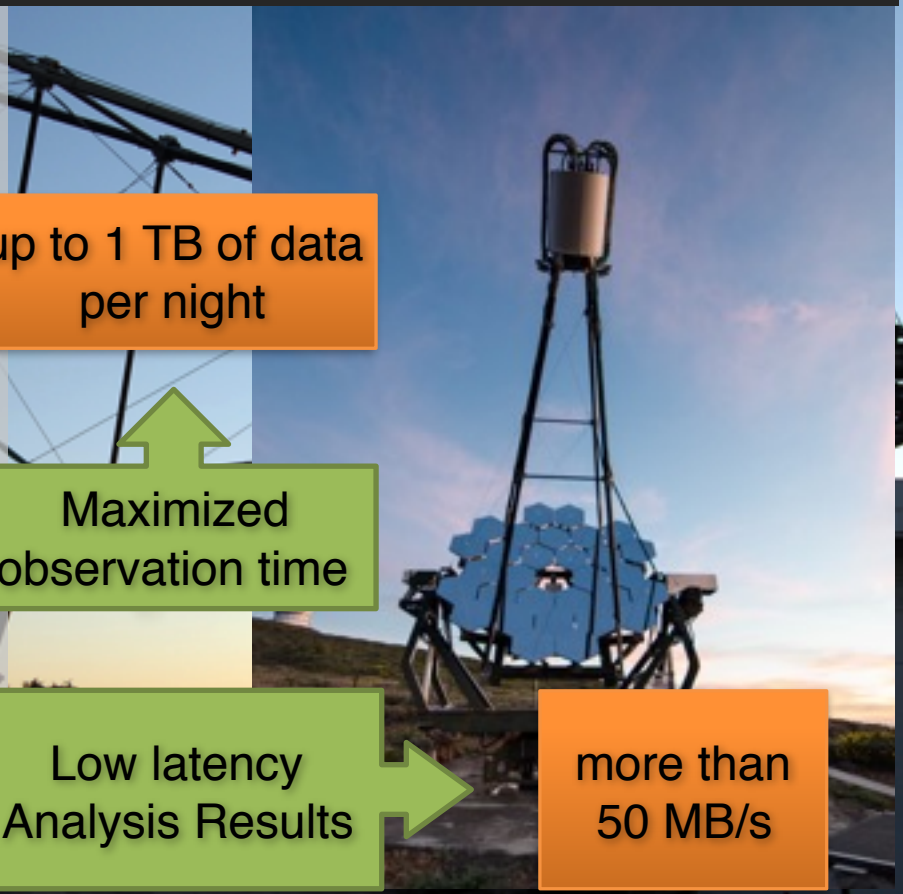
Type: IACT with 1440 Pixels

Location: La Palma, 2200m a.s.l.

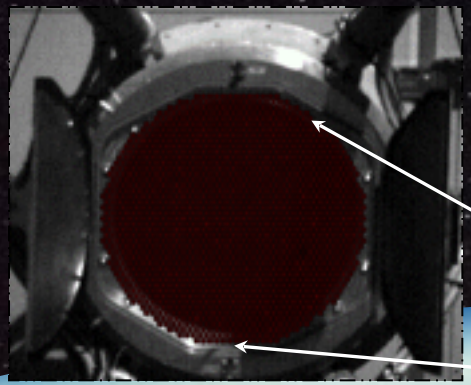
First Light: October 2011

Specialty: Semiconductor based Pixels (SiPM)

Goal: Monitoring of bright TeV Blazars



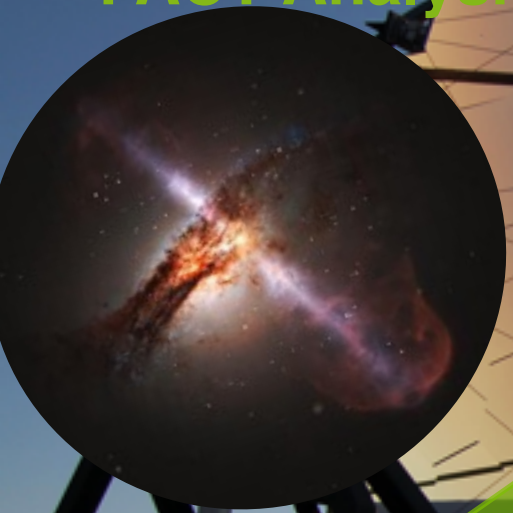
IACT Detection Principle

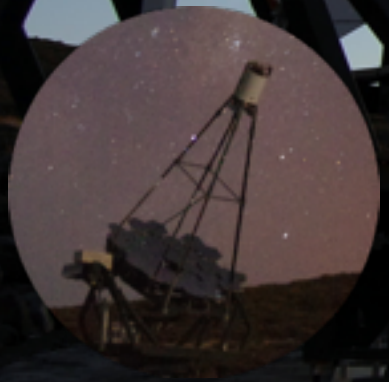


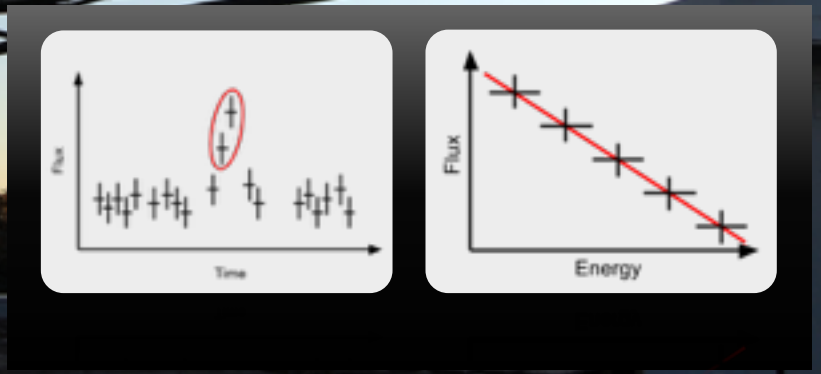
Gammas

Charged
Particles

FACT Analysis Chain

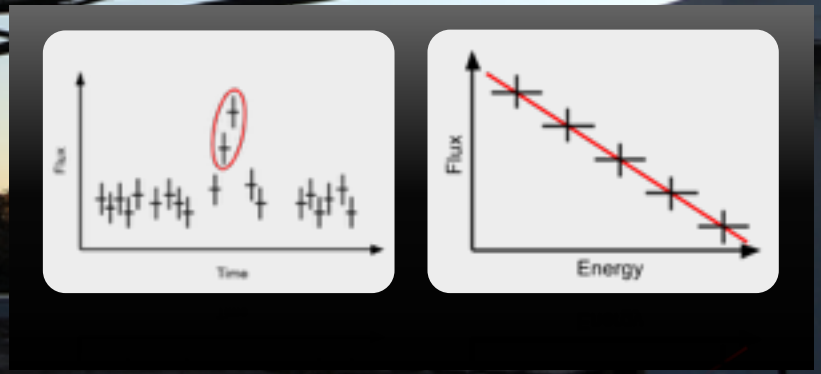






FACT Analysis Chain

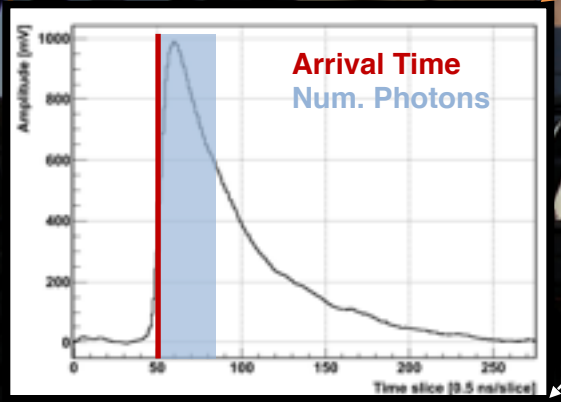
Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

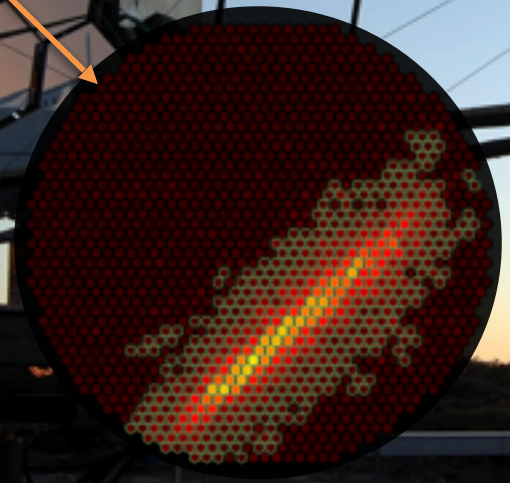
FACT Analysis Chain

Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

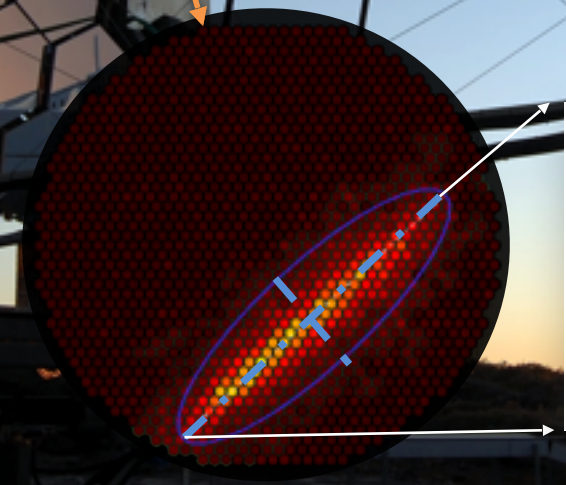
FACT Analysis Chain

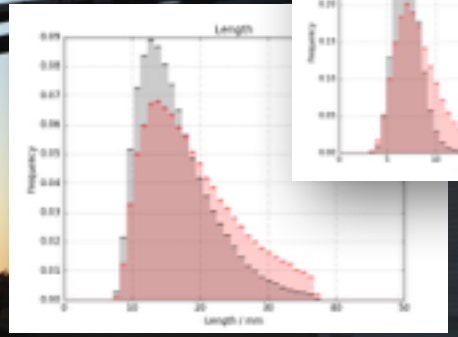


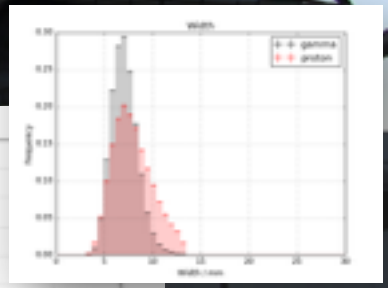
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT Analysis Chain

Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation







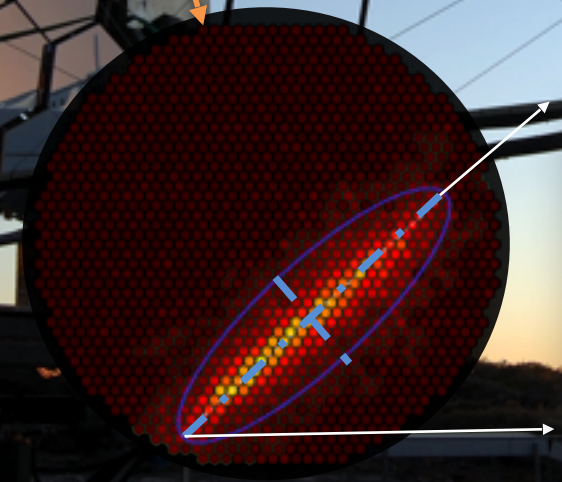
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

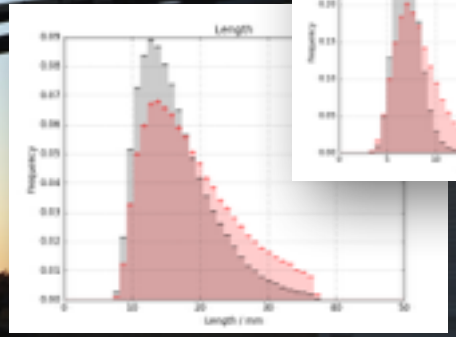
FACT Analysis Chain

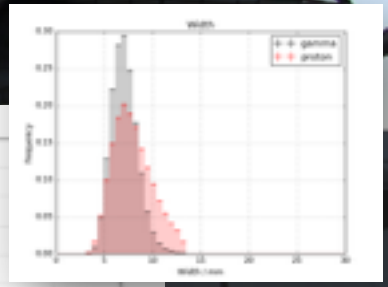
Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation

Feature Representation

ID	M	B	F1	F2	Z	#Isles	Width
1	3,54	4,93	7,53	2,99	6,78	4,52	5,64
2	4,05	0,62	4,13	1,41	3,13	1,40	1,98
3	3,68	8,88	4,23	5,05	4,78	7,75	8,48
4	6,75	4,58	0,48	3,79	6,37	3,81	1,76
5	7,59	3,68	5,52	1,23	5,76	6,70	1,10
6	0,25	3,77	7,06	5,63	0,25	9,02	5,51
7	5,29	3,44	7,93	6,00	4,05	0,08	8,35
8	8,34	1,71	4,63	7,07	0,63	7,71	4,84







Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT Analysis Chain

Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation

Feature Representation

ID	M	B	F1	F2	Z	#Isles	Width
1	3,54	4,93	7,53	2,99	6,78	4,52	5,64
2	4,05	0,62	4,13	1,41	3,13	1,40	1,98
3	3,68	8,88	4,23	5,05	4,78	7,75	8,48
4	6,75	4,58	0,48	3,79	6,37	3,81	1,76
5	7,59	3,68	5,52	1,23	5,76	6,70	1,10
6	0,25	3,77	7,06	5,63	0,25	9,02	5,51
7	5,29	3,44	7,93	6,00	4,05	0,08	8,35
8	8,34	1,71	4,63	7,07	0,63	7,71	4,84

MC: Train

Random Forest

Data: Apply

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT Analysis Chain

Machine Learning

scikit learn

RAPID|MINER

WEKA The University of Waikato

moa

FACT Analysis Chain

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT Analysis Chain

Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation

Fact-Tools
Library

Streams
Framework

Machine Learning

scikit learn

RAPID|MINER

WEKA The University of Waikato

moa

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

FACT Analysis Chain

Photon Extraction → Image Cleaning → Feature-Extraction → Energy-Estimation → Signal Separation

Online Application of ML

Fact-Tools
Library

Streams
Framework

Machine Learning

RAPID|MINER

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

The `Streams` framework

- Provide middle layer framework -> streams
- Simple API for data stream processes
- Model data flow graphs to describe process
- Embeddable user functions .e.g `Fact-Tools`

Algorithms JAR + XML Process Design

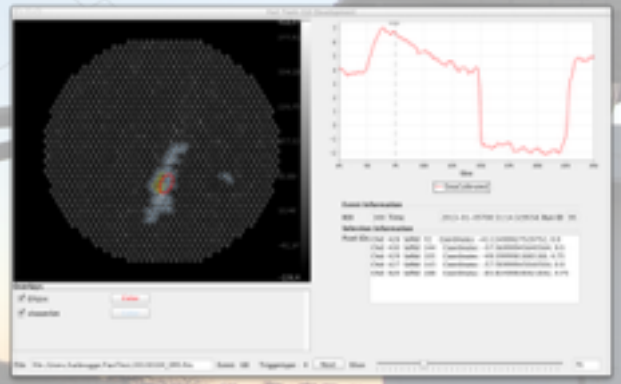
`https://sfb876.de/streams/`
`https://bitbucket.org/cbockermann/streams`

The `Fact-Tools` Library

Collection of IACT specific `Streams` processors

Methods for e.g.:

- Pixel feature extraction (# Photons, Arrival time)
- Image cleaning
- Image feature extraction (Width Length)
- Event visualization



Rapid Prototyping of processing pipeline

- test new features, different classifiers

`https://sfb876.de/fact-tools/`
`https://github.com/fact-project/fact-tools`

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Example: `fact-tools` RAW Data Processing

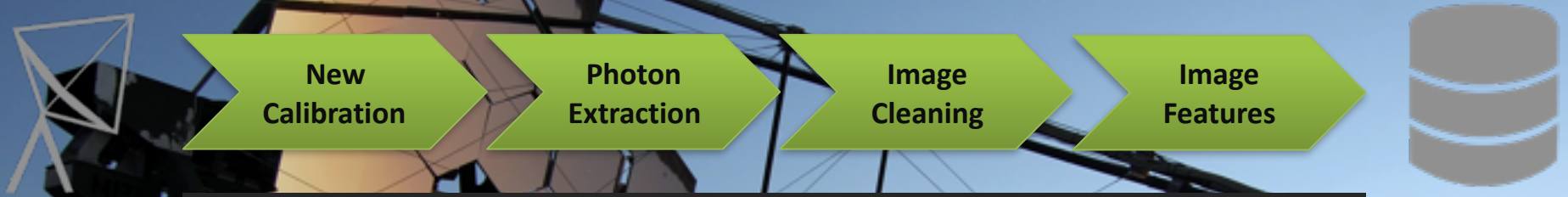
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Example: `fact-tools` RAW Data Processing


```
<stream id="fact" class="fact.ZFitsStream" url="file:/data.fz">

<process input="fact" copies="1">

     <Calibration someParameter="foobar"/>

     <Extraction outputKey="photonsPerPixel" />

     <Cleaning input="photonsPerPixel" output="shower" />

     <ImageParameter   input="shower" what="size" />

     <WriteResult  where="file:/some/url" />
</process>
```


Jens Björn Buß | TU Dortmund | jens.buss@udo.edu

Example: `fact-tools` RAW Data Processing




```
<stream id="fact" class="fact.ZFitsStream" url="file:/data.fz">

<process input="fact" copies="1">

      <MyFancyNewCalibration evenBetterParameter="foobar2"/>

      <Extraction outputKey="photonsPerPixel" />

      <Cleaning input="photonsPerPixel" output="shower" />

      <ImageParameter    input="shower" what="size" />

      <WriteResult  where="file:/some/url" />
</process>
```


Jens Björn Buß | TU Dortmund | jens.buss@udo.edu

How to scale this to streaming and Big Data?

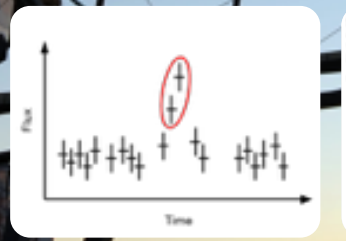
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

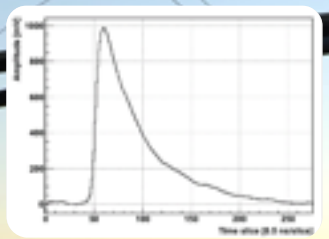
Requirements

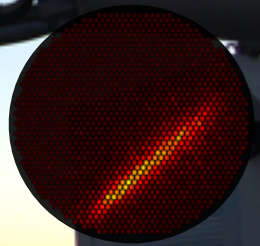
Offline Analysis
Batch

~ 5 Years

> 350 TB (source raw data)
> 240 k Files







Real Time
Stream

2 byte * 1440 pixel * 300 samples * 60 Hz > 50 MB/s

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Lambda architecture

Batch Layer

Speed Layer

Serving Layer

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Lambda architecture

Batch Layer

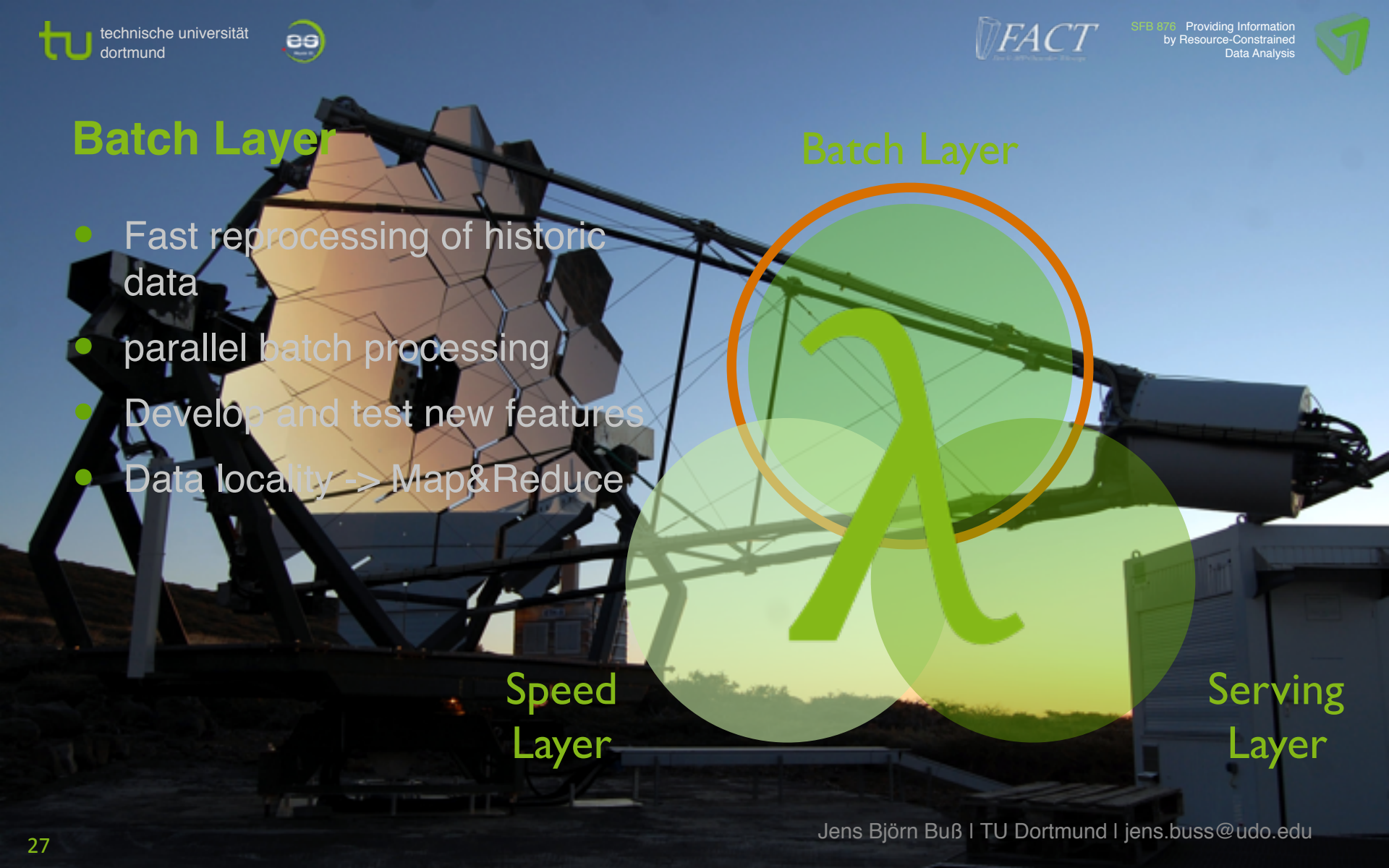
Speed
Layer

Serving
Layer

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Batch Layer

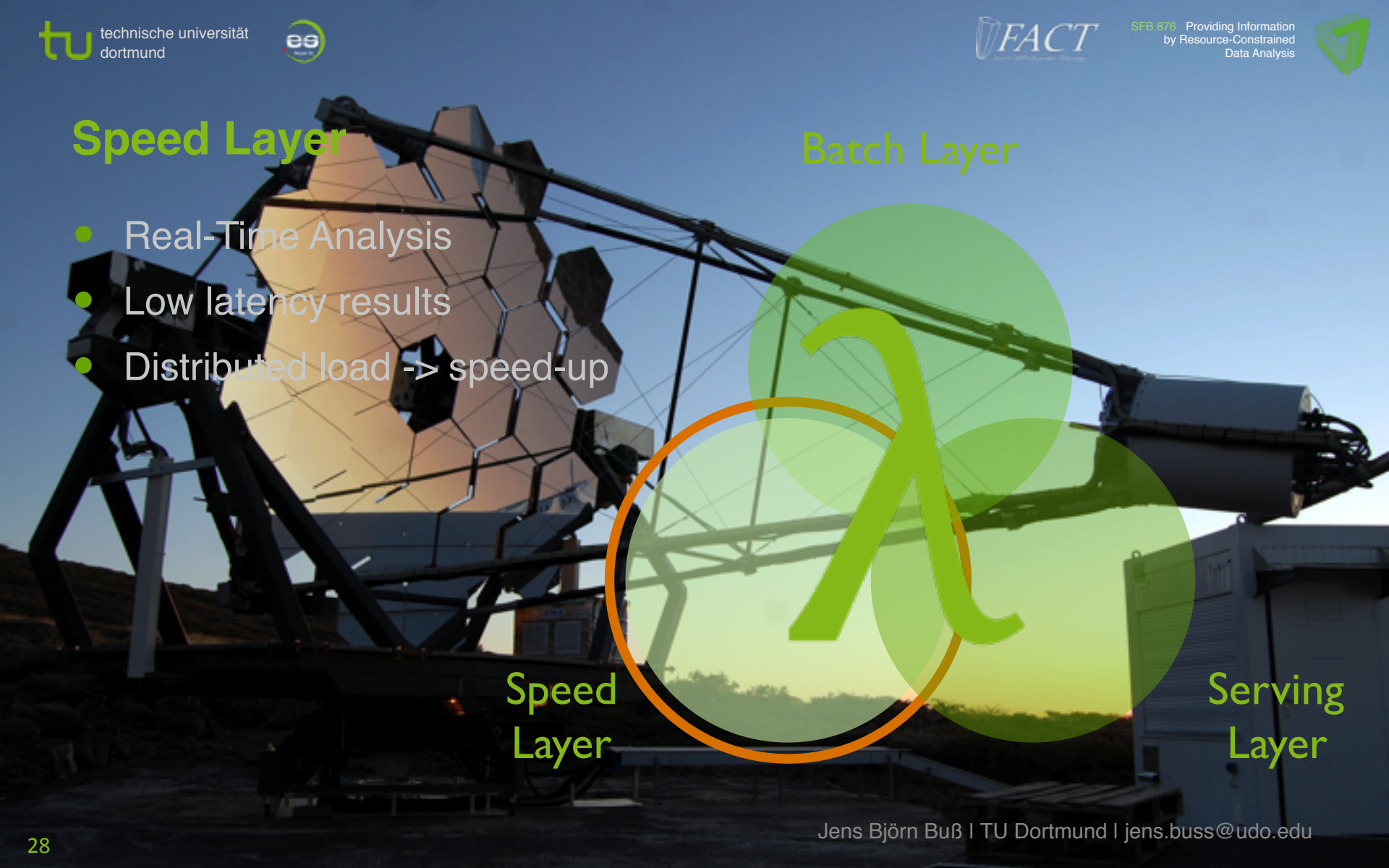
- Fast reprocessing of historic data
- parallel batch processing
- Develop and test new features
- Data locality -> Map&Reduce



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Speed Layer

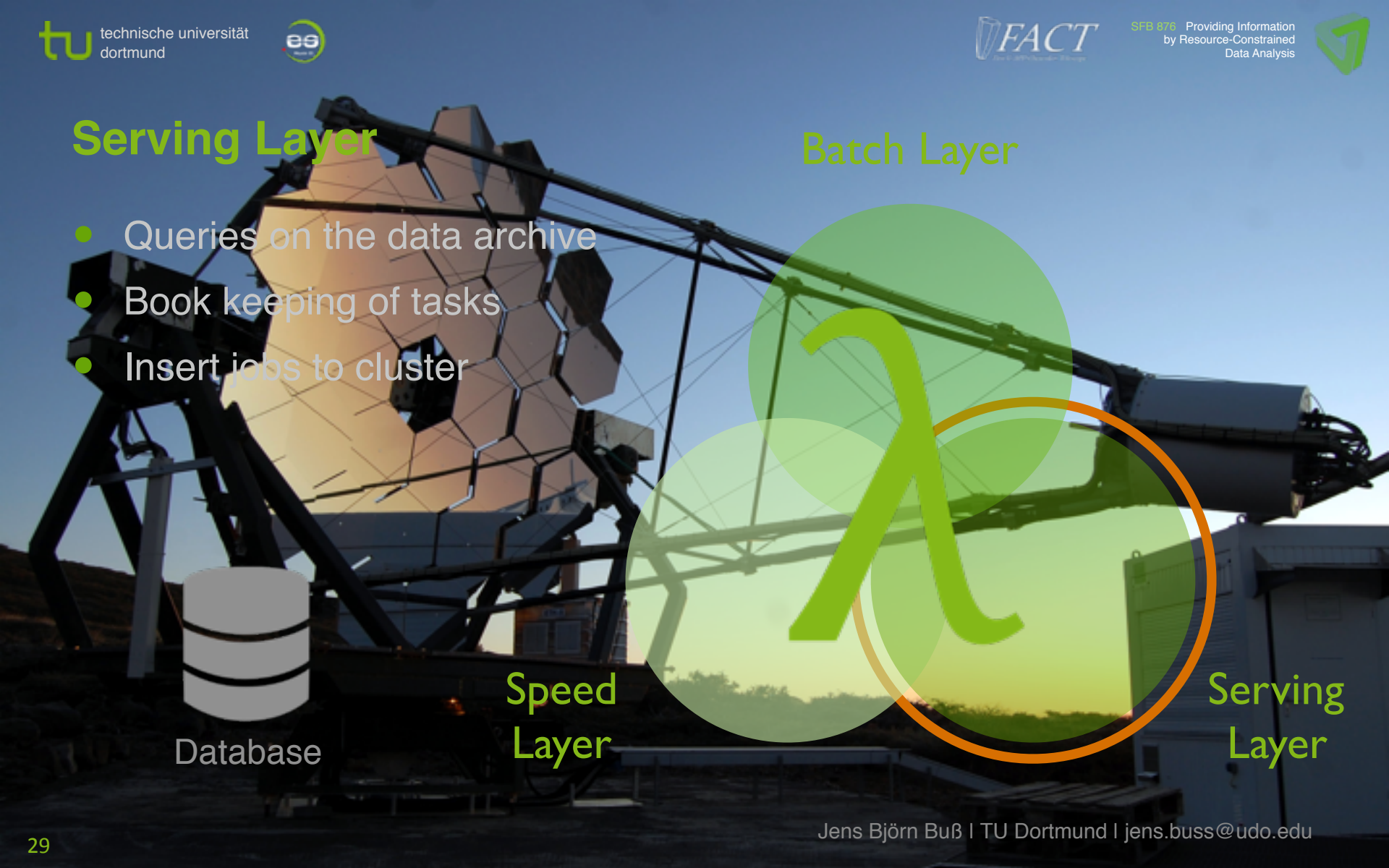
- Real-Time Analysis
- Low latency results
- Distributed load -> speed-up



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Serving Layer

- Queries on the data archive
- Book keeping of tasks
- Insert jobs to cluster



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Lambda architecture

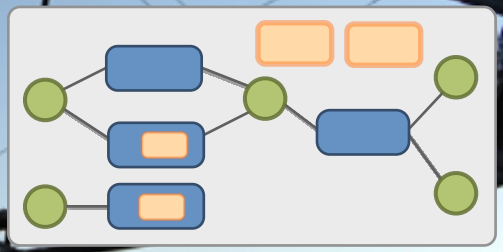
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

How to connect these tool to `Fact-Tools`?

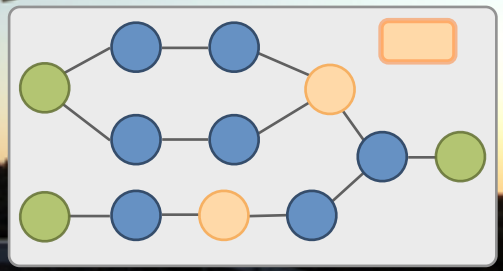
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Extention for Streams

- Use Streams as middle layer to adapt e.g. SPARK topology



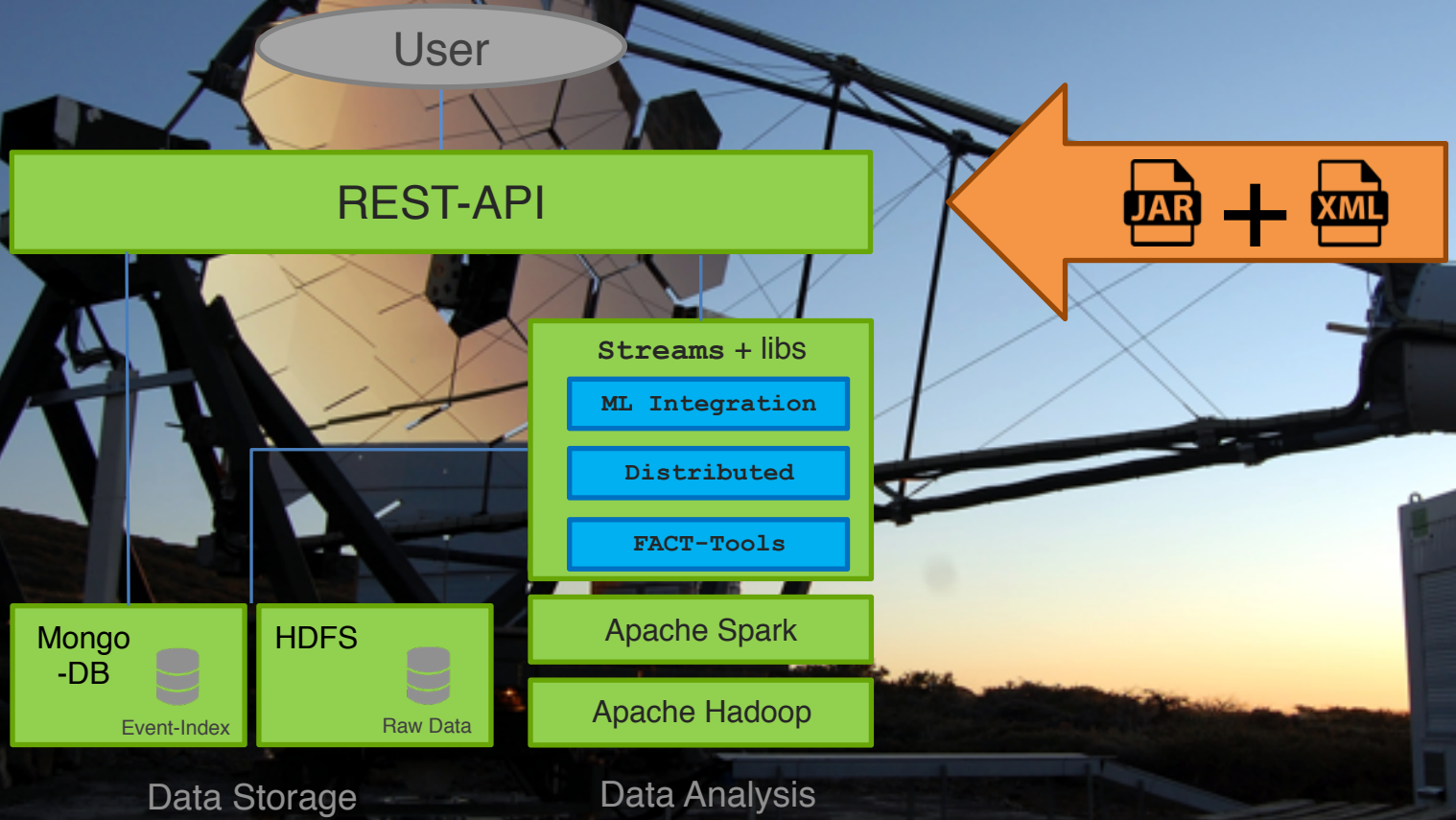
Streams Runtime



Spark

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

First approach for integration of batch system

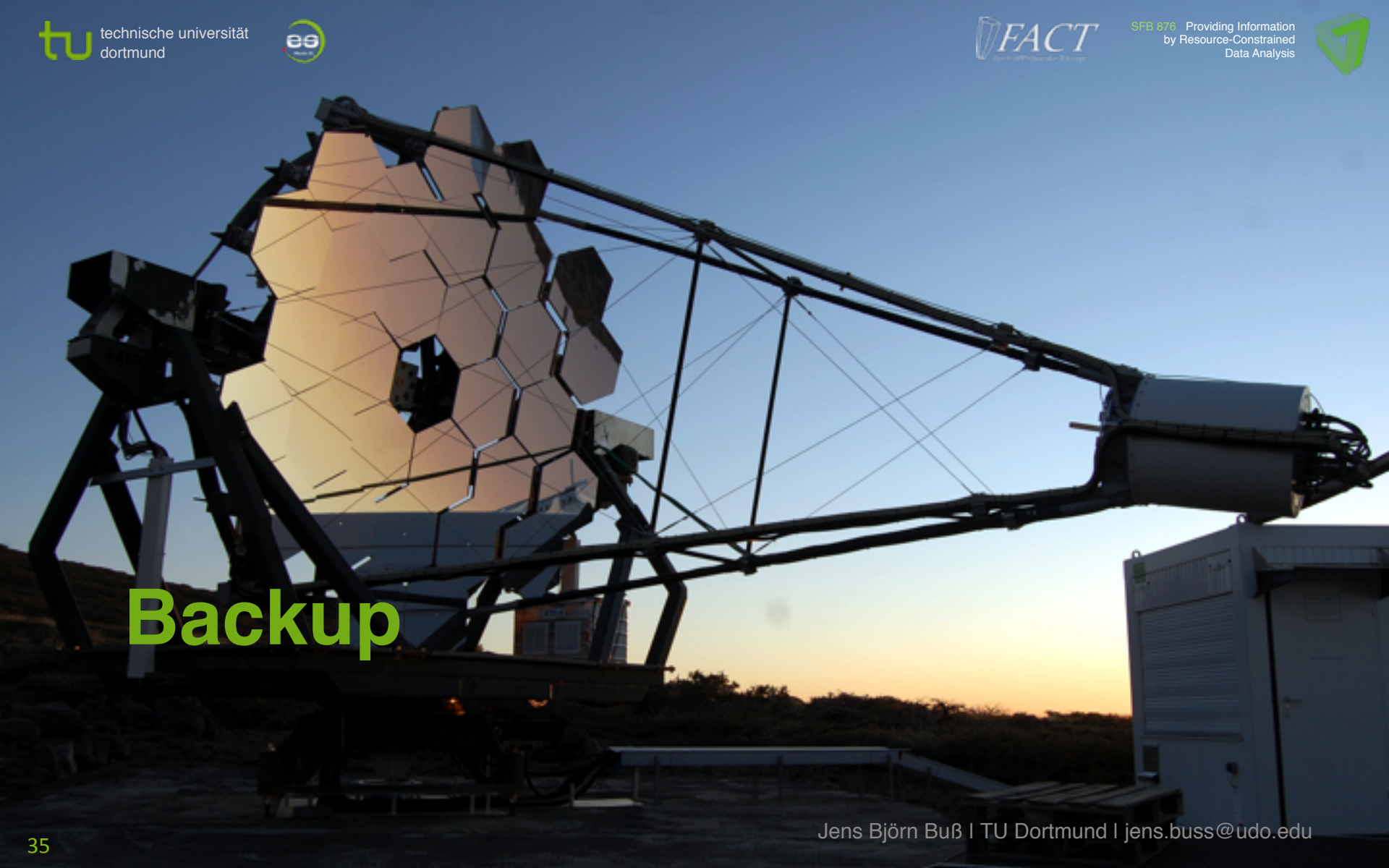


Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Summary

- Used `streams` as modeling tool for data stream processes
- XML provides flexible way for rapid-prototyping (new pipelines, test different preprocessing steps, sharing of processes)
- Same code base for speed layer (RTA) and batch layer (Archive)
- Proof of concept for `streams` as Middle layer framework to use modern cluster and streaming technology (e.g. Hadoop, Spark, Storm)
- Joined effort of computer scientists and physicist is fruitful for both fields

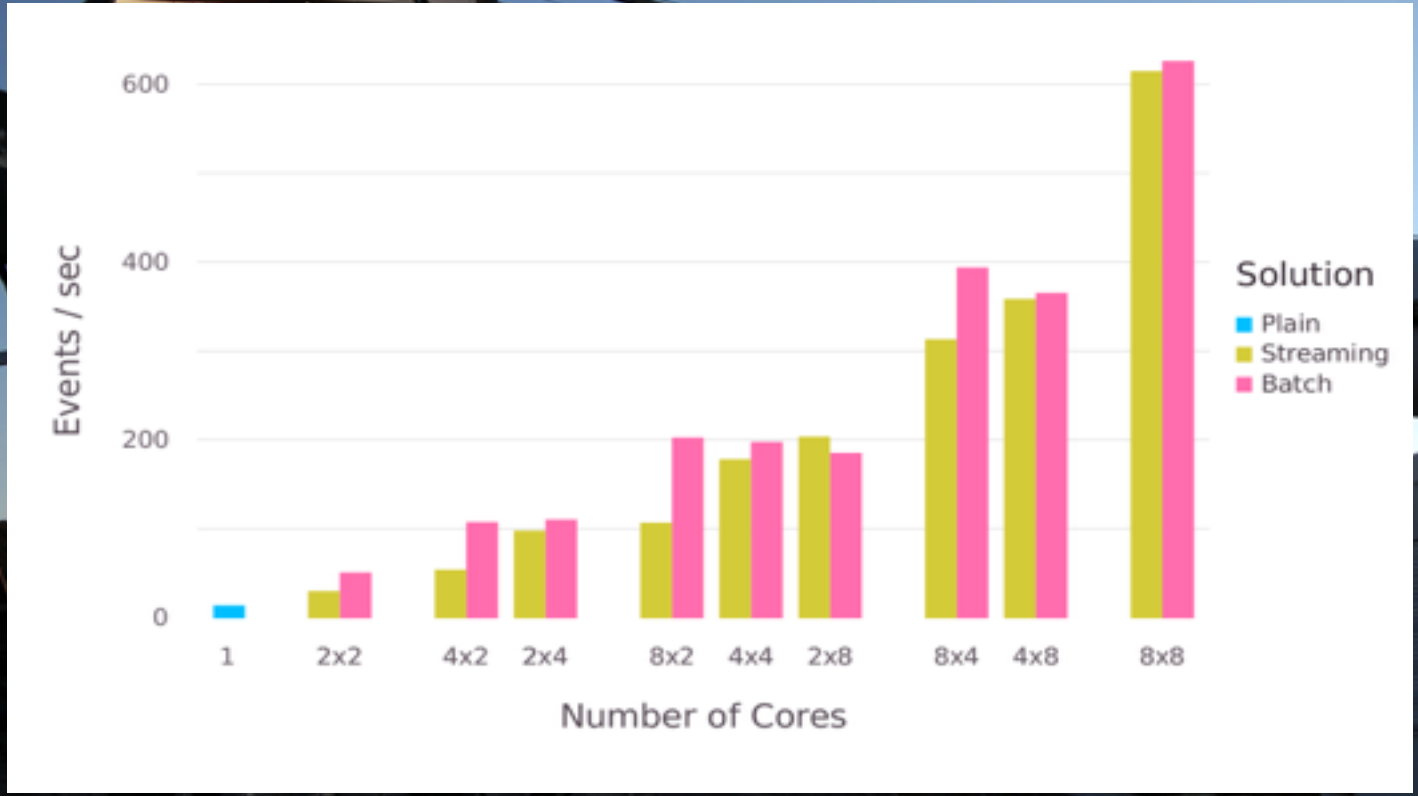
Jens Björn Buß I TU Dortmund I jens.buss@udo.edu



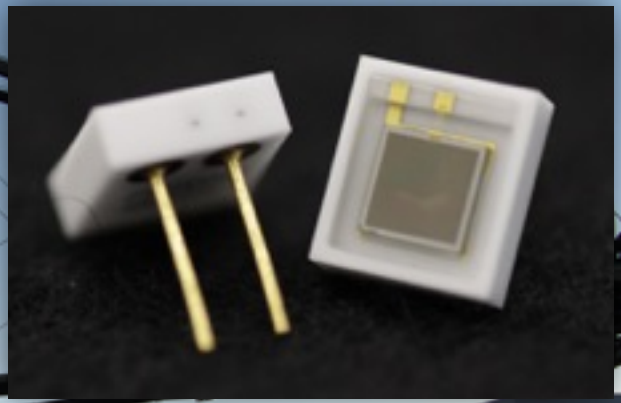
Backup

Jens Björn Buß I TU Dortmund I jens.buss@udo.edu

Performance



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu



Robust Detectors

Jens Björn Buß I TU Dortmund I
jens.buss@udo.edu

Motivation

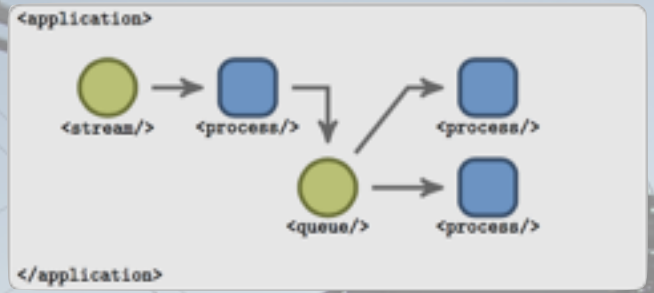
Charged Particles

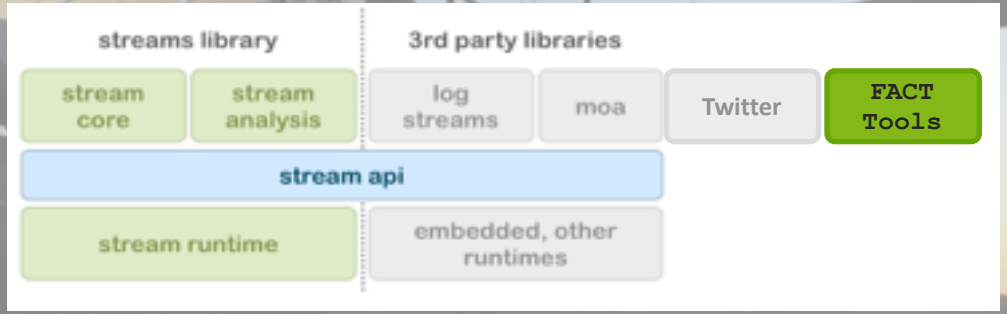
Neutrinos

Gammas

The streams framework

- Provide middle layer framework -> streams
- Simple API for data stream processes
- Model data flow graphs to describe process
- Embeddable user functions





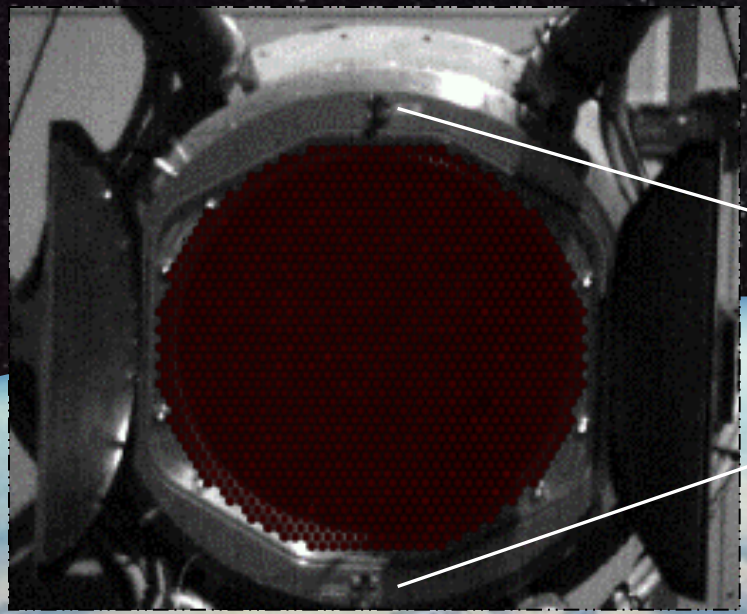
https://sfb876.de/streams/

Signal and Background

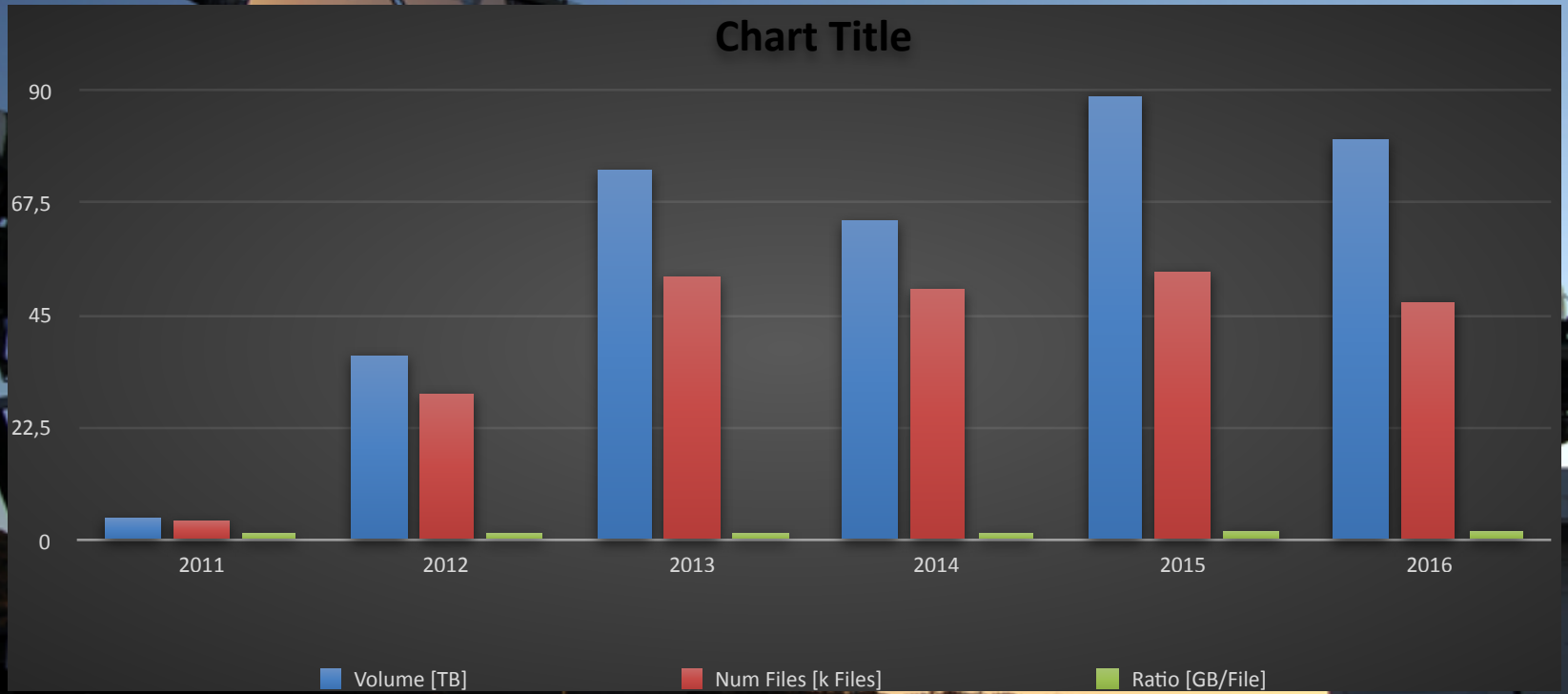
Charged Particles

Gammas

IACT Detection Principle



Data Rate



Jens Björn Buß I TU Dortmund I jens.buss@udo.edu