An Integrated Software Framework for Magnetic Measurements

-

From Raw Data to Assets

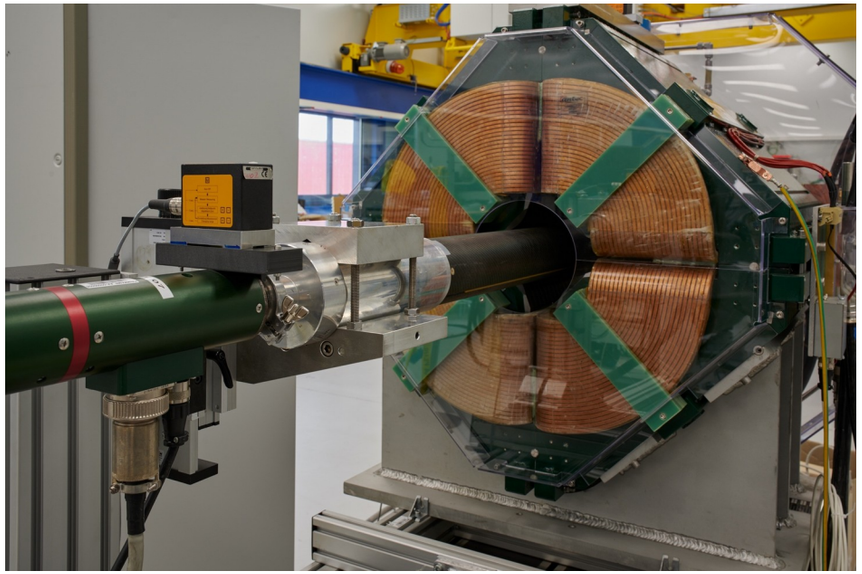
Matthias Bonora

CERN
TE Technology Department

Outline

- Magnetic measurement requirements
- FFMM – A Flexible Framework for Magnetic Measurements
  - Concept and idea
  - Components of a measurement script
  - Connection to webservices
  - Integration examples
- Development goals and future plans

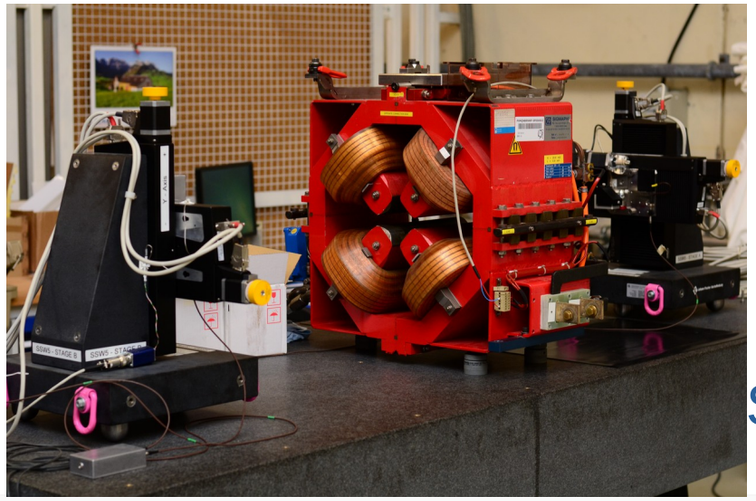
Magnetic Measurements



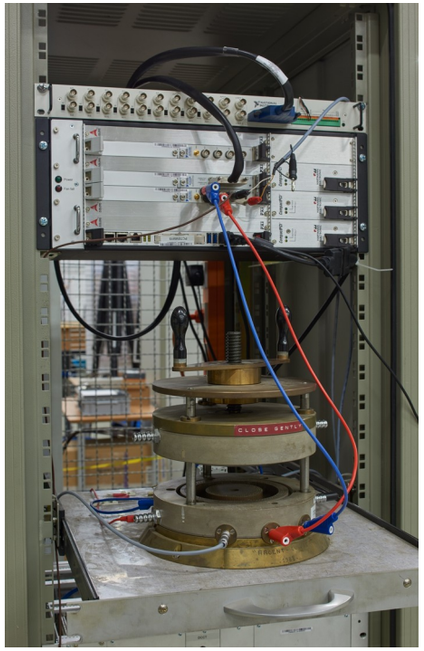
Rotating coil systems

Helmholtz coils

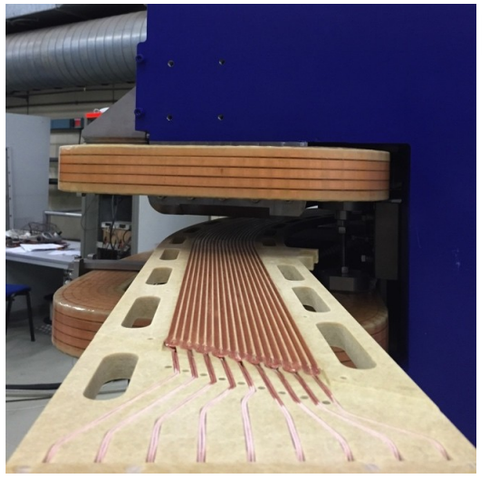
3D mapper



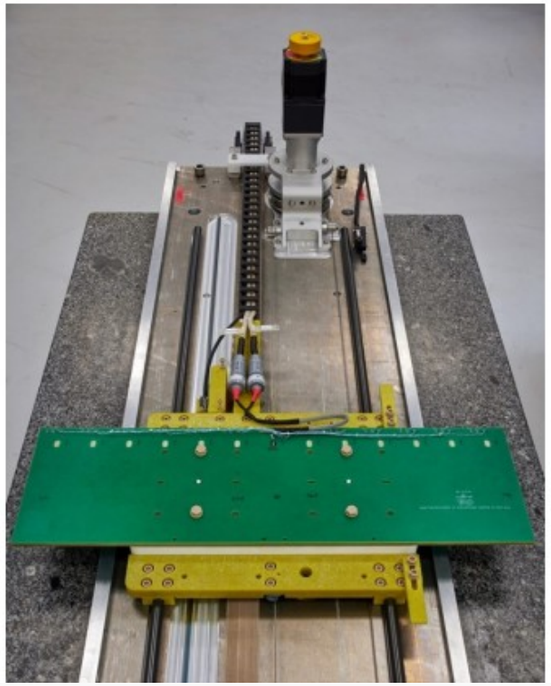
Stretched wire systems



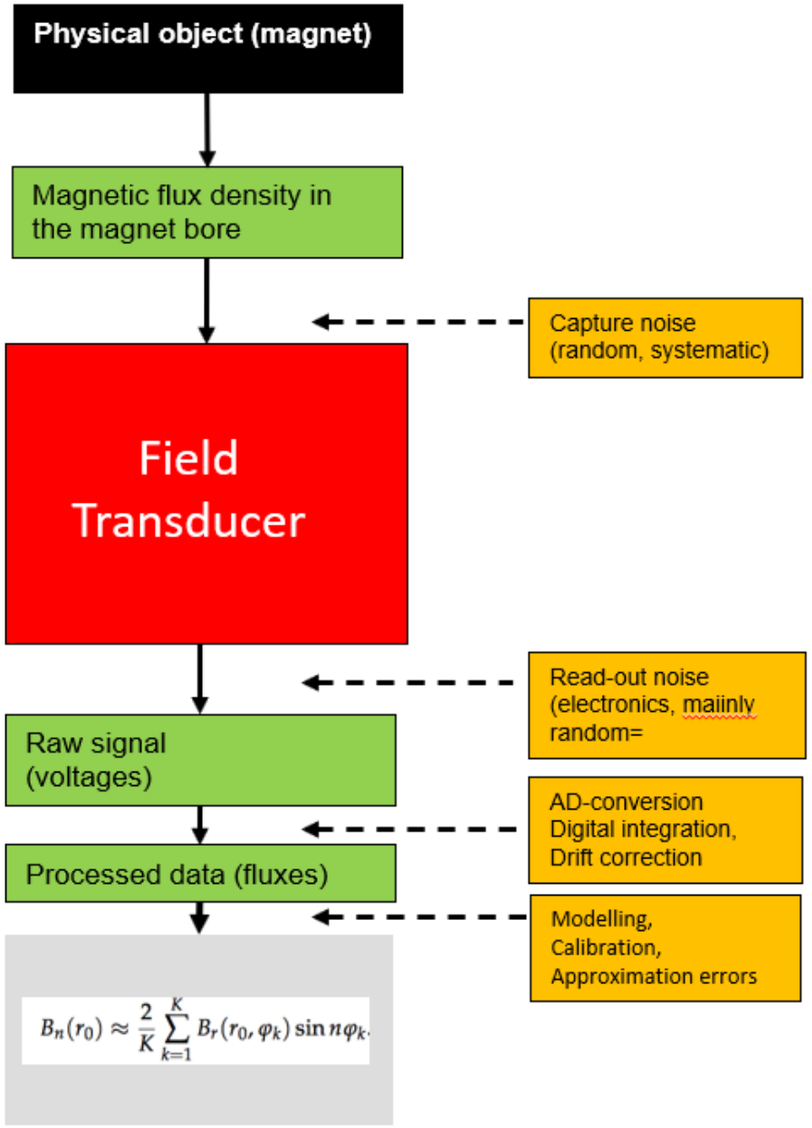
Ring-Sample Permeameter



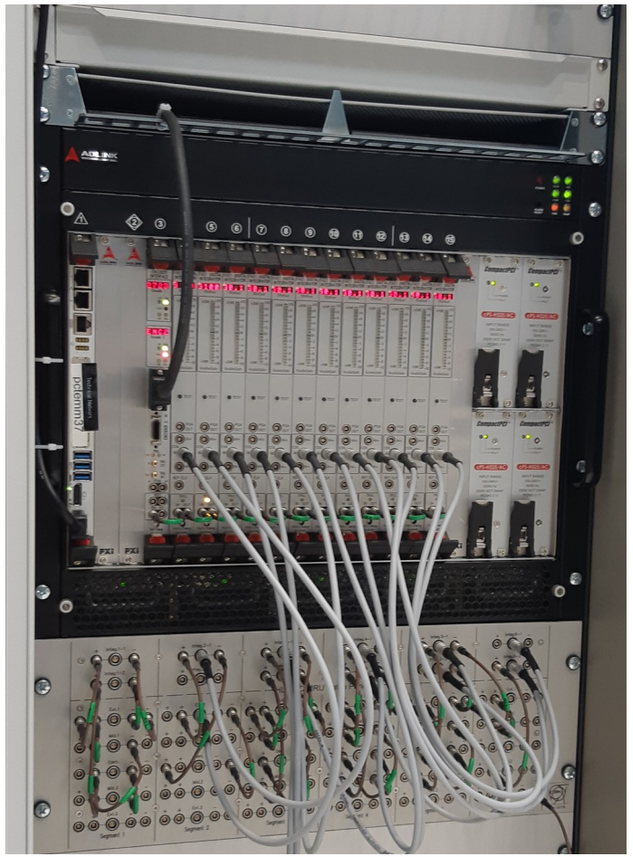
Fluxmeter

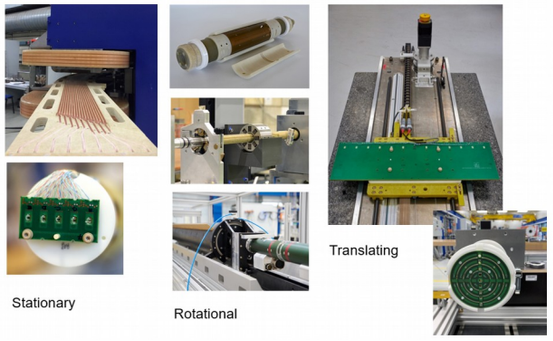


Translating Fluxmeter



A Magnetic Measurement Bench (1)





- DAQ
- Levemeters
- Motors
- Multimeters
- Teslameters
- Power Converters
- Digital Integrators
- Waveform Generators

Devices

A Magnetic Measurement Bench (2)

Setup → Acquisition → Analysis → Results

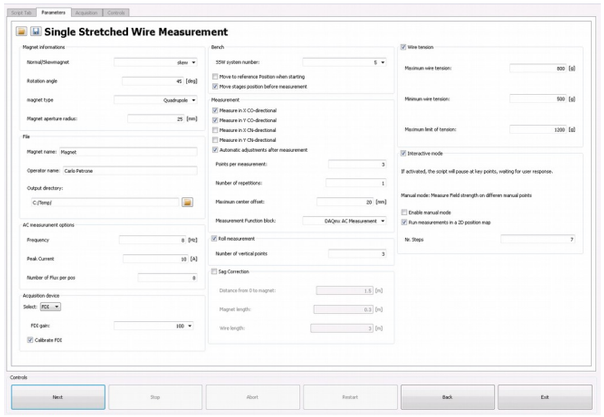
- Measurement script
- Parameters

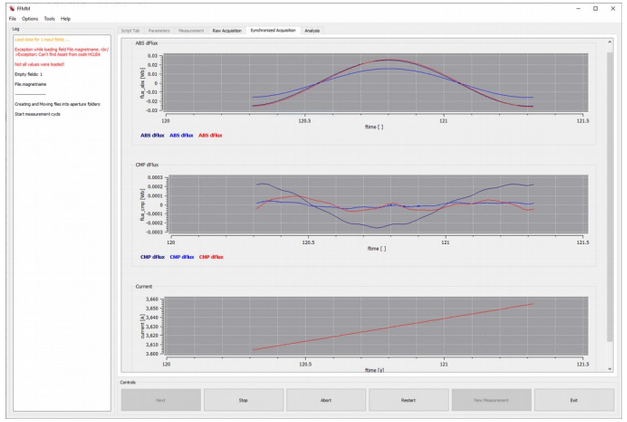
- Raw data
- Pre-processed data

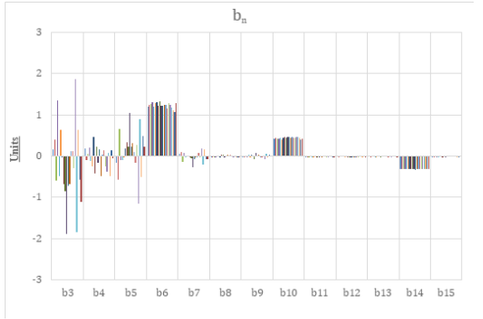
- Voltages, currents, fluxes

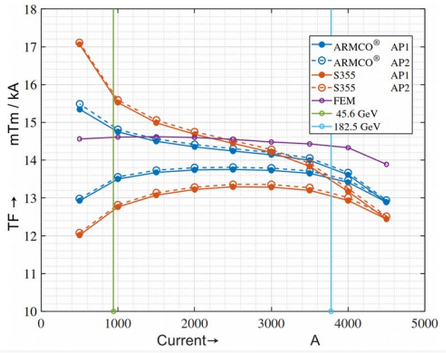
- Harmonics, multipoles, field, center offset, roll angles

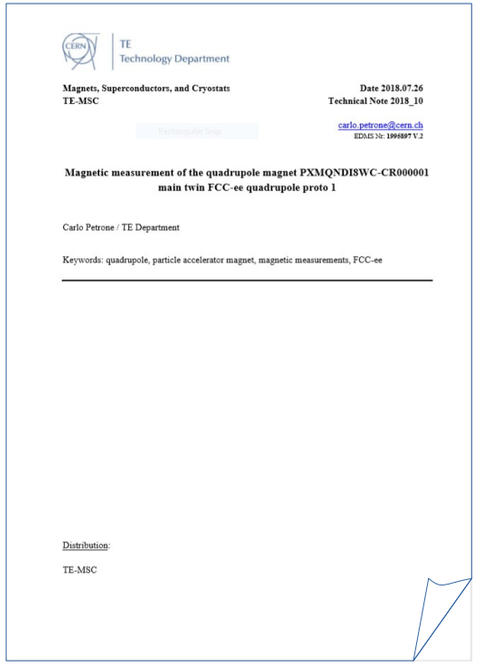
- Measurement report
- Summaries










TE Technology Department

Magnets, Superconductors, and Cryostats
TE-MSC

Date 2018.07.26
Technical Note 2018_10

carlo.petrone@cern.ch
EDMS Nr: 1998897 V.2

Magnetic measurement of the quadrupole magnet PXMQNDI8WC-CR000001 main twin FCC-ee quadrupole proto 1

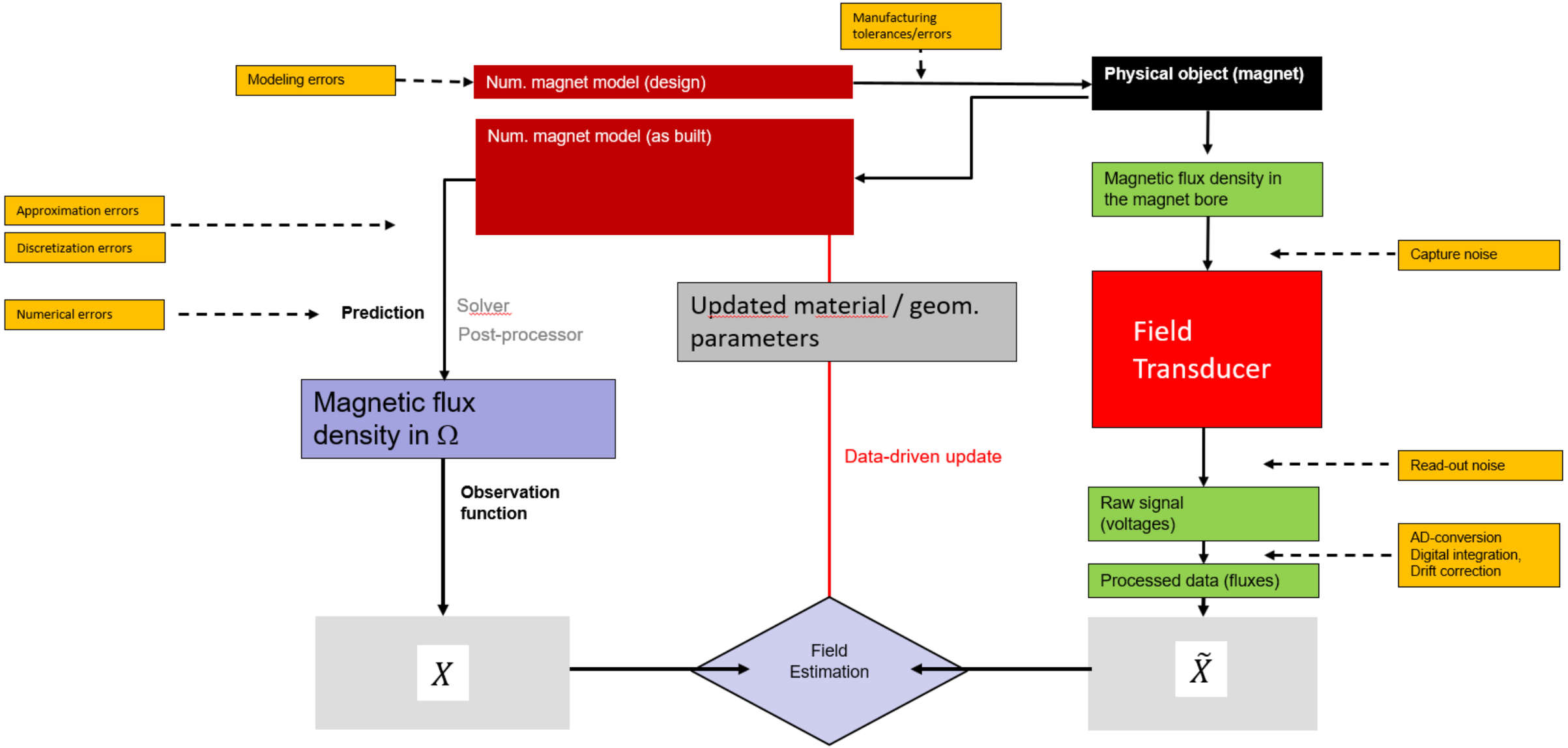
Carlo Petrone / TE Department

Keywords: quadrupole, particle accelerator magnet, magnetic measurements, FCC-ee

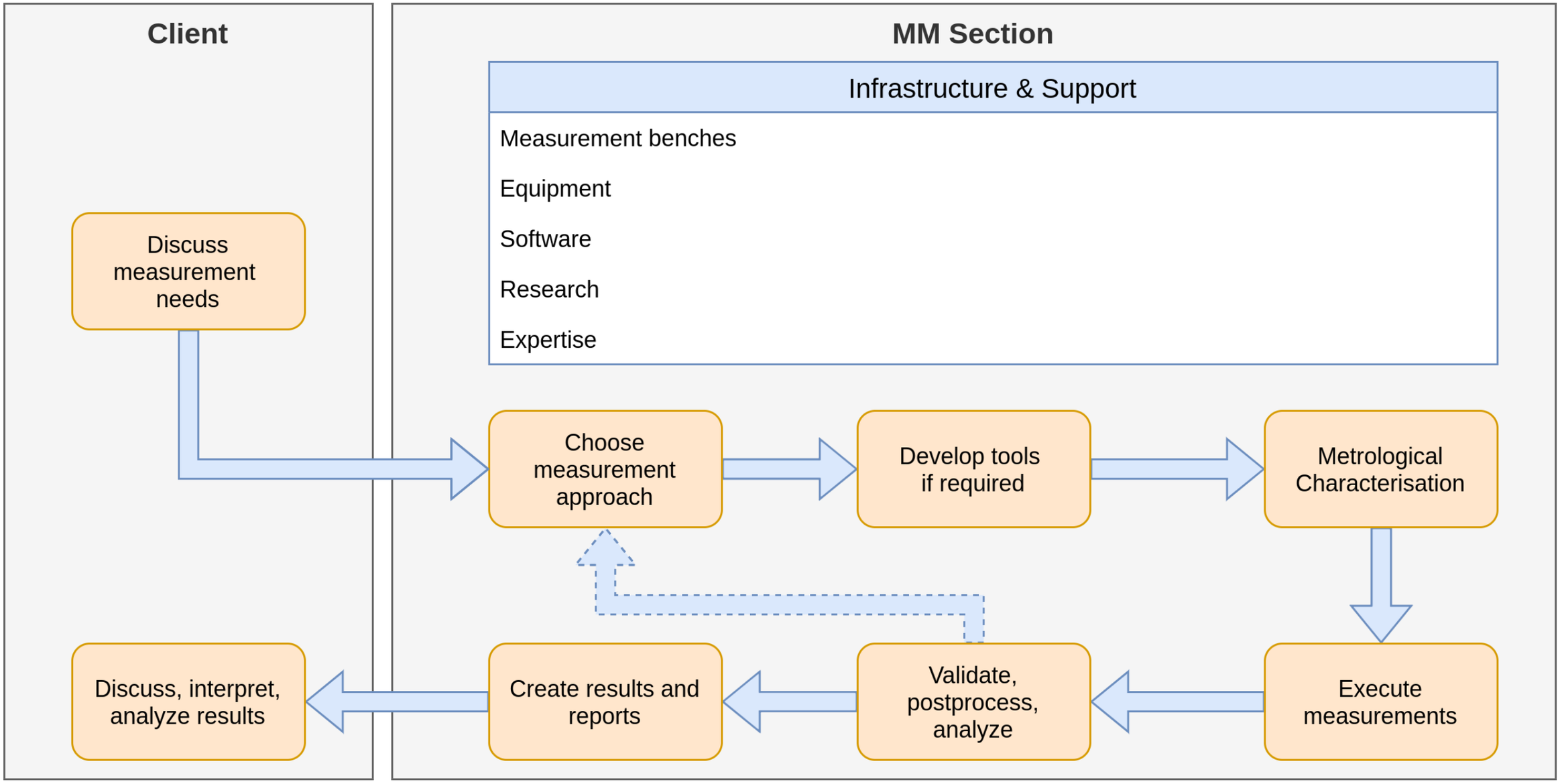
Distribution:

TE-MSC

Magnetic Measurement Results



The TE-MSC-MM Value Shop

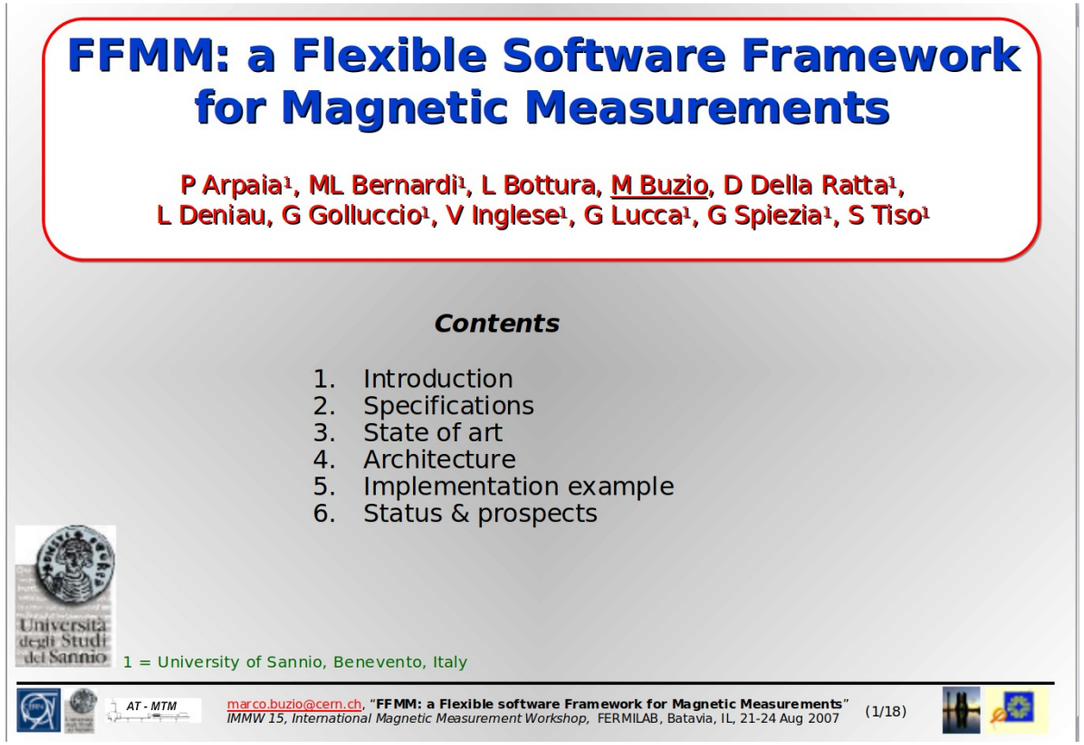


- Operation of many different measurement benches and types
- Similar, yet different acquisition systems
- Use of different sensors and actuators
- **Resource optimization**
  - Symbiotic benefit from measurement bench developments
  - Keep and reuse development expertise
  - Benefit from short term contracts and student contributions

- Need for an efficient software platform as base of operations

- Idea for a software framework for magnetic measurements

- Reusable
  - Easy to implement small, independent blocks
  - Blocks are reusable when needed
- Flexible
  - Easy to write measurement applications
  - Still full control and extendability



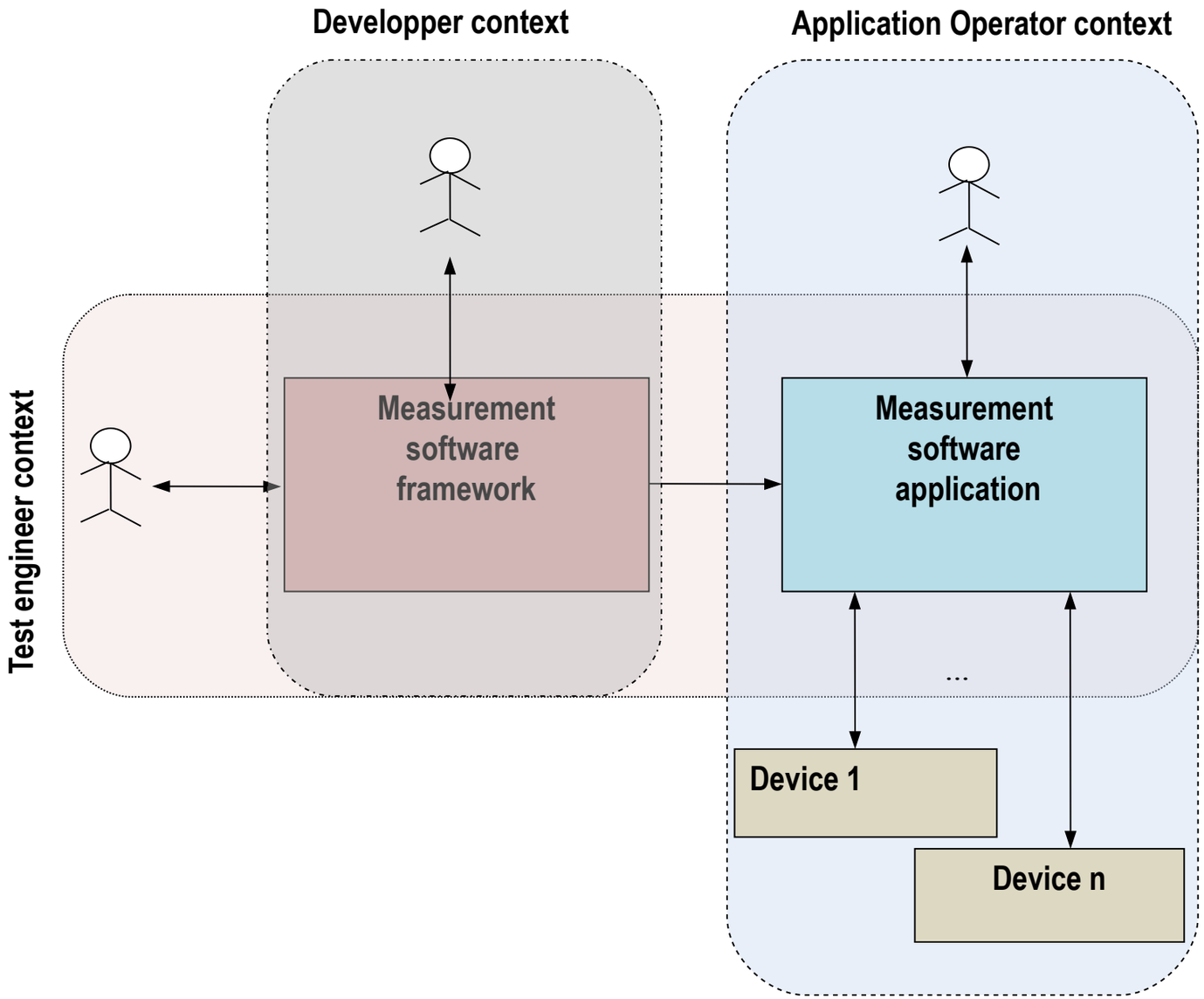
marco.buzio@cern.ch, "FFMM: a Flexible software Framework for Magnetic Measurements"
IMMW 15, International Magnetic Measurement Workshop, FERMILAB, Batavia, IL, 21–24 Aug 2007

- Framework
  - A **software framework**, in computer programming, is an **abstraction** in which common code, providing **generic** functionality, can be selectively **overridden or specialized** by user code for providing *specific functionalities*.

A Flexible Framework for Magnetic Measurements (FFMM)

- Works well with available resources
  - R&D with short term contracts
  - Implemented components stay integrated in framework
- Platform Independent, vendor independent
  - Build on open source software
  - Linux an option
  - Driver implementation for commercial devices optional
- Separation of Users, Test Engineers and Framework Developers

FFMM – Roles


Developper context
Application Operator context
Test engineer context
Measurement software framework
Measurement software application
...
Device 1
Device n

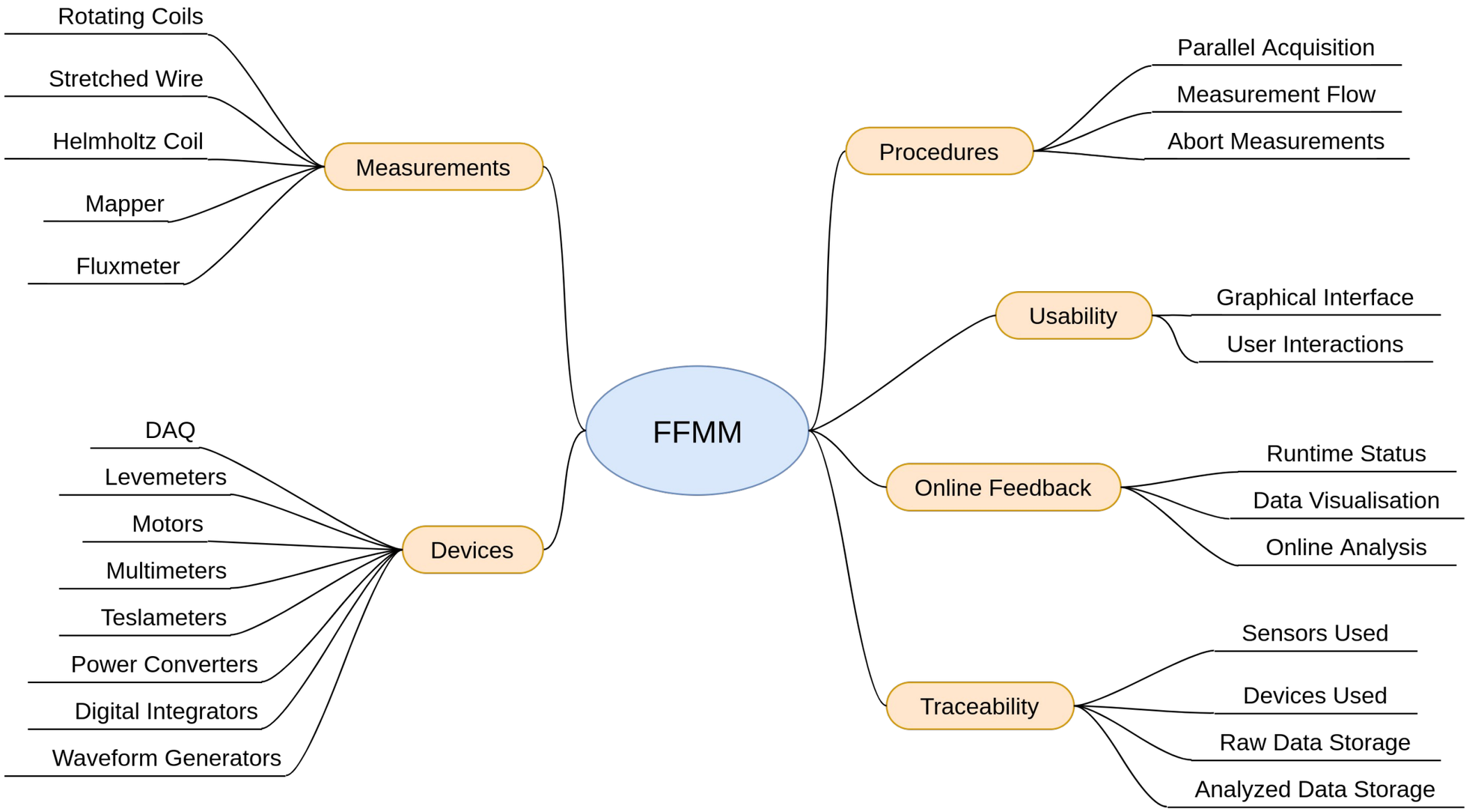
FFMM – Development History

- First idea in 2007
- First implementation for SM18 rotating coil benches
- Extension to all rotating coil benches
- Extension to wire benches (2011)

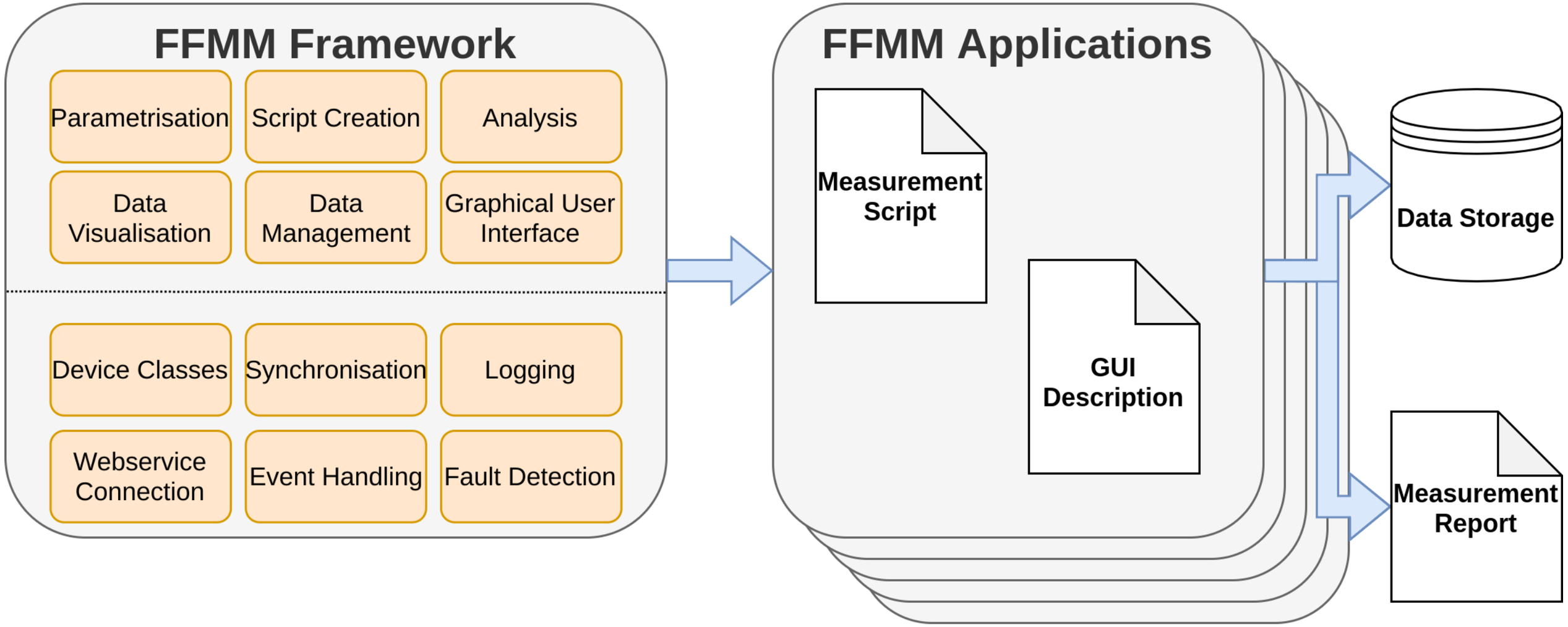
- Coverage all of MM sections platforms
  - Around 40 systems for measurements plus R&D
  - Increased requirements on functionality and features
  - Coverage of functions beyond simple data acquisiton

- Many contributions from initial idea to current state by staff, students, short-term contracts, collaborations
  - Present developers: Matthias Bonora, Lucio Fiscarelli, Carlo Petrone

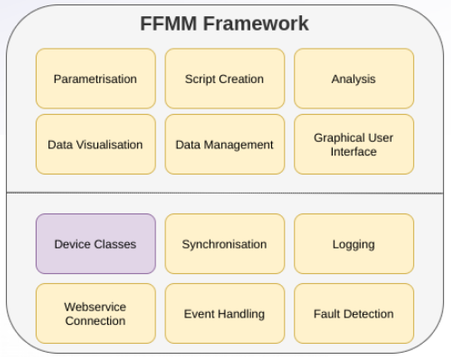
FFMM – Requirements



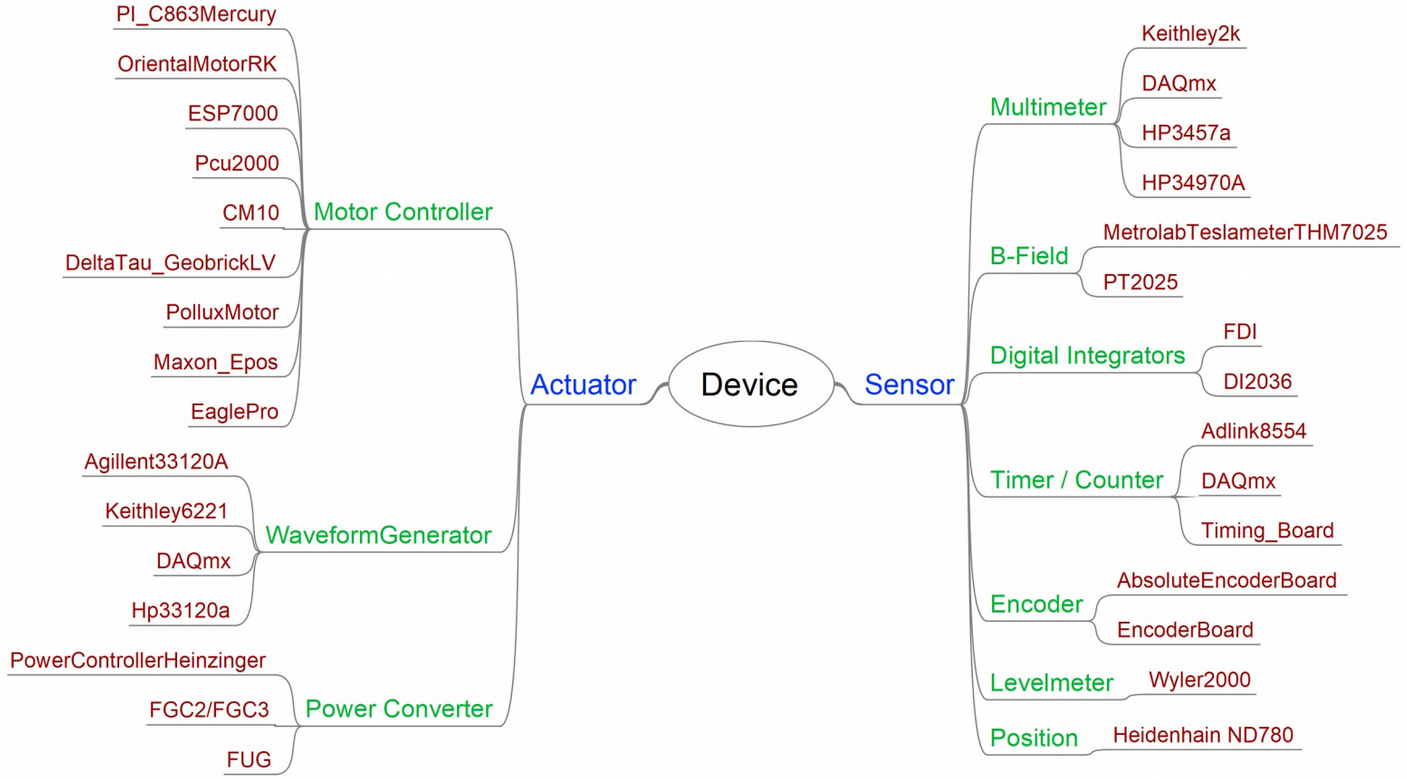
FFMM – Current State of Implementation



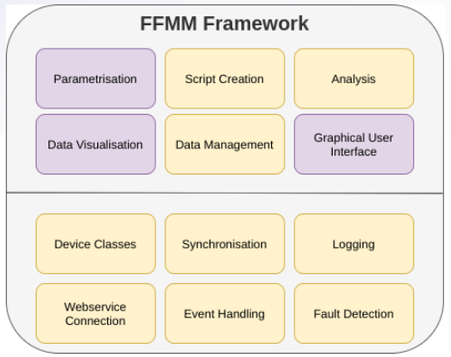
Magnetic Measurements – Devices



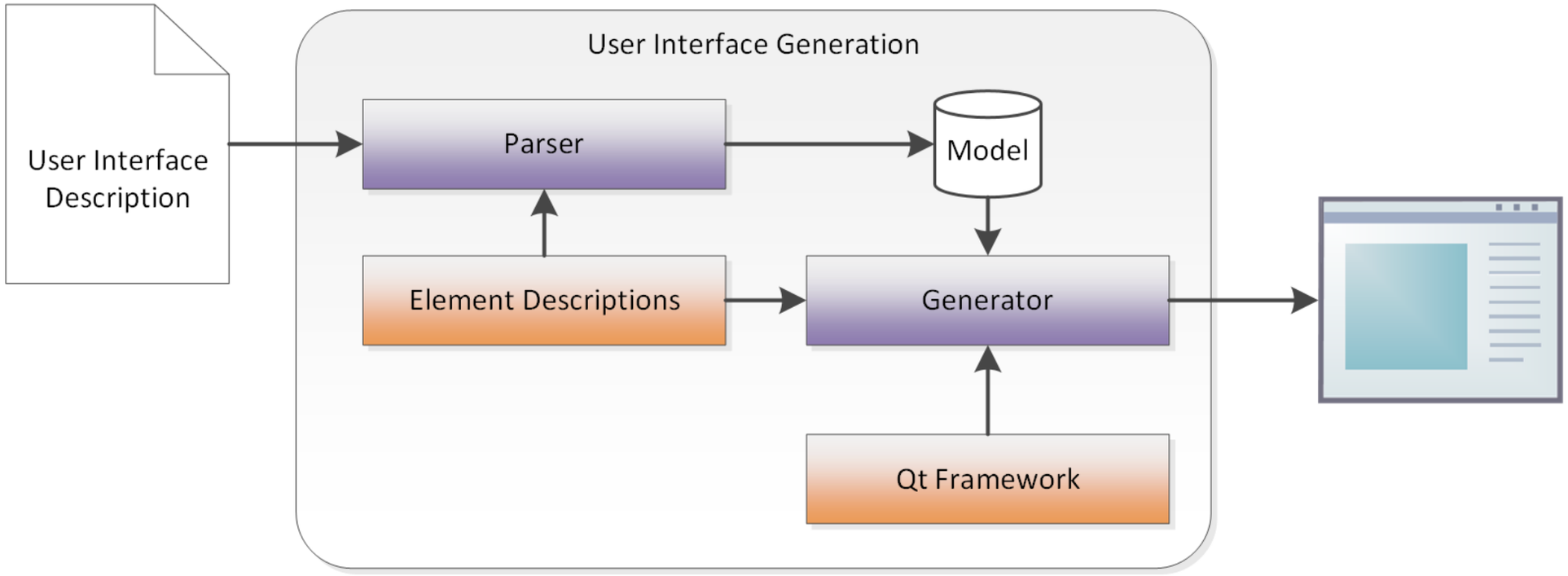
- Implementation of many devices
- Data acquisition
- Multimeters / tesla-meters
- Motor controls
- Power converter controls



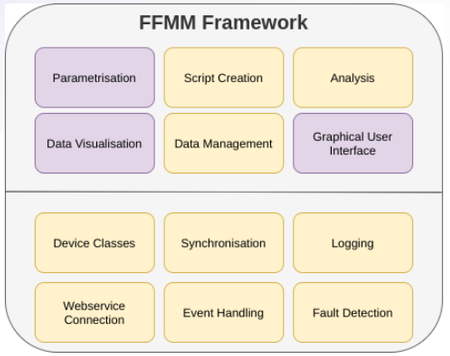
FFMM: Graphical User Interface Generation



- Implementation of a GUI generator
- Description of a user interface in a few lines of text
- Generation of a user interface with parameter settings, plots, and user controls

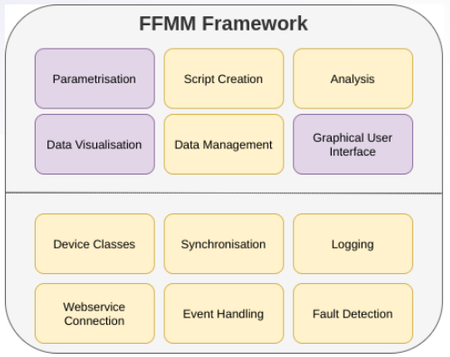


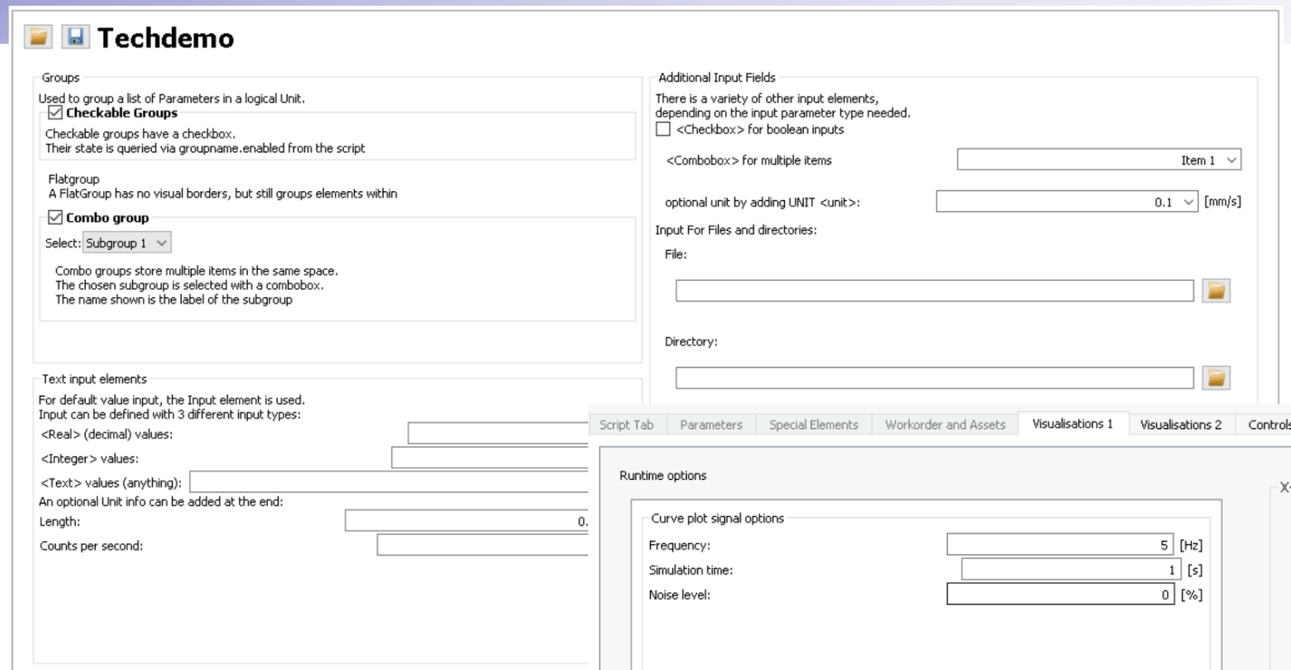
FFMM: Gui Generation Output

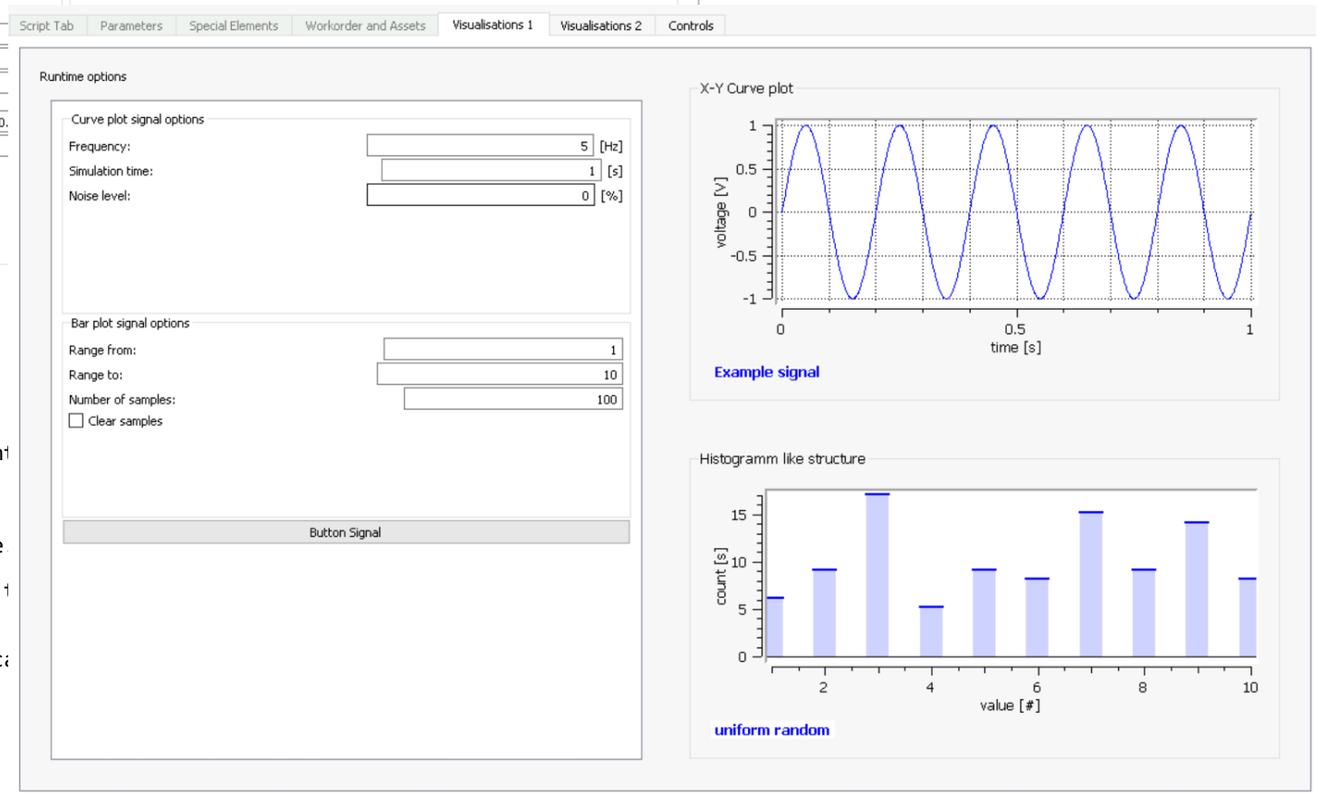


- Inputs for parametrizing a measurement
  - Input fields
  - Grouping of settings
  - Tables
  - Cycles
  - Device settings (ports, locations, types)
  - Measurement parameters
- Simple plots
  - Time series plots for fluxes/currents/DAQ signals
  - Bar plots for histograms and multipoles
  - Status signals
- Inputs for measurement control
  - Dialogues, status messages, dynamic controls

FFMM: Gui Generation Output



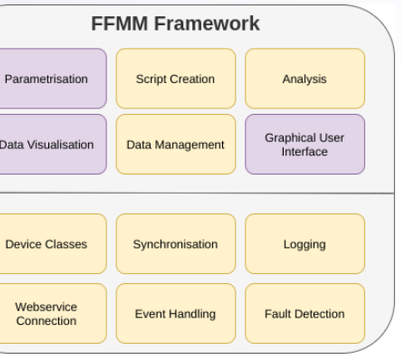


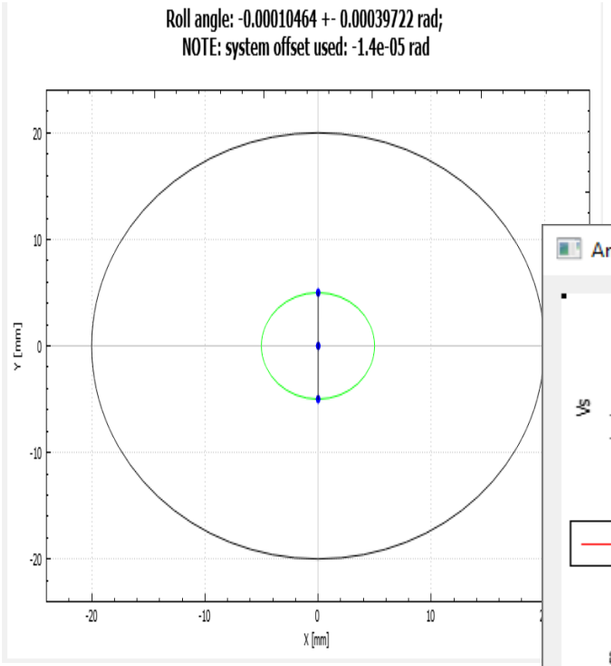


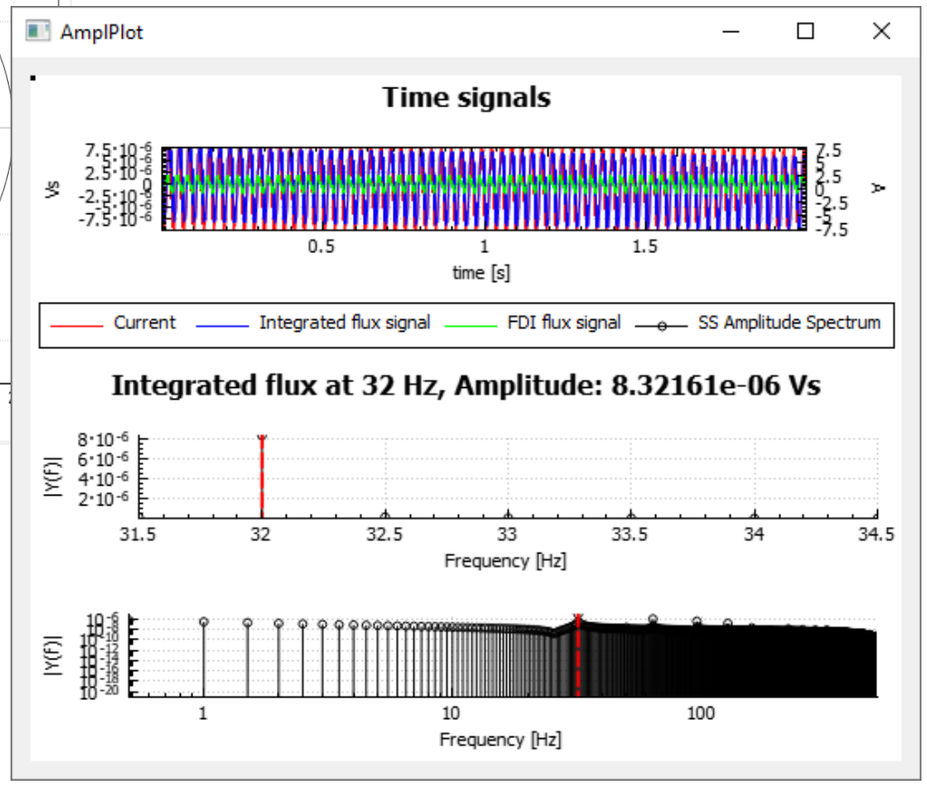

```
#Script = Techdemo 2

grouping = Group Groups
grouping.description = Text "Used to group a l

grouping.check = Group "Checkable Groups" chec
grouping.check.description = Text "Checkable g

grouping.flat = FlatGroup "A flat group"
grouping.flat.name = Text Flatgroup
grouping.flat.description = Text "A FlatGroup has no visual borders, but still groups element

grouping.select = ComboGroup "Combo group" checkable
grouping.select.sub1 = FlatGroup "Subgroup 1"
grouping.select.sub1.description = Text "Combo groups store multiple items in the same space
grouping.select.sub2 = Group "Subgroup 2" checkable
grouping.select.sub2.description = Text "Any group can be a subitem of combogroup.\nTo read t

InputFields = Group "Text input elements"
InputFields.description = Text "For default value input, the Input element is used.\nInput ca
InputFields.reals = Input "<Real> (decimal) values:" Real
InputFields.ints = Input "<Integer> values:" Integer
InputFields.txt = Input "<Text> values (anything):"
InputFields.desUnit = Text "An optional Unit info can be added at the end:"
InputFields.unit = Input Length: Real mm
InputFields.cps = Input "Counts per second:" Integer cps
```

FFMM: Complex Plotting

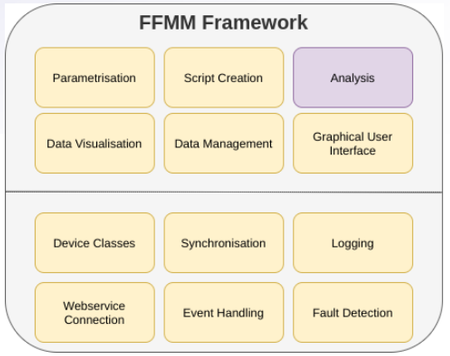
- No limit in plotting functionality
- Simple plots in GUI description
- Complex plots as code within script



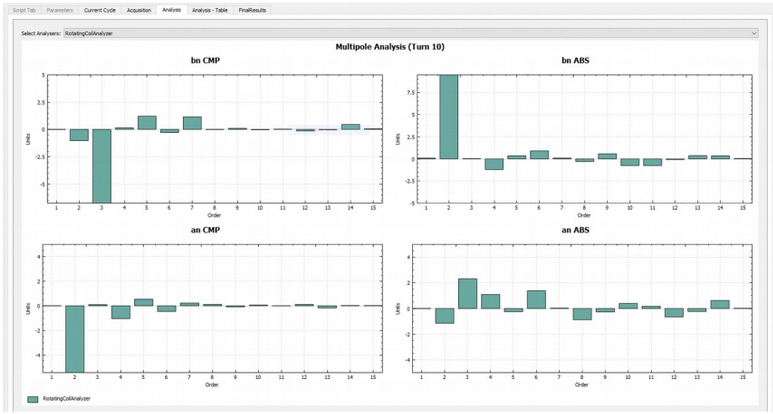




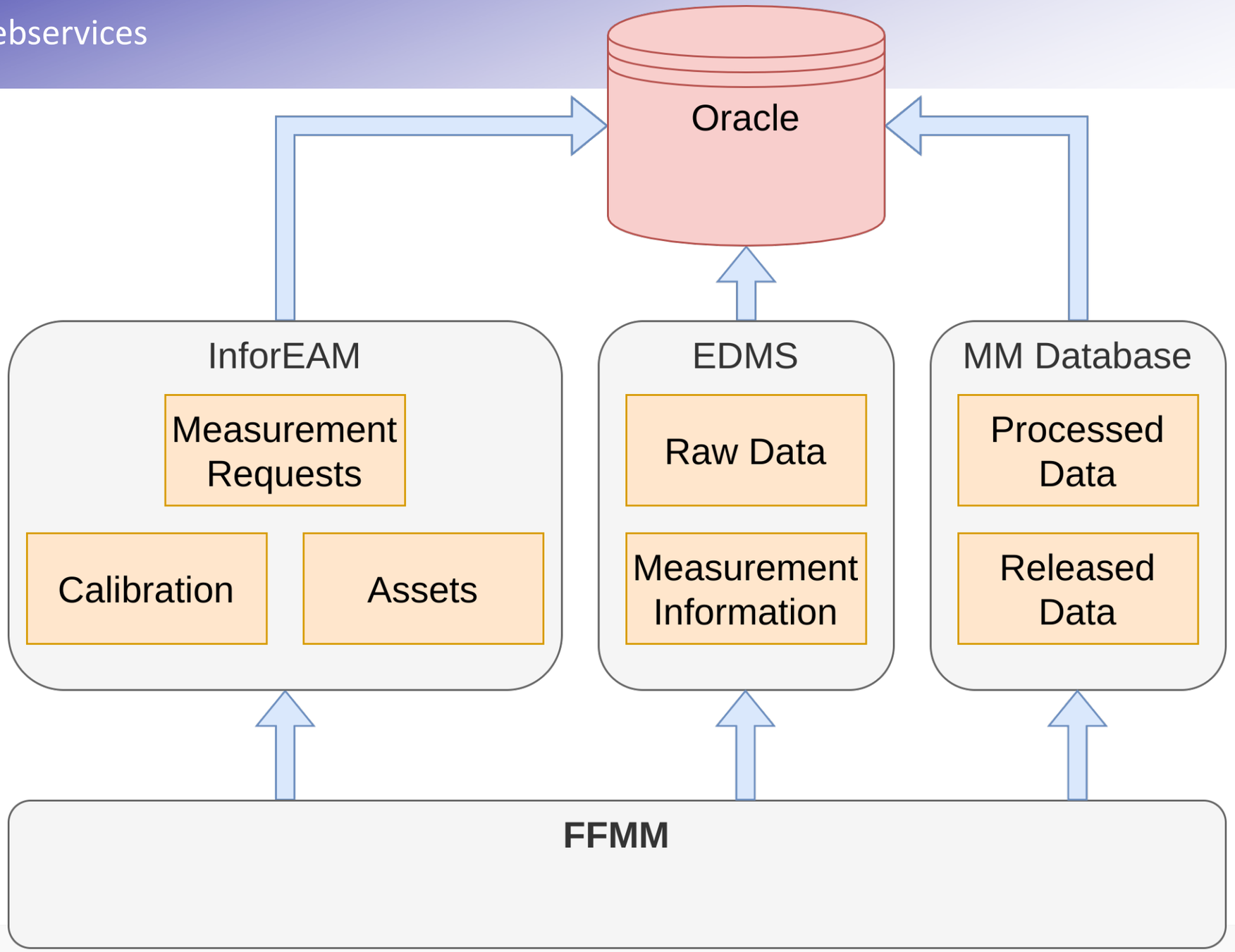
FFMM: Analysis

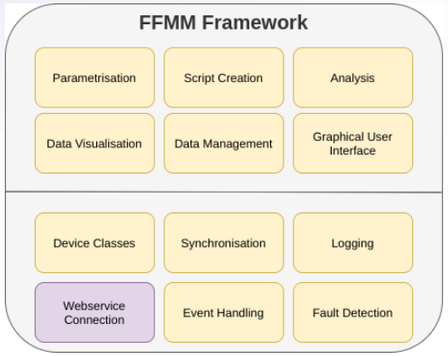


- Provide features to run analysis online during data taking
  - Immediate feedback
  - Early results
- Multiple approaches
  - a) Use Matlab generated code for analysis (compiled C++)
    - Code sharing between offline post-processing scripts and online analysis
    - Quick implementation
  - b) Use native C++ implementation
    - Use library for common needs (linear algebra, solvers, FFT calculation)
    - No dependencies on Matlab

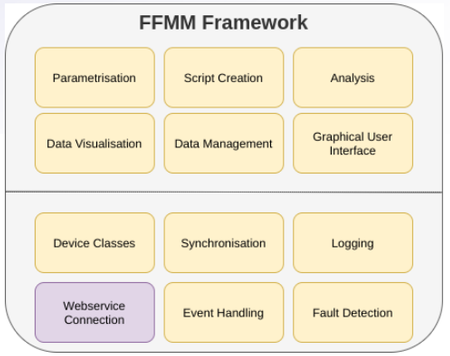


FFMM: Webservices

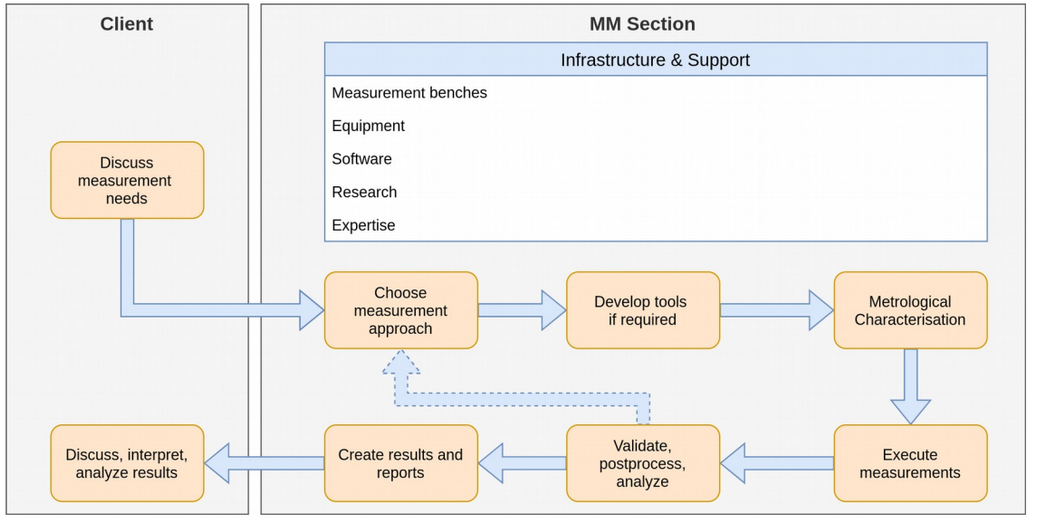




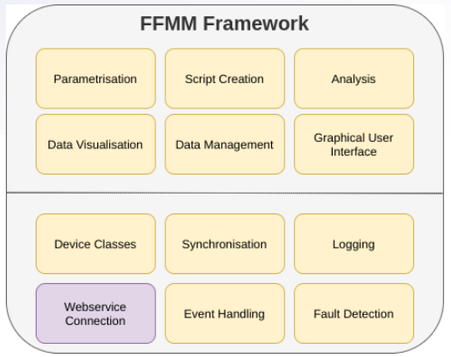
FFMM: InforEAM → Workorders and Assets



- Workorders derive from a client measurement request
  - Track status of a measurement
  - Central point for accessing information and data about a measurement
    - Results, data, execution information, used devices
- Assets represent MM devices and sensors
  - 2000 devices registered
  - Rotating coil shafts, integrators, coils, probes, measurement benches,...
  - Calibration data (including history)
- Both integrated into FFMM
  - Link a measurement to a workorder
  - Track used assets for a measurement
  - Add devices and equipment by barcode scan
  - Access all measurement data through a workorder

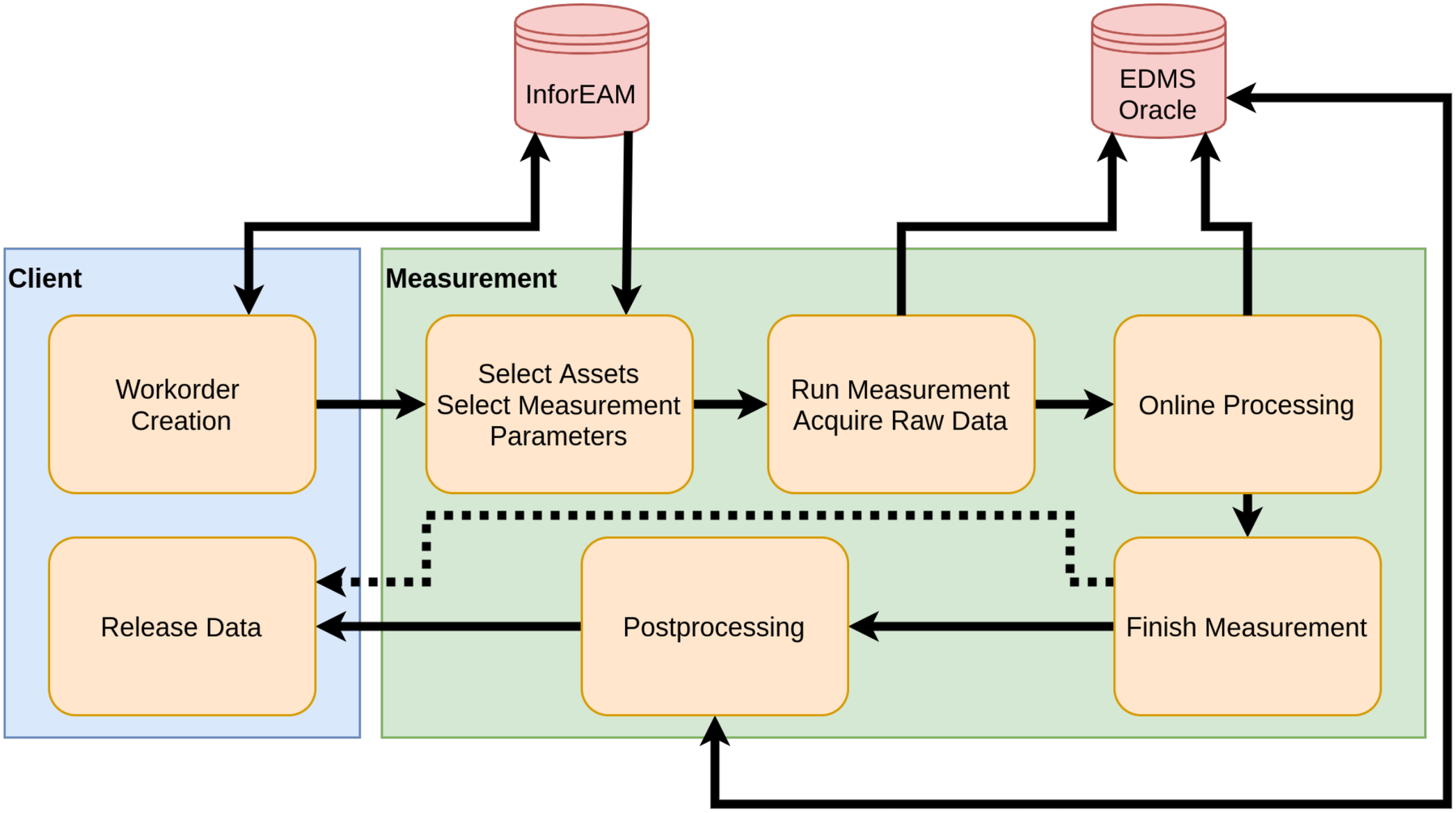


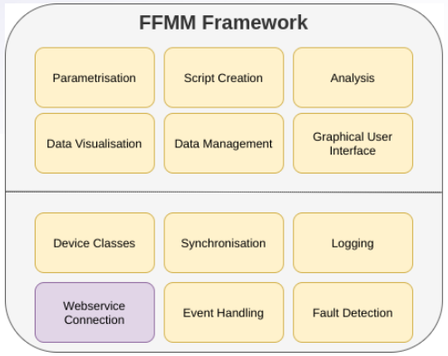
FFMM: Data Storage



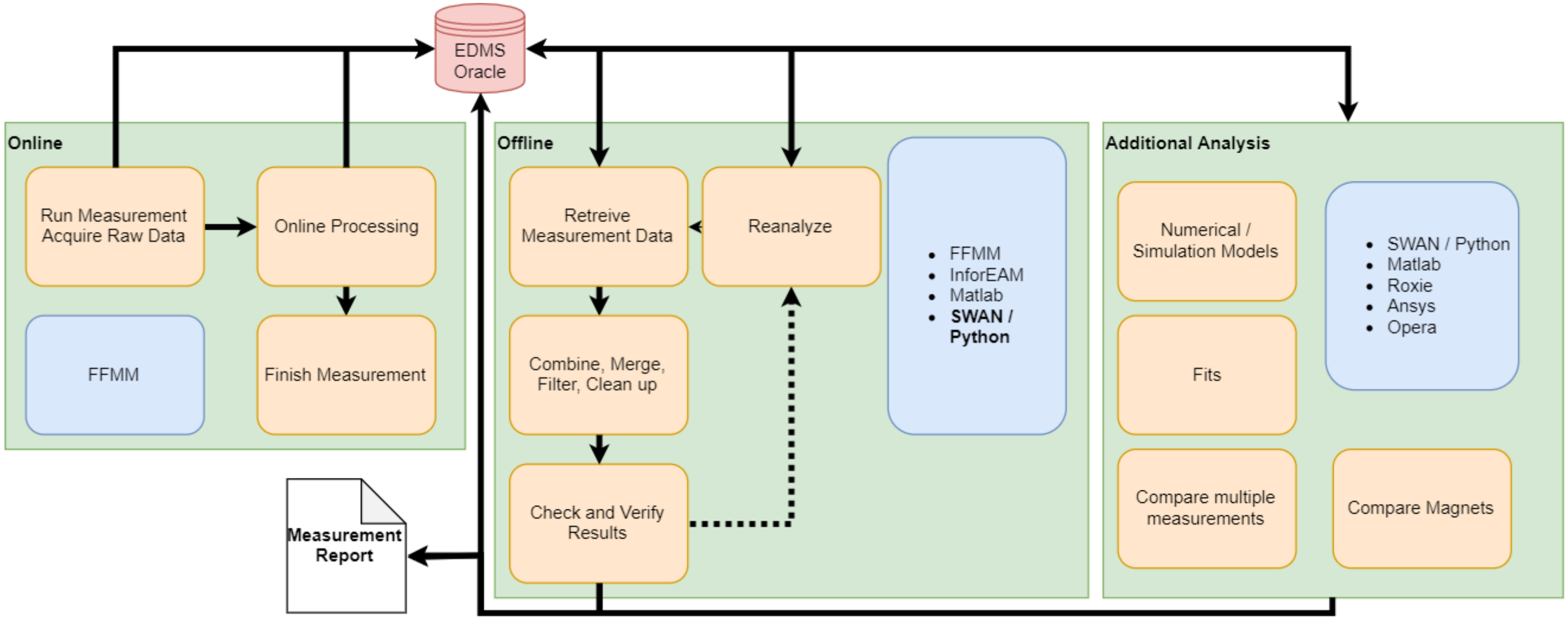
- Combination of InforEAM, EDMS, and custom Oracle database
- Trade-off between ease of access and ease of implementation
  - Simple results into InforEAM
  - Complex outputs into Oracle database
  - Raw data files and measurement parameters into EDMS
- Storage of all measurement parameters
  - Devices in use
  - Settings
  - Calibration data
- No raw data are discarded → reconstruction possible
  - In case of incorrect analysis parameters or errors
  - Future investigation of measurement results

FFMM: Measurement Dataflow





FFMM: Measurement Postprocessing

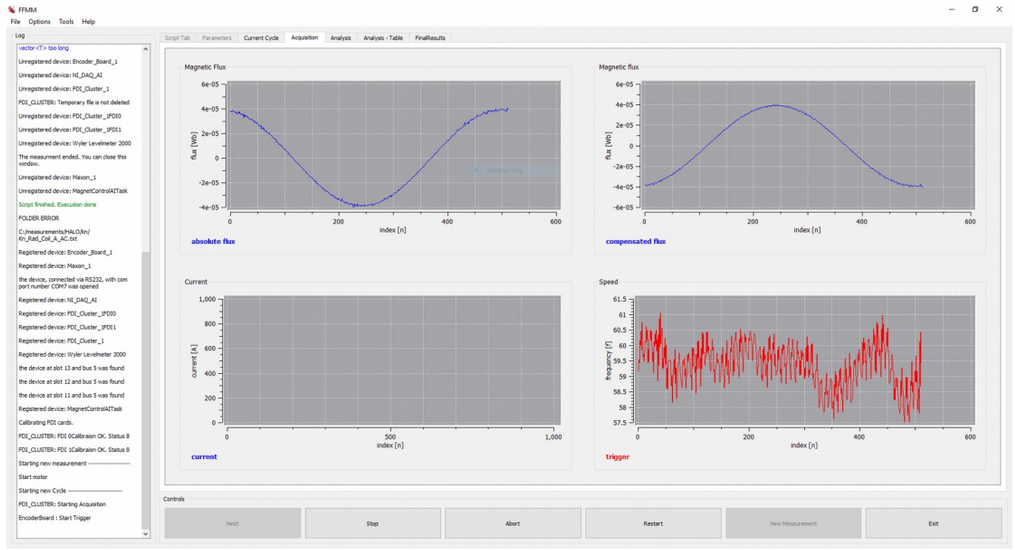


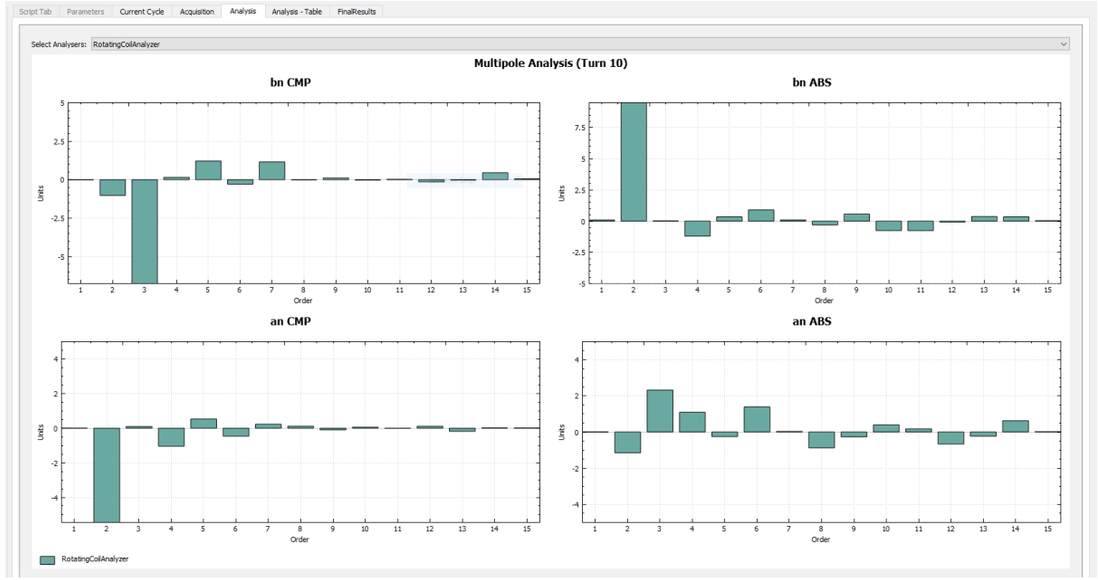
FFMM: Showcase – Rotating Coils in SM18

- Upcoming measurement campaign for HL-LHC
  - High volume in measurements
  - Reduction of feedback loop
- Quality assurance and traceability of measurements

FFMM: Showcase – Rotating Coils in SM18

- Webservice implementation
  - Tracking of used devices and rotating coil shaft
  - Direct loading of calibration data
  - Storage of raw fluxes and processed multipole data
- All data linked to a measurement request
  - Traceability of measurement
- Online analysis for full system
  - Multiple segments, multiple apertures
- Faster access to results for postprocessing
- Immediate feedback of measurement
  - Connection errors show in analysis
  - Early possibility to restart measurement





FFMM: Showcase – Postprocessing

- Postprocessing with Python/Jupyter scripts
  - Retrieve data by measurement request
  - Perform data cleanup and checks on measurements
  - Generate plots, tables, **release data into database**
- Reusable
  - Generic template with parameters
  - Customized template for specific magnets
  - Common core classes for processing, plotting and database access
- Traceable
  - Common code in version control
  - Applied transformations stored in script on EDMS
  - released data linked to EDMS document

Postprocessing Rotating Coil Measurements

Postprocessing step to check, clean and Release Rotating Coil Measurements

Parameters

Selection of Run

There are multiple ways of selecting a Measurement

- parentWO: Measurement Request Workorder
- woNumbers: Run Workorder numbers (Rotating Coil measurement runs)
- magnetNames: Magnet asset names
- edmsIdsSel: Directly via a list of Run Analysis edms ids

In [4]: removeinput

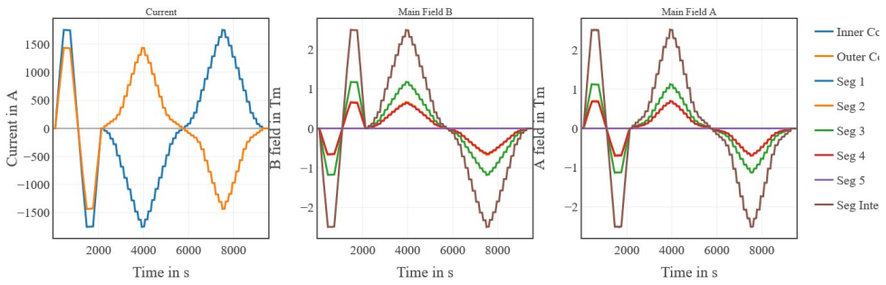
```
display(Markdown("""### Version information
tm_analysis_tools package version: {0}
""".format(tm_analysis_tools.__version__)))
```

Version information

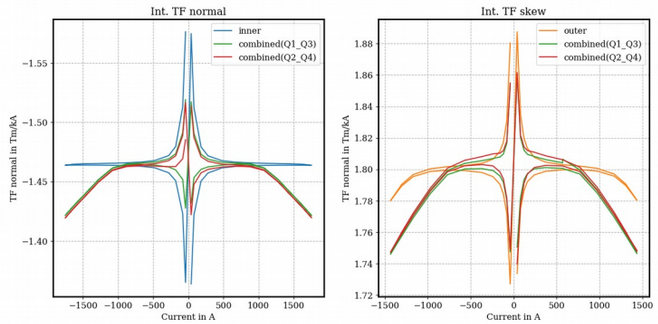
tm_analysis_tools package version: 2021.8.14.dev19+g6ccdc09

In [5]: parameters

```
parentWO = 29849814
woNumbers = None
magnetNames = None

# Enter EDMS number
edmsIdsSel = None

run_names = {}
FDI_cal = 1.0000
LMagnet = 1.2000

# List of relevant segments, or NONE if all segments should be kept
relevant_segments = None


interested_poles = range(2,16)

segment_lengths_override = None
segment_zpos_override = None

#L = 253.88*1e-3*0.9975
#G = (255-253.88+1)*1e-3*0.9975
#segment_lengths_override = {'{0}_{1}'.format(i,b).format(i) : L for i in range(1,6) for b in {'A','E'}}
#segment_zpos_override = {'{0}_{1}'.format(i,b).format(i) : 0 for i in range(1,6) for b in ['A','E']}
#segment_zpos_override['1_A'] = -(L+G)
```



	Unit	Value
I	A	1754.9999
TF	Tm/kA	1.4593
Integrated Strength	Tm	2.5610



n	bn	an
1	10000.000	-0.039
2	-0.344	-1.483
3	-11.075	0.841
4	-0.074	1.032
5	-6.702	0.656
6	0.021	0.459
7	-3.475	0.456
8	-0.207	0.043
9	-0.147	-0.087
10	-0.142	0.123
11	1.757	-0.340
12	0.162	-0.097
13	-1.523	0.423
14	-0.026	-0.001
15	0.147	-0.030

FFMM: Summary of current state

- FFMM now established baseline for magnetic measurement scripts
- Continuous development of new features and integrations
- Integration of features beyond simple data acquisition
- An effective way of retaining implementations and R&D
  - Growing library of devices, components and features
  - R&D effort on a single script automatically integrated in framework
    - Mostly from students and short-term personnel

- What's next?

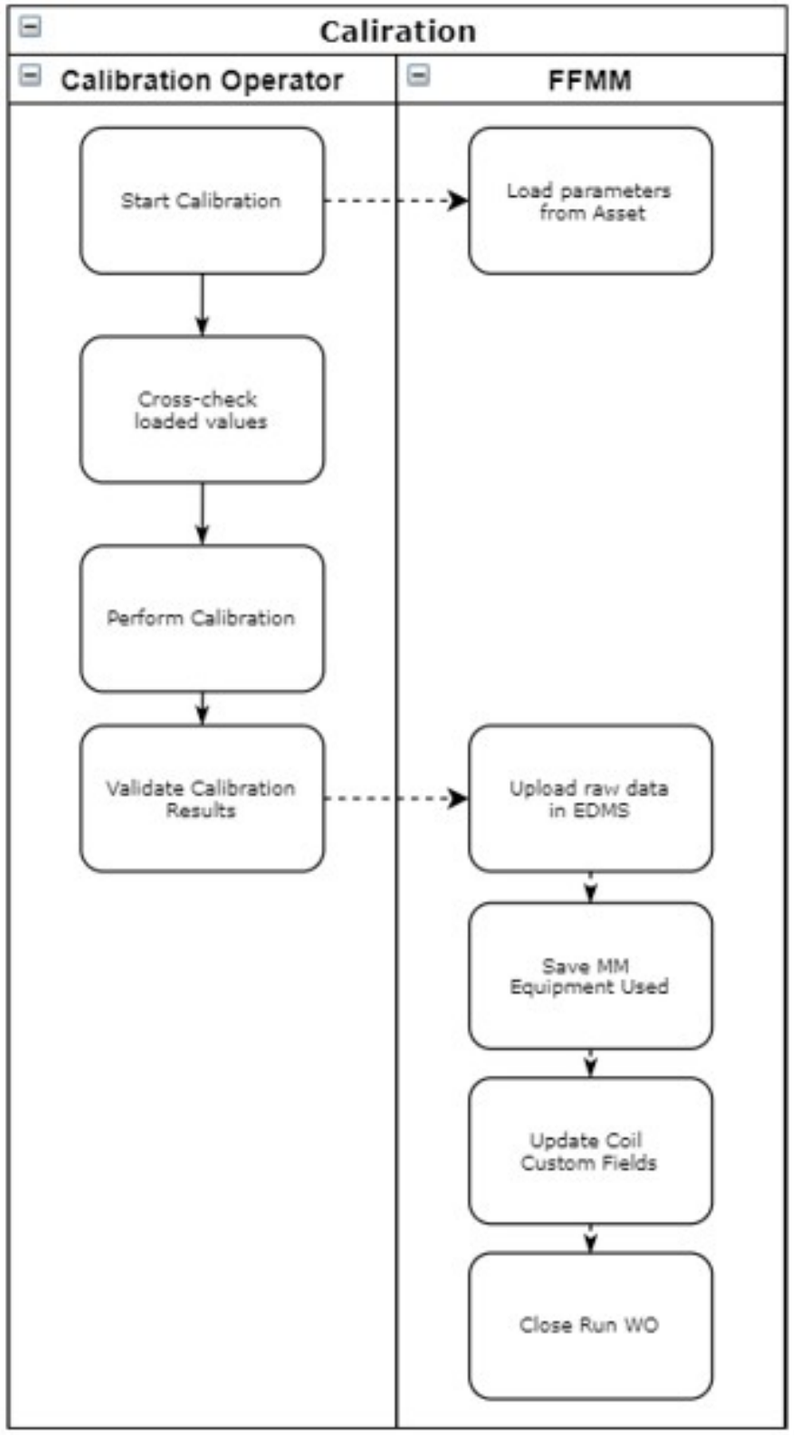
FFMM: Beyond Measurements

- Extend focus of FFMM beyond magnetic measurements
- For any workflow using basic data acquisition or object manipulation
- Framework provides
  - GUI
  - Visualisation
  - Device implementations
  - Access to InforEAM
    - Workorder access
    - Asset tracking
  - Access to EDMS
    - Store data
    - Load documents
  - Access to Oracle database

FFMM: Beyond Measurements - Calibration

- In-house calibration of coils and arrays
- Coils, arrays and PCBs added as assets to system
- Operator loads request for calibration in FFMM
- Operator performs calibration in FFMM
- FFMM uploads calibration values to InforEAM
- FFMM uses uploaded calibration values

- Example of script using simple daq + webservice features
- Mutual benefits from FFMM framework implementation
- Same concept easily adaptable to other domains wherever needed
  - FGC control (Current Cycles, pulsing, degaussing)

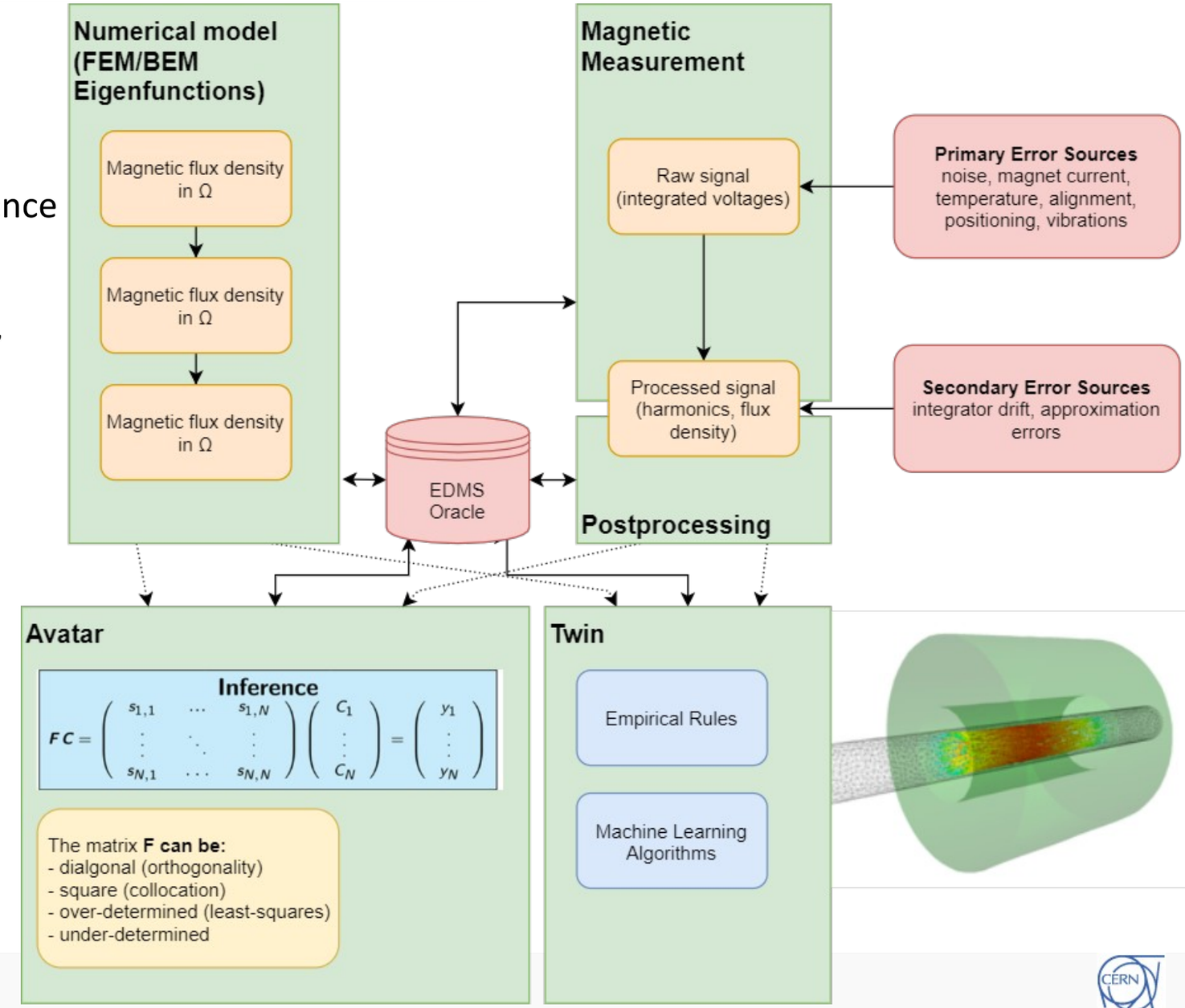


FFMM: Beyond CERN

- Increasing Requests for building measurement systems for external entities (INFN, CEA)
  - Building and selling measurement systems not within CERN mandate
- Concept: Measurement platforms based on commercial systems
  - Rotating coil platform (PCB coils, commercial motors, commercial DAQ)
  - Stretched wire systems (commercial stages, commercial DAQ)
  - Measurement bench based on commercial products
  - Provide a material list, technical drawings and instructions for setting up measurement benches
- FFMM as measurement software
  - Open source libraries
  - Base for collaboration and extensions
  - No dependencies on platform beyond device choice
- Provide our expertise as service (collaborations)

FFMM: Beyond Acquisition

- Combine Models and Measurements
  - Avatar: Measurement,Simulation,Inference Based on Kirchhoff's theorem
  - Twin: Fusion of measurements, models, empirical rules, machine learning
- Track Avatars, Twins and applied algorithms in database
- Improve models with measurement data
- Better understanding of the magnet
- Feedback for new transducers or measurements



FFMM: Conclusion

- Universal Magnetic Measurement Platform
  - Covers majority of MM needs
  - Separation of roles keeps development ongoing and operation simple
  - Fully integrates with CERN webservices
  - Results stored and ready for further development
- Potential Extension of Scope
  - Use outside CERN
  - Use beyond magnetic measurements