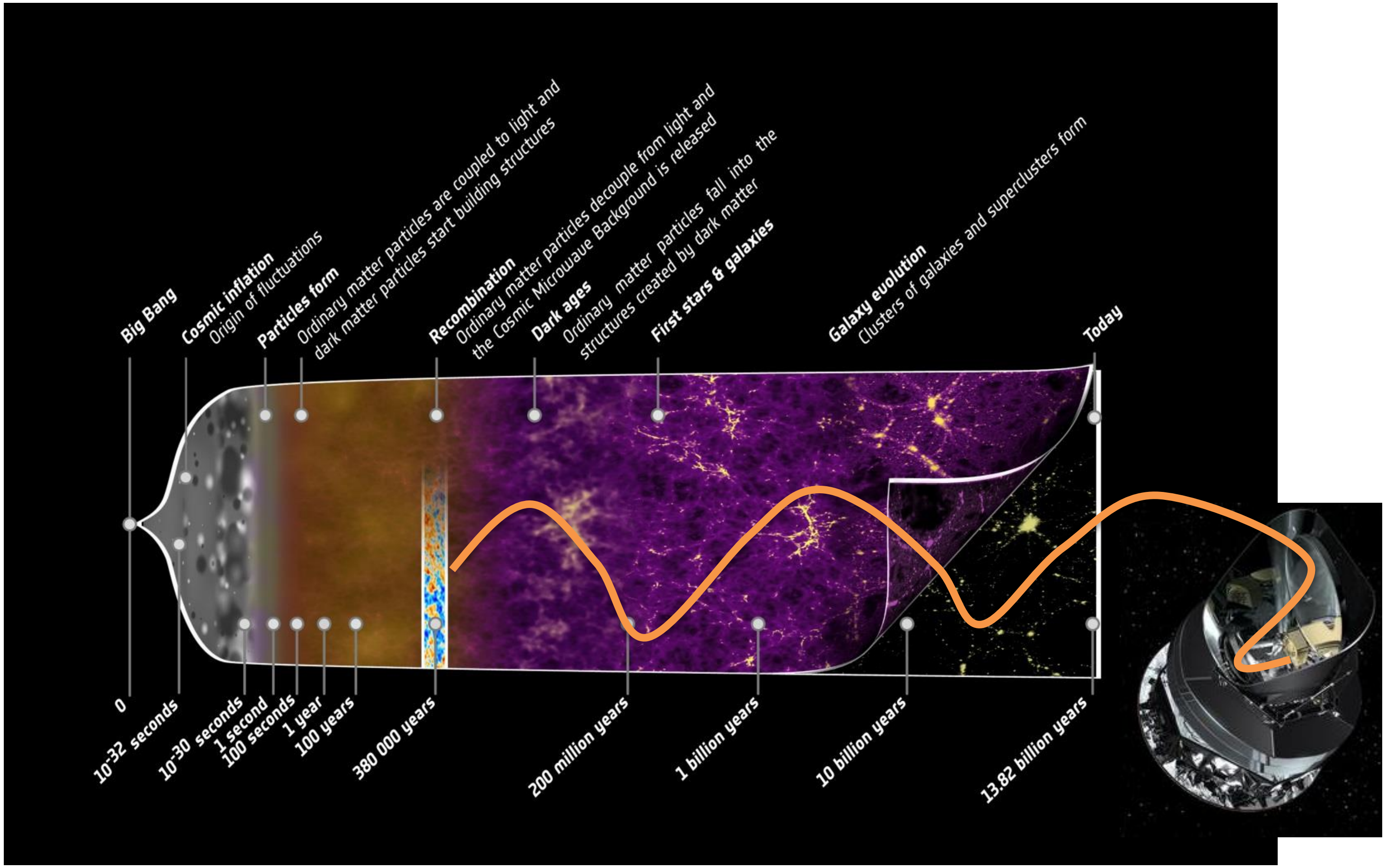
Lessons learned from Planck-HFI Data Processing Center software infrastructure

The Cosmological Microwave Background (CMB)



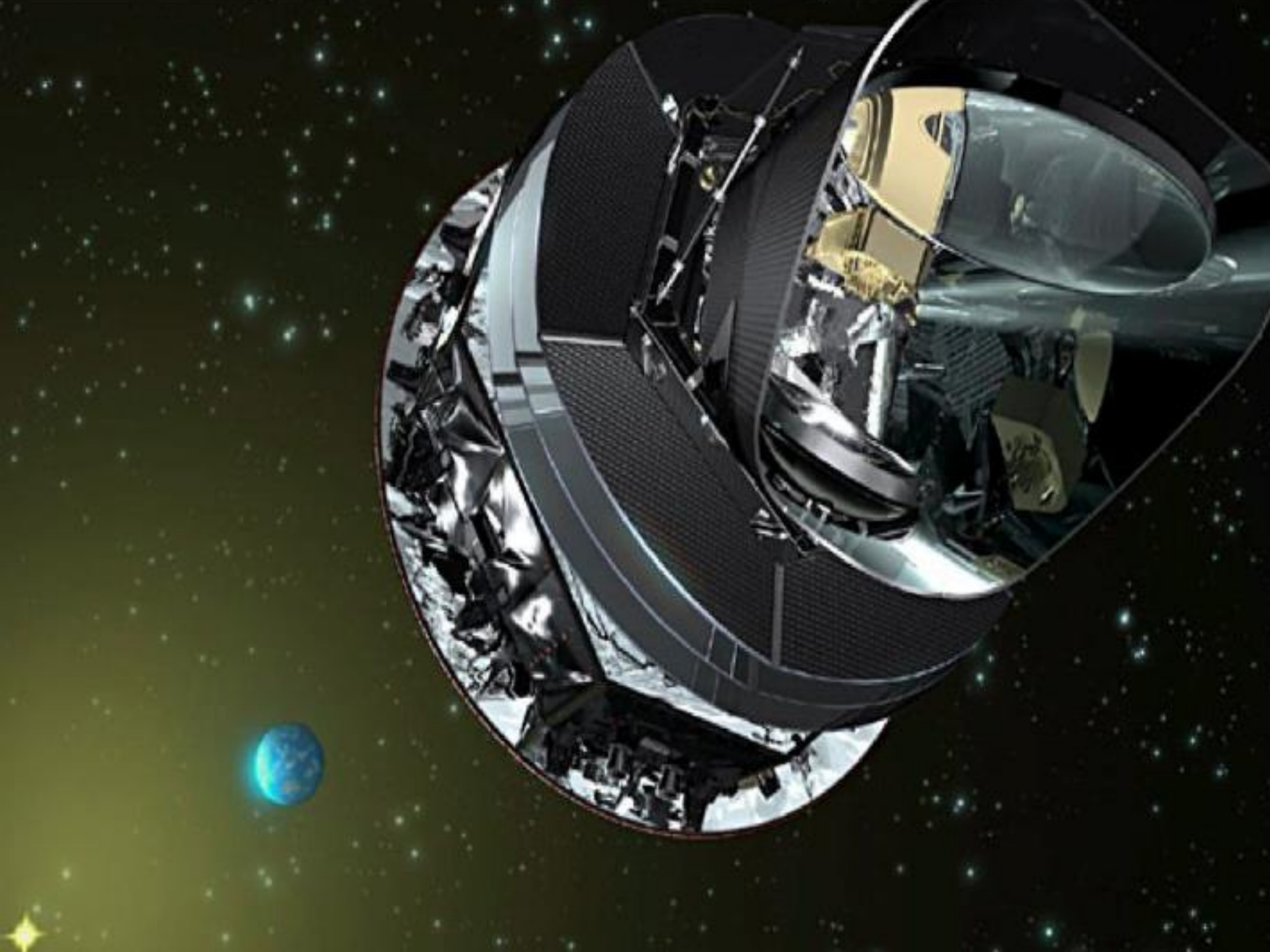
The Planck mission

Planck's major objectives are (From ESA website):

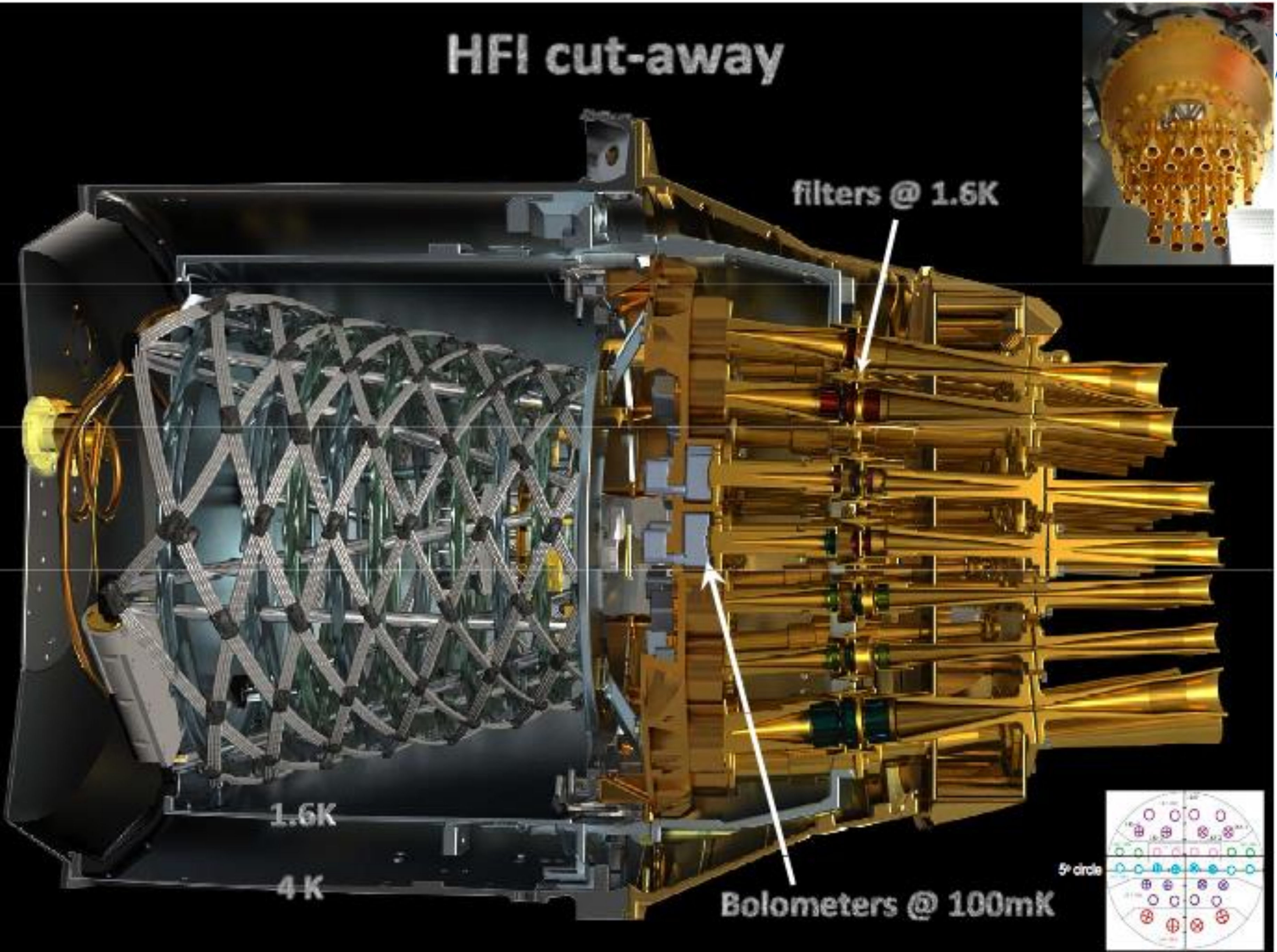
- To determine the large-scale properties of the Universe with high precision.
- To test theories of inflation,
- To search for primordial gravitational waves.
- To search for 'defects' in space
- To study the origin of the structures we see in the Universe today.
- To study our and other galaxies in the microwave.

The Planck mission

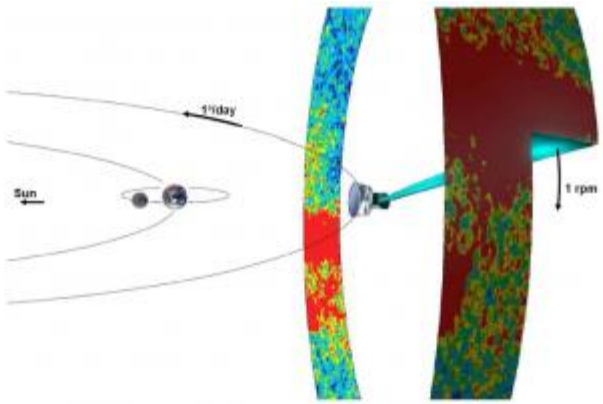
- Planck proposal was 1993
- goal was
  - to measure the temperature anisotropies down to the fundamental limits set by foregrounds and photon noise of the background
  - to measure polarization with the accuracy set by the detectors
  - measure near the peak frequency of the CMB and the minimum of foregrounds in a broad range to measure the foregrounds
- requires a large telescope to measure most of the structure (5 arc min)
- need to cool the telescope down to <50K (reached 35K)
- need to cool the detectors to very low temperature: 100mK



HFI cut-away

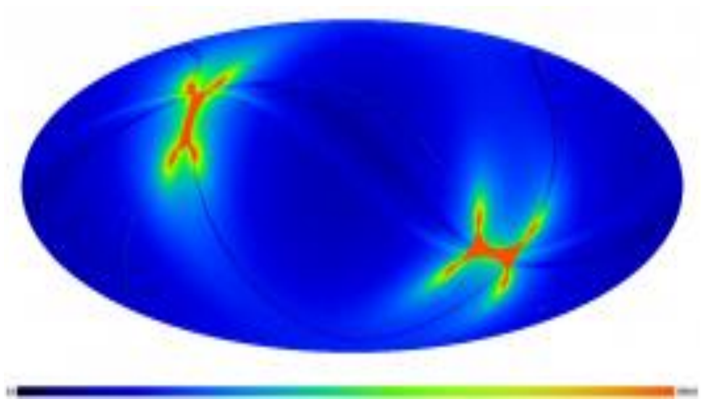


Scanning Strategy and Raw data structure

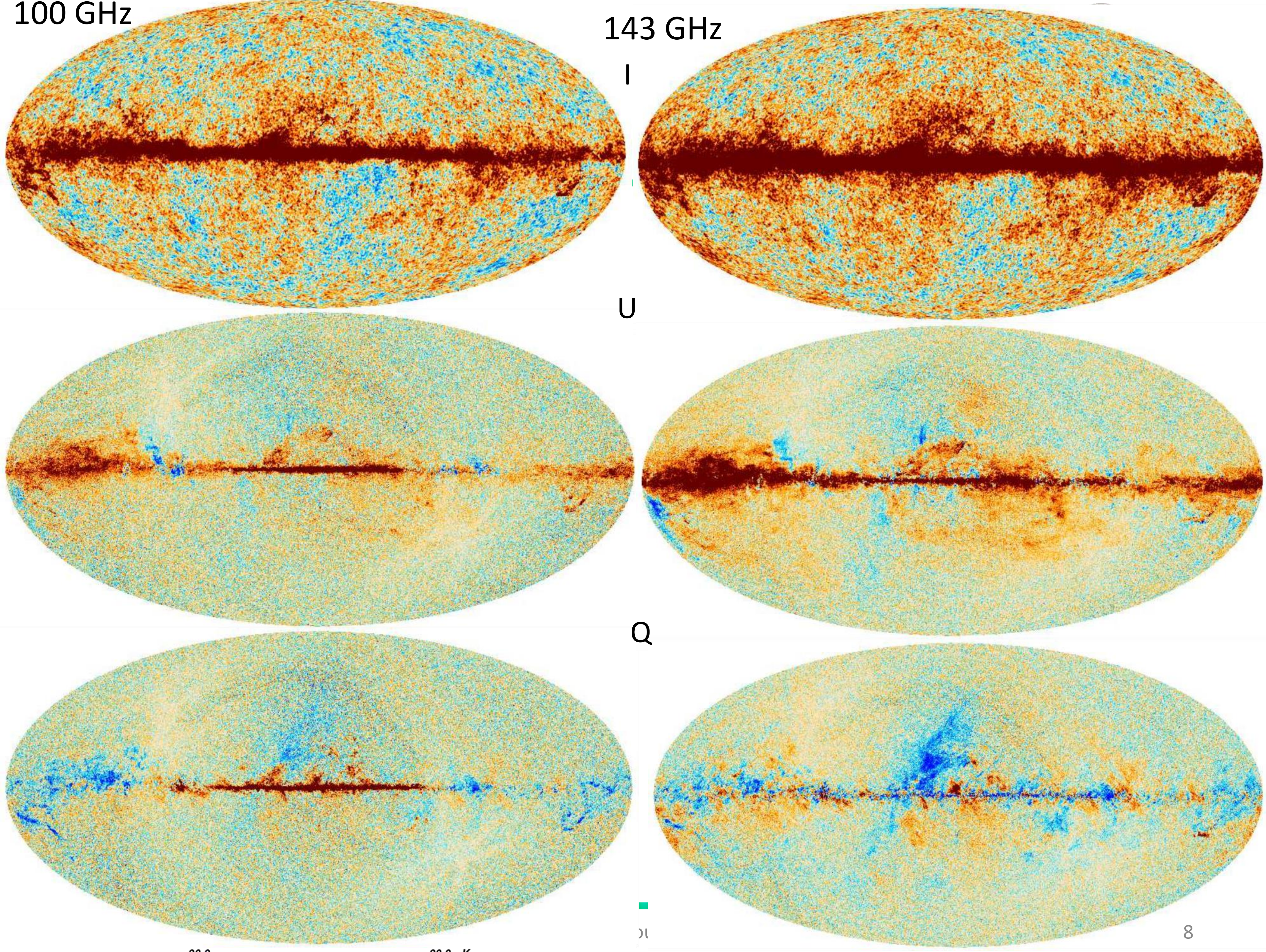


- 52 detectors timelines (~ 52x 35GB)
- Pointing solution (4*70GB)
- Duration : ~ 2,5 Year
- Telemetry to fit systematic during map making

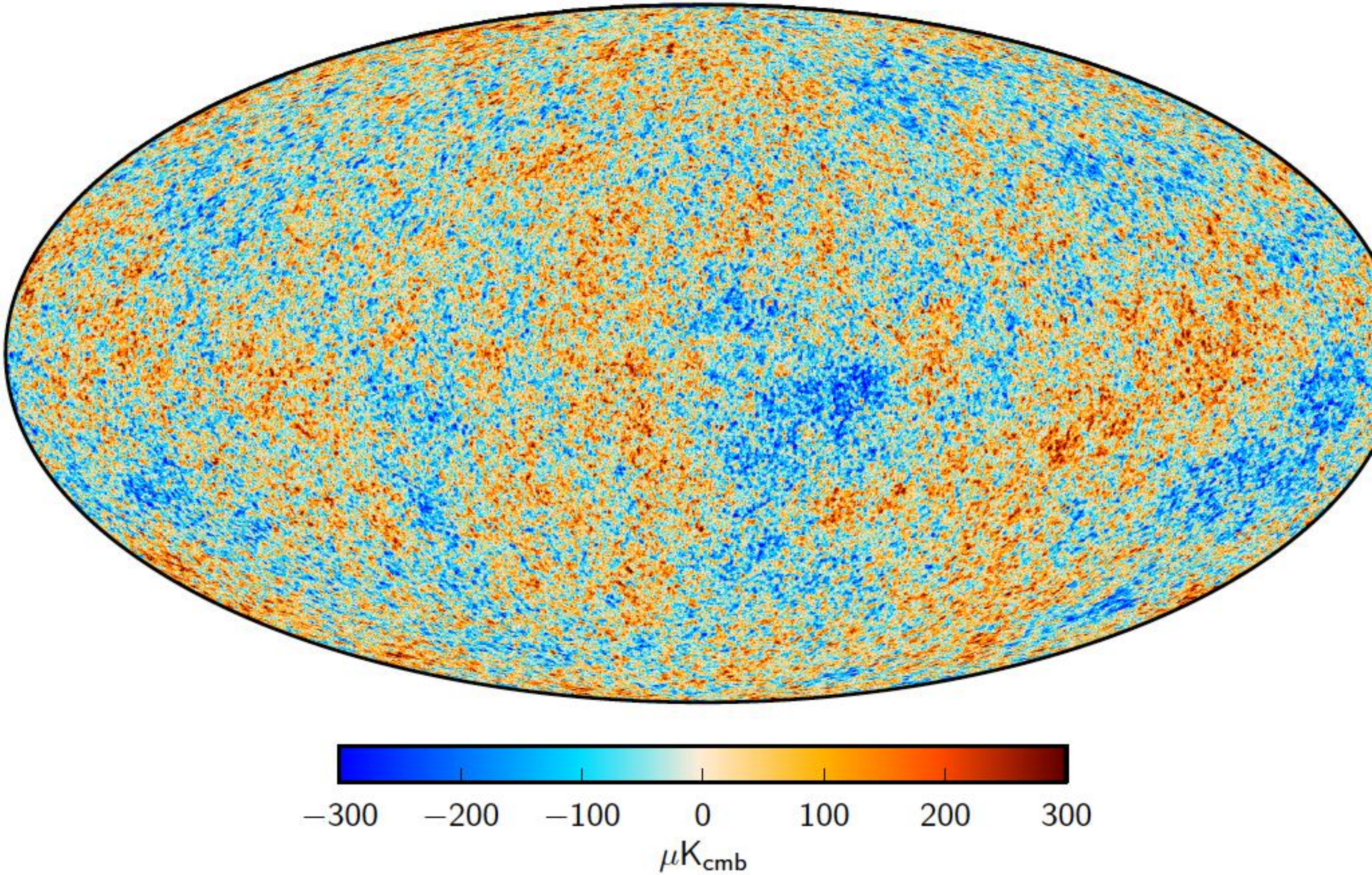
⇒ Up to 8TB for one frequency channel.



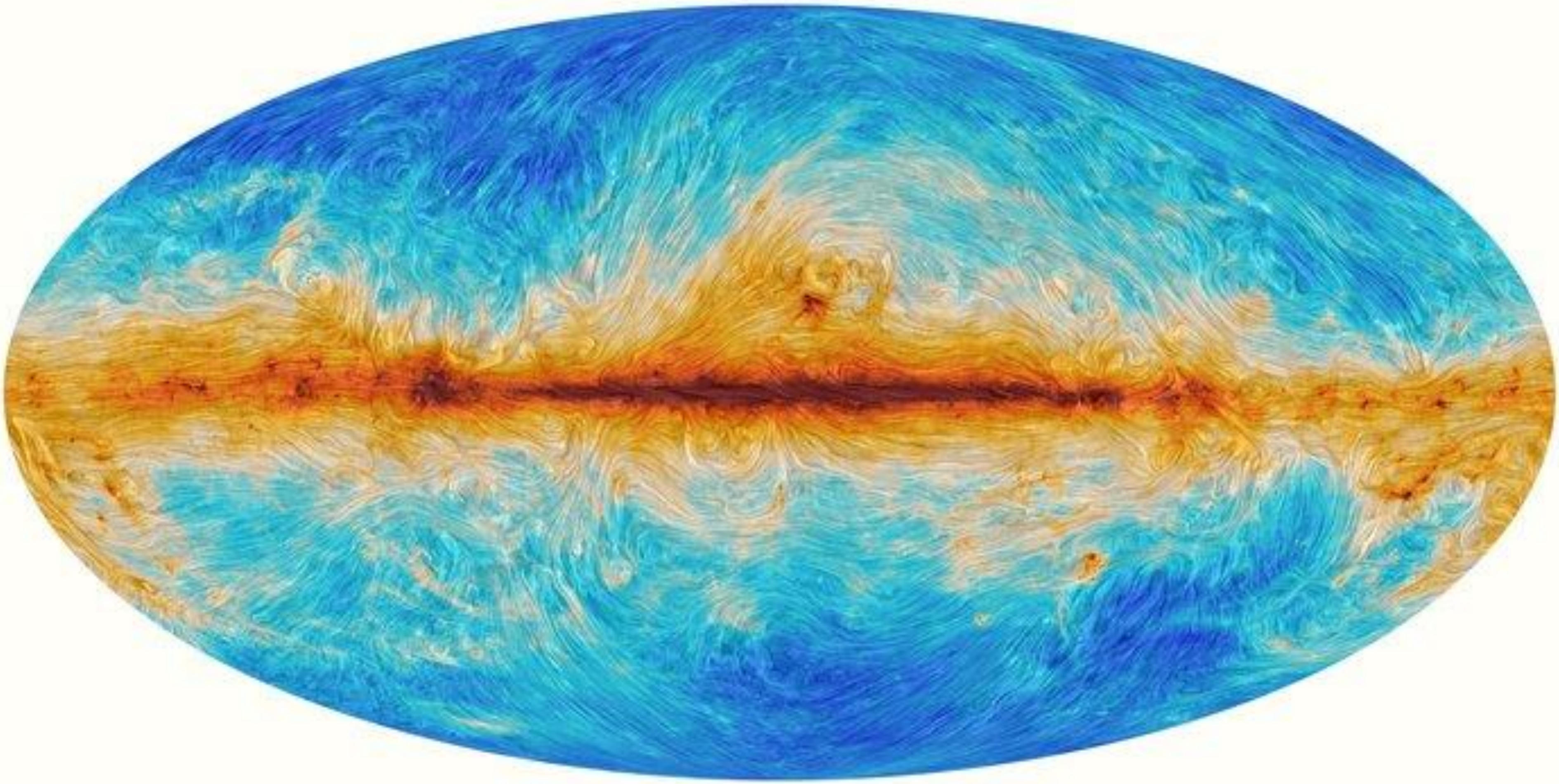
Sky coverage after one year


100 GHz
143 GHz
I
U
Q

Cosmic Microwave Background



Polarized Galactic Dust Emission



Planck specific issues

- Systematic effects dominated (not noise dominated).
- Global solution :
  - Instrumental parameters (e.g. Frequency band pass, Calibration, electronic, etc.) determined more precisely « On Sky » than during calibration.
  - Best results by merging the all data set.

Related requirements

- Minimize system overhead as much as possible :
  - I/Os, Memory, CPU.
  - Smaller is the overhead, Higher is the data quality!
- Make many data processing to understand systematic effects from the full chain (Foregrounds, instruments, Data processing pipeline)

Control Data Quality = **SIMULATIONS**!

⇒Keep track of what happen to the data.

Is Planck a ***Big Data*** project ?

- NO!:
  - Data strongly structured.
  - Simple data partitioning (Healpix, Time period).
- SO…:
  - Use PostgreSQL for metadata and history tracking.
    - Incredibly stable !!!
  - Use files for data.
    - Native format , may be an error (FITS?).
  - Languages : C/ ~~C++, F90, Java~~, Python, IDL
  - Massive computing : MPI, *OpenMP*

DPC organization

- Two instruments = two DPCs
- One main computer (IAP for HFI):
  - ~1000 cores cluster.
  - 1PB GPFS disk
- Use computing center (*CCIN2P3*, Nersc) for simulations production.
  - NERSC = 20 millions CPU hours (150000 Cores)

DPC in past .....

- Modular data processing pipeline.
- Modules developed by ~100 scientist/engineers.
- Query data inside the database to analyze the fine bias.
- Optimize I/Os.
- End-to-End simulation capability before the launch.
- Ability to process other instrument data to validate the pipelines (WMAP).

…

But .....

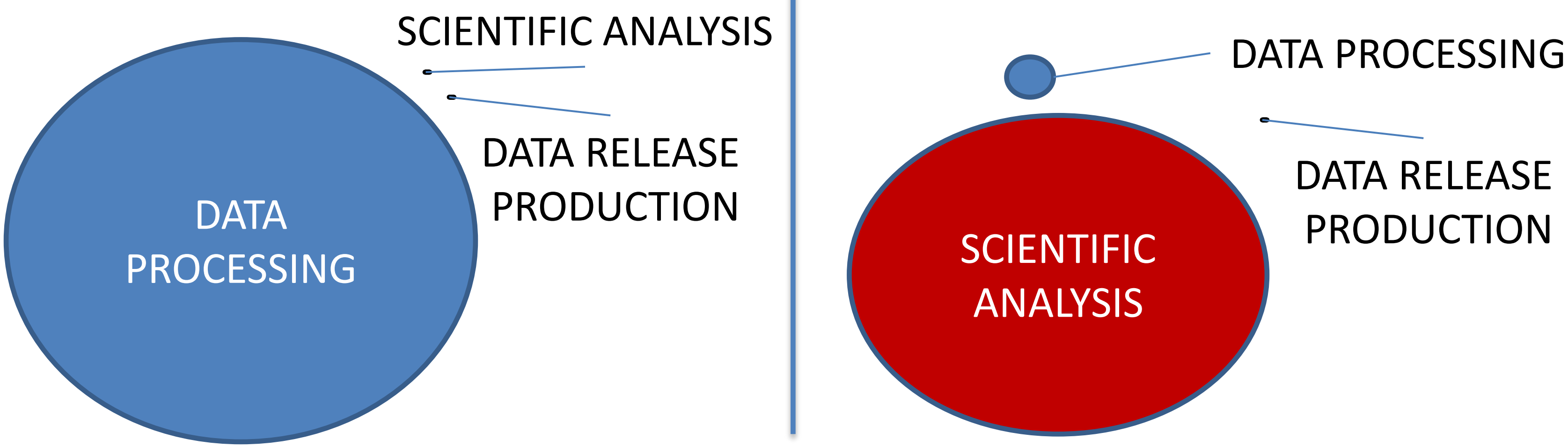
…. DPC in real

- ~~Modular data processing pipeline.~~
- ~~Modules developed by ~100 scientific.~~
- ~~Query data inside the database to analyze the fine bias.~~
- Optimize I/Os.
- ~~End-to-End simulation capability before the launch.~~
- ~~Ability to process other instrument data to validate the pipelines (WMAP).~~

HFI DPC day-to-day life

Computing Resources
Software Development

Human resources



- 99% of the computation is used to understand the Data (Processing + Simulations)
  - and even more for simulations to characterize the production.
- This is done by much less than 10% of the DPC participants.
- Most of the scientific analysis is done outside the DPC S/W infrastructure:
  - Need for data access and computing faculties.

Management answer

- Data processing (and simulation) developers should be an identified team with an unified management:
  - ~ 30 people.
  - Need for a synthetic view to find fine systematics.
  - Dangerous to cut the processing in several level under several responsibility.
  - Cut could be done vertically (between instruments) but not horizontally (inside the pipeline).

- Wide scientific expertise is needed to qualify the product and support the development team:
  - >100 people.
  - Short loop on data characterization.
  - Propose algorithmic solutions.

Database & S/W for data processing

- **No need for an advance database** that was not use for any algorithm.
- Keeping processing **history tracking is the key issue** to understand the data.
- Less than 20% of the software infrastructure functionalities have been used.
  - No need for a complex and generic infrastructure, the data processing developers are sufficiently clever to survive with low level of services.
  - Easer to get the needed I/O, CPU efficiency with a simple infrastructure.

⇒ **Normalize logs and parameters input/ouput** to keep history,

⇒ Only **develop the blocking issue** for infrastructure (no nice to have) and in an **agile** manner.

⇒ Develop **common tool** between instruments should be **opportunity based** (not mandatory).

⇒ Efficiency for large data sets.

Database & S/W for data processing

- Final HFI pipeline tend to be **ONE big module** (including simulation).
- **Multi module pipeline is not efficient** to couple the processing with the simulation and for performances (e.g. I/Os).
- All data processors should be able to run the all chain in order to characterize its change against the final accuracy.

⇒ **Build libraries** (common configuration management),
⇒ **One language** (C in order to wrap it in what ever is needed latter),
⇒ **One module to run end-to-end** in a standalone model.
⇒ **MPI** to get use of **HPC** center built to run one huge job.

Database & S/W for production

- Production is mainly declaring a data set official.

⇒**No particular concern about production** it was a **subdominant** aspect in term of computing and human resources.

Database & S/W for scientific analysis

- Paper and **science analysis were most often done outside the software infrastructure (e.g. using maps)**.

⇒No plan for any software tool, scientist use their own (support for hardware and s/w development expertise).

Simulations

- Key point to understand the data.

- Should be integrated from the beginning inside the data processing pipeline/module.

- **Common and simple infrastructure between data processing and simulations**:
  - Non commonality between simulation and processing pipeline cost a very huge price for HFI-DPC.

Finally …

- Thanks to hundreds of production and simulations the large scale polarization systematic effects have been divide by x1000!

⇒Despite our ground segment design errors we succeed to reach almost the noise limit at all scales for the 2016 release.

Design errors = Time consuming = cost increasing

Conclusion

- Data processing characterization needs 100x the resource for the data production (volume and CPUs).

SIMULATION&DATA PROCESSING FULLY INTEGRATED

- Data production is subdominant (~1/100 of the data analysis)

⇒ Pipeline tend to be **ONE MODULE (C/MPI).**

- Scientific activities happen outside the infrastructure (mandatory to debug products and propose solution).

SOFTWARE INFRASTRUCTURE SHOULD BE MINIMIZED

- Data processing is driven by few people (mainly software engineers with signal processing background) who also know the instrument.

ALGORITHM DEVELOPMENT = SOFTWARE DEVELOPMENT

Questions