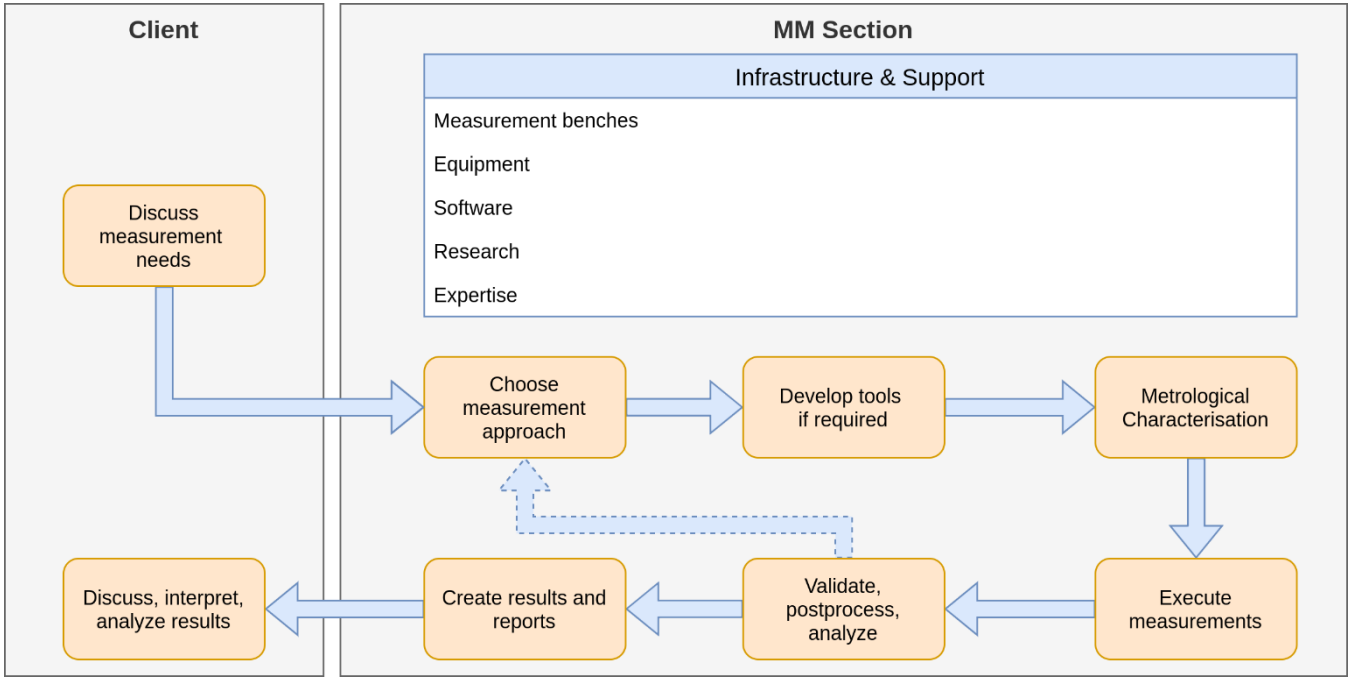
Magnetic Measurement Framework for Fully Traceable Measurements

Matthias Bonora, Marco Buzio, Lucio Fiscarelli, Carlo Petrone, Stephan Russenschuck

Outline

- MM needs systems
- Measurement flow
- Flexible Framework for Magnetic Measurements
- Processing
- Analysis / Results

MM Value Shop



Magnetic Measurement Systems

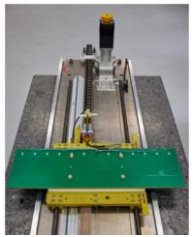
Rotating coil systems

Ring-Sample Permeameter

Helmholtz coils

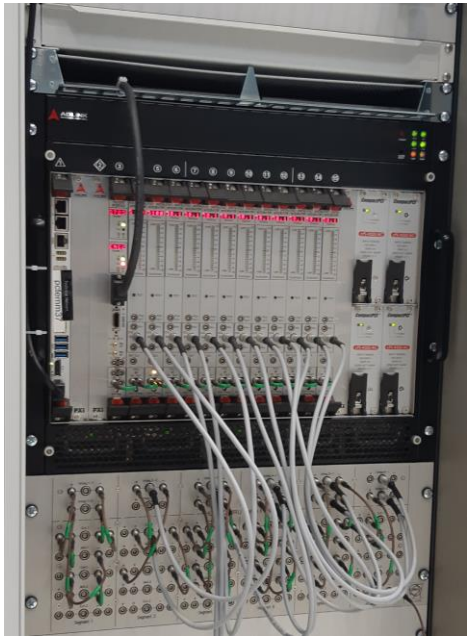
3D mapper

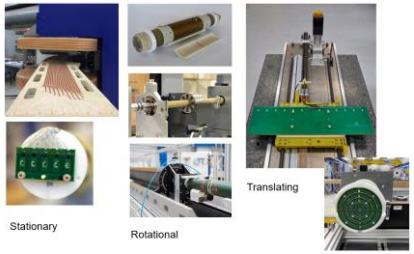
Translating Fluxmeter

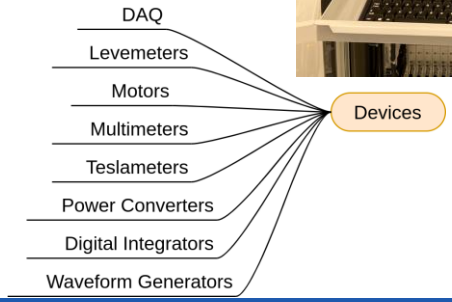


Stretched wire systems

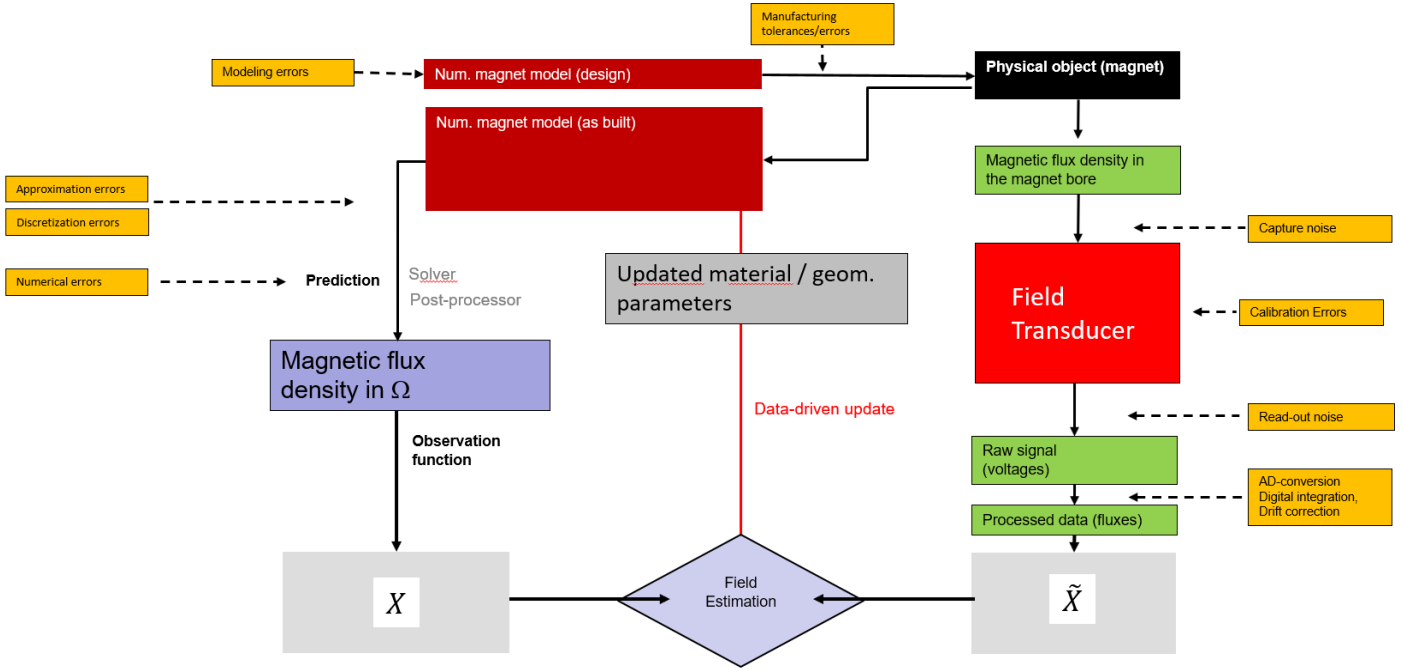
Magnetic Measurement Systems





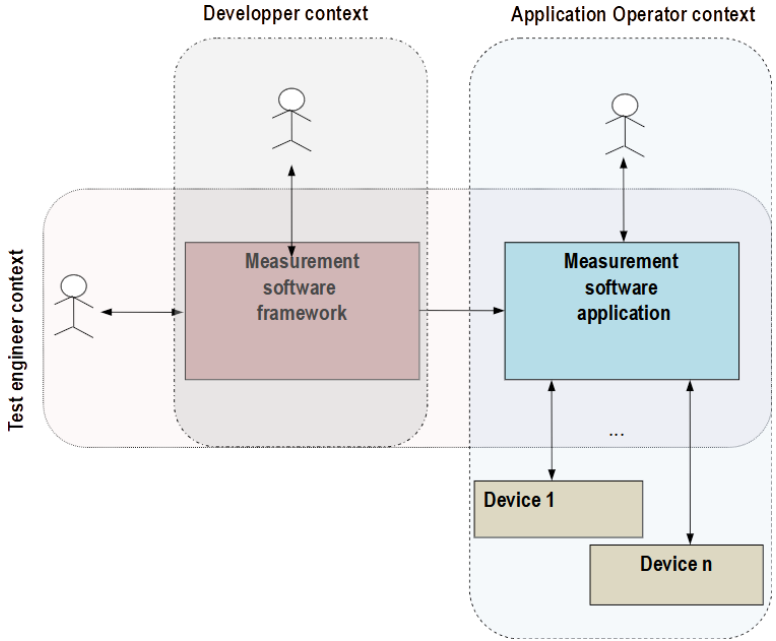


Merge Measurements and Simulations



Magnetic Measurement Roles

- Different roles
  - Development
  - Script development
  - Operation



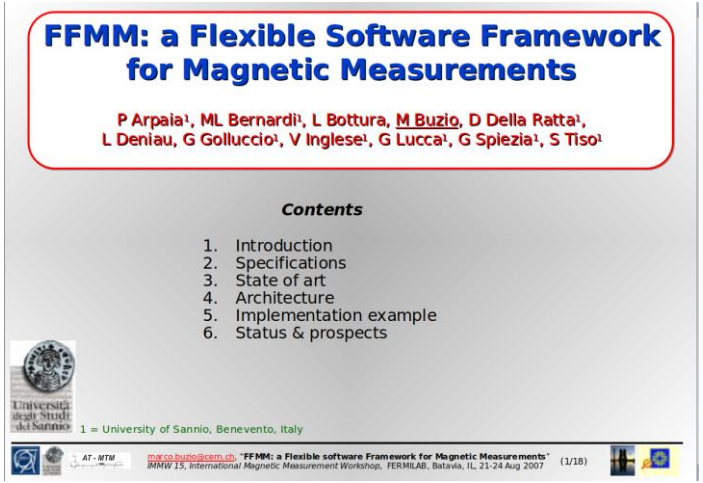
MM Software Requirements

- Different measurement benches and types
- Similar, yet different acquisition systems
- Different sensors and actuators
- Different actors of usage
- Measurement data storage and retrieval
- Postprocessing and analysis

- → Need for an efficient software platform as base of operations

A Flexible Framework for Magnetic Measurements

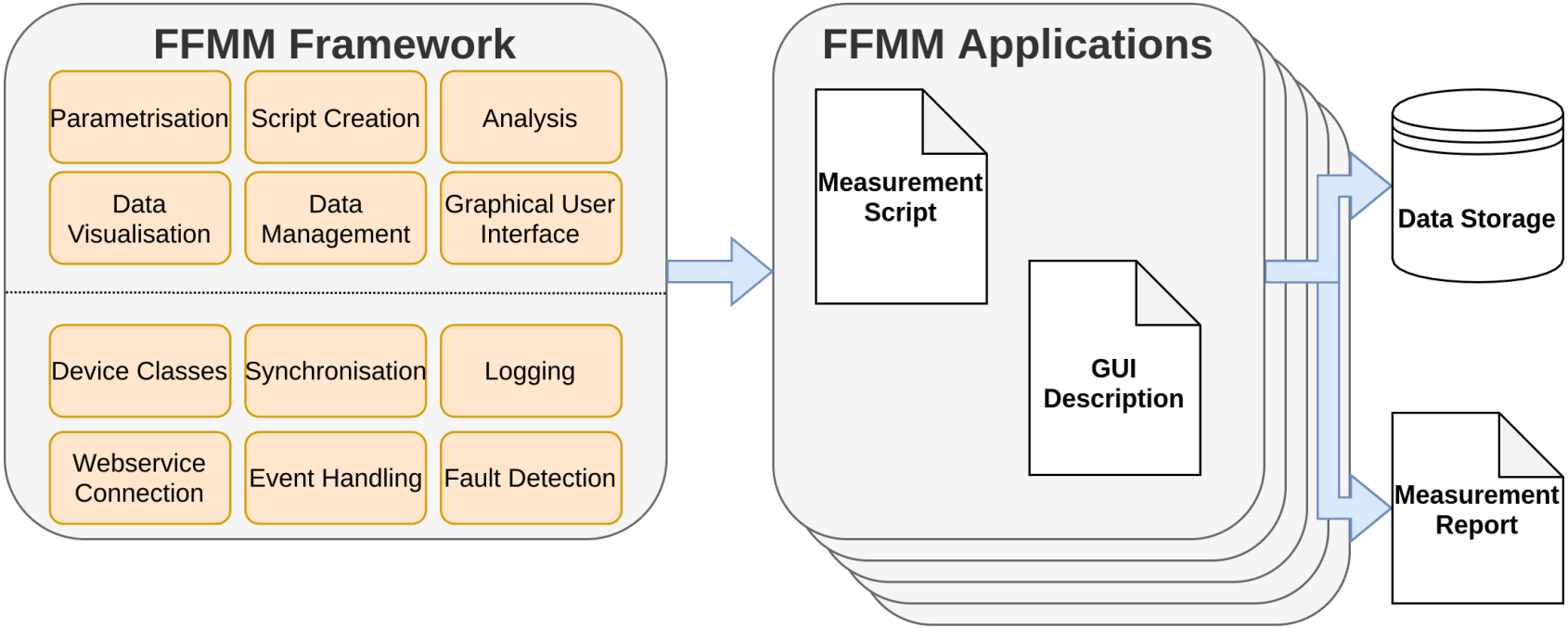
- Idea for a software framework for magnetic measurements
- Reusable
  - Easy to implement small independent blocks
  - Blocks are reusable when needed
- Flexible
  - Easy to write measurement applications
  - Still full control and extendibility



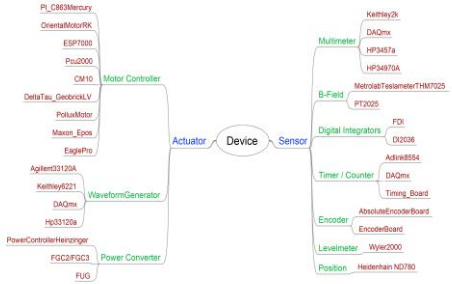
marco.buzio@cern.ch, "**FFMM: a Flexible software Framework for Magnetic Measurements**"
IMMW 15, International Magnetic Measurement Workshop, FERMILAB, Batavia, IL, 21–24 Aug 2007

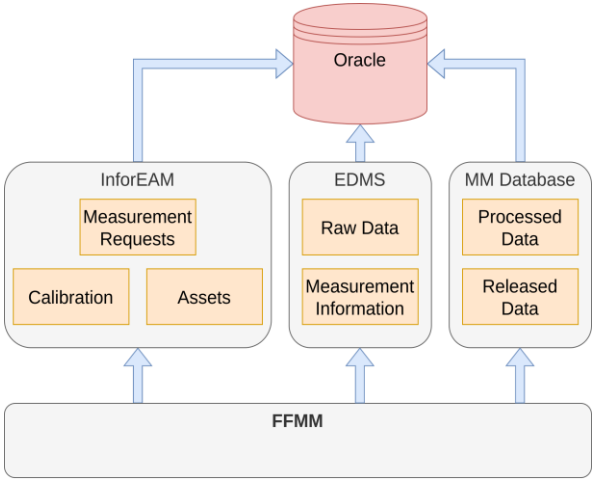
A software framework, in computer programming, is an abstraction in which common code, providing generic functionality, can be selectively overridden or specialized by user code for providing specific functionalities.

A Flexible Framework for Magnetic Measurements

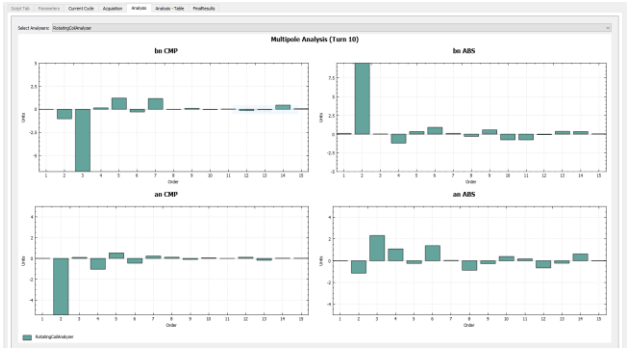


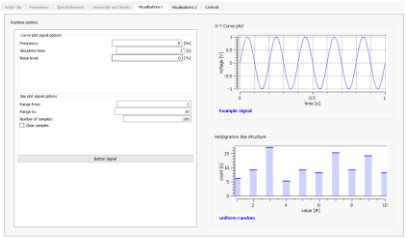
FFMM: Components

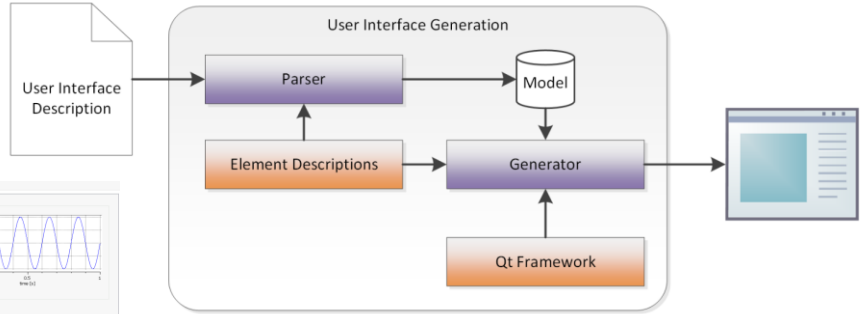




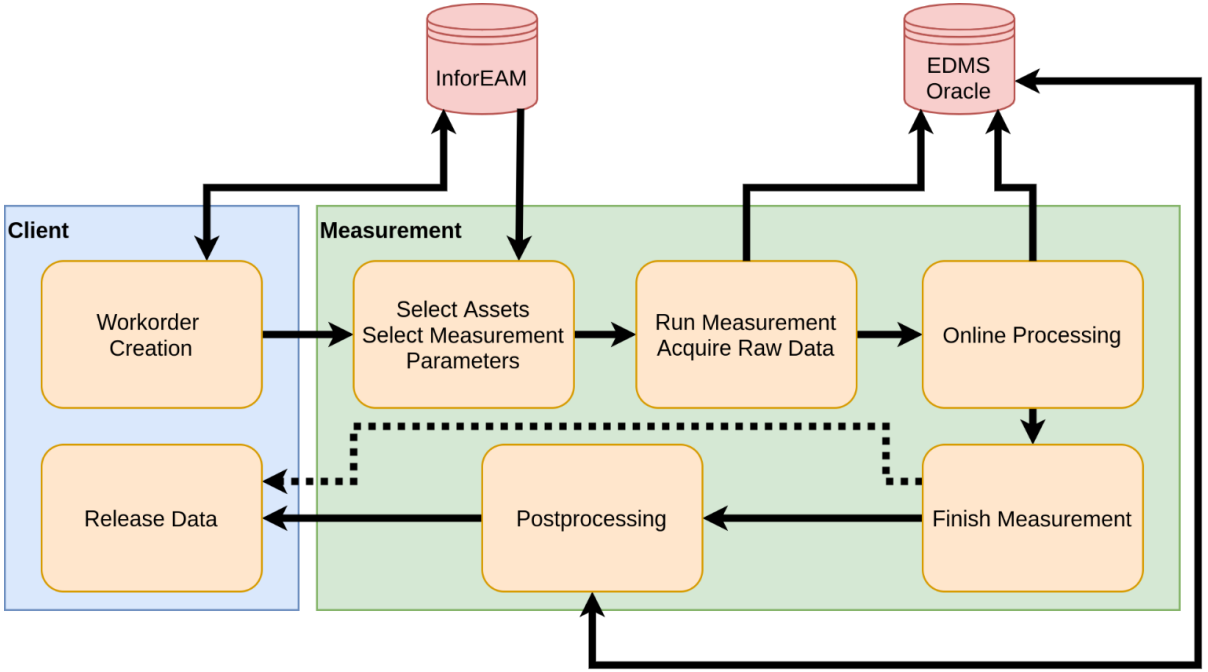

```
#Script = Techdemo 2

grouping = Group Groups
grouping.description = Text "Used to group a list of Par

grouping.check = Group "Checkable Groups" checkable
grouping.check.description = Text "Checkable groups have

grouping.flat = FlatGroup "A flat group"
grouping.flat.name = Text Flatgroup
grouping.flat.description = Text "A FlatGroup has no vis

grouping.select = ComboGroup "Combo group" checkable
grouping.select.sub1 = FlatGroup "Subgroup 1"
grouping.select.sub1.description = Text "Combo groups st
grouping.select.sub2 = Group "Subgroup 2" checkable
grouping.select.sub2.description = Text "Any group can b

InputFields = Group "Text input elements"
InputFields.description = Text "For default value input
InputFields.reals = Input "<Real> (decimal) values:" Rea
InputFields.ints = Input "<Integer> values:" Integer
InputFields.txt = Input "<Text> values (anything):"
InputFields.desUnit = Text "An optional Unit info can be
InputFields.unit = Input Length: Real mm
InputFields.cps = Input "Counts per second:" Integer cps
```



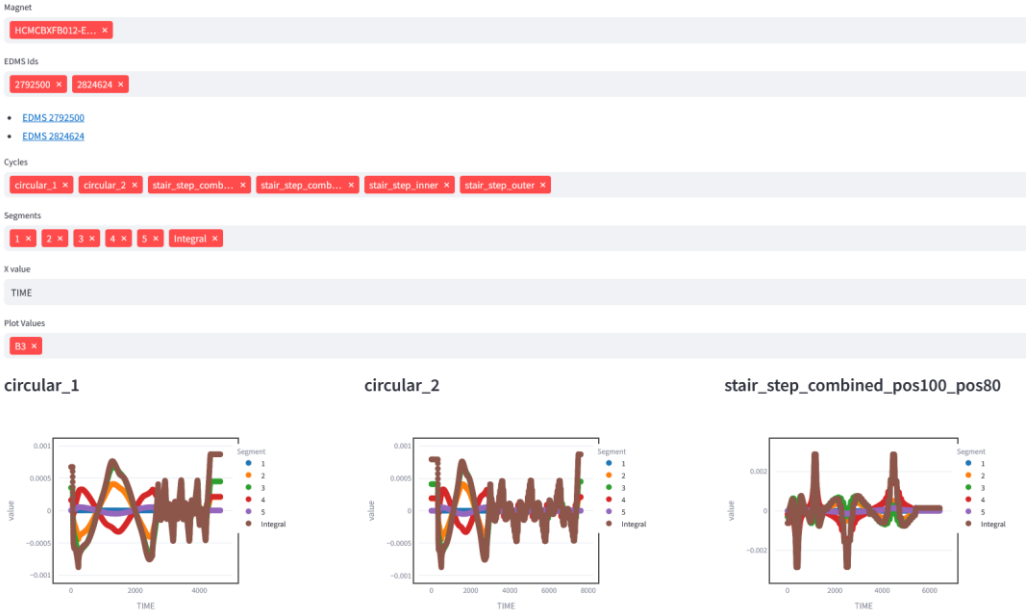


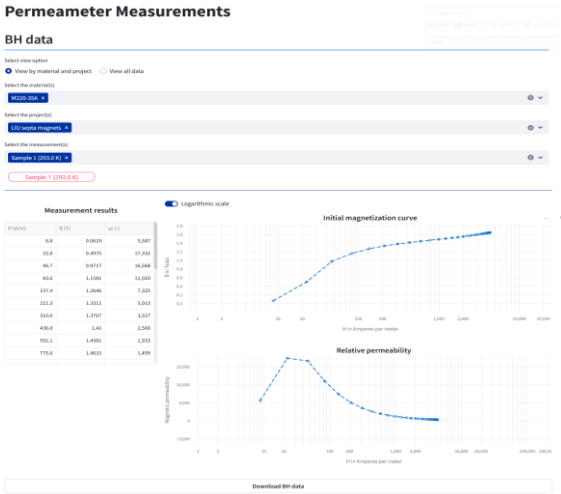


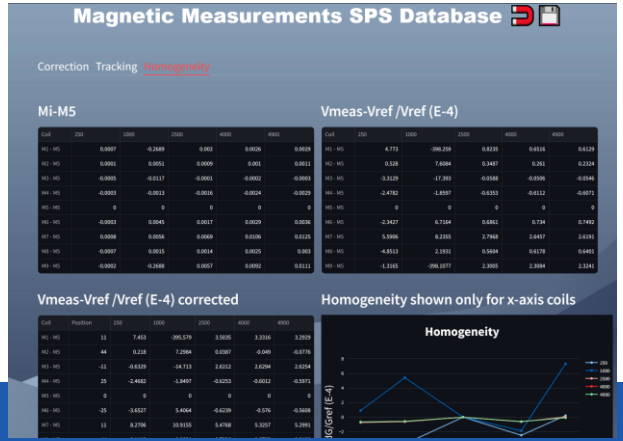
FFMM Measurement Workflow and Data Storage



Access to Released Data

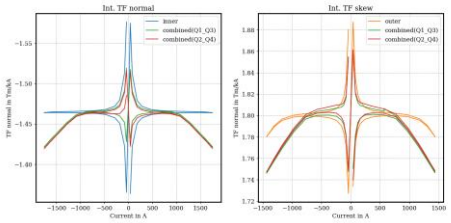






FFMM Postprocessing

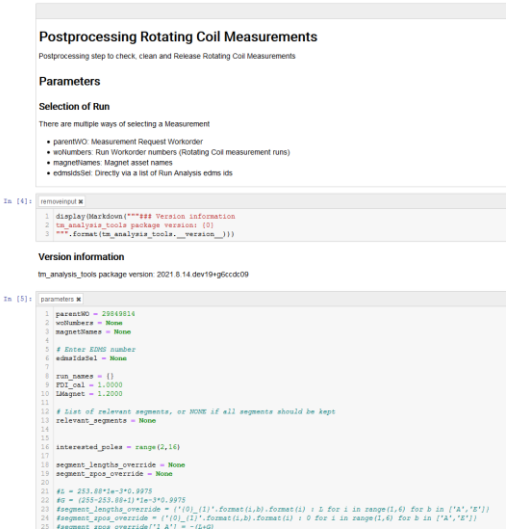
- Postprocessing with Python/Jupyter scripts
  - Retrieve data by measurement request
  - Perform data cleanup and checks on measurements
  - Generate plots, tables, release data into database
- Reusable
  - Generic template with parameters
  - Customized template for specific magnets
  - Common core 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

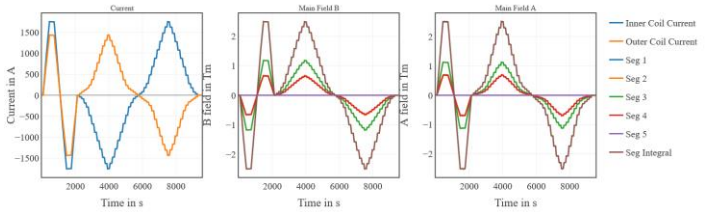


Current: 1755 A

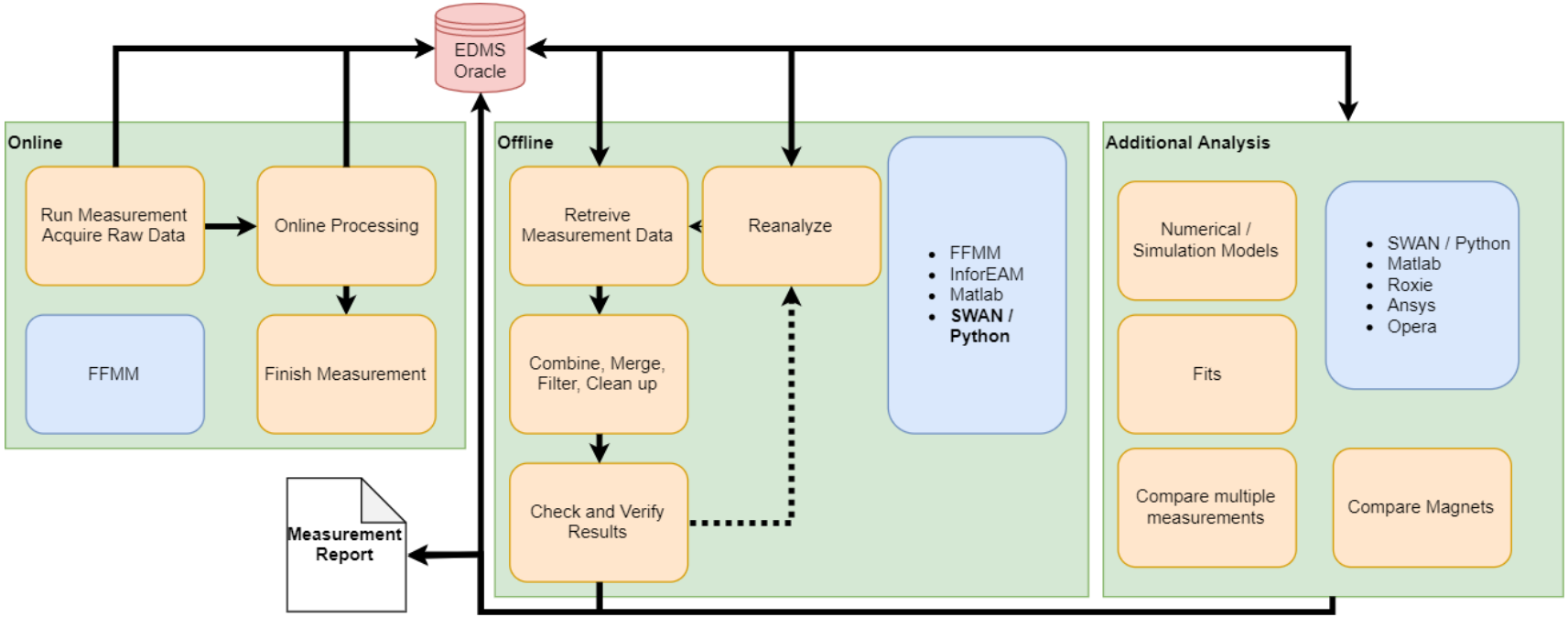
	Unit	Value
I	A	1754.9999
TF	Tm/kA	1.4593
Integrated Strength	Tm	2.5610

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





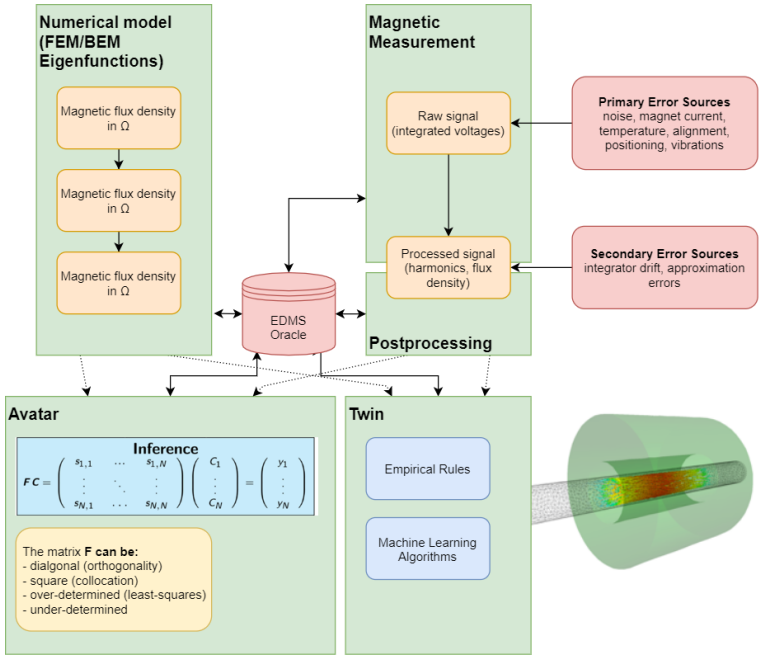
FFMM Postprocessing Workflow



Further Postprocessing

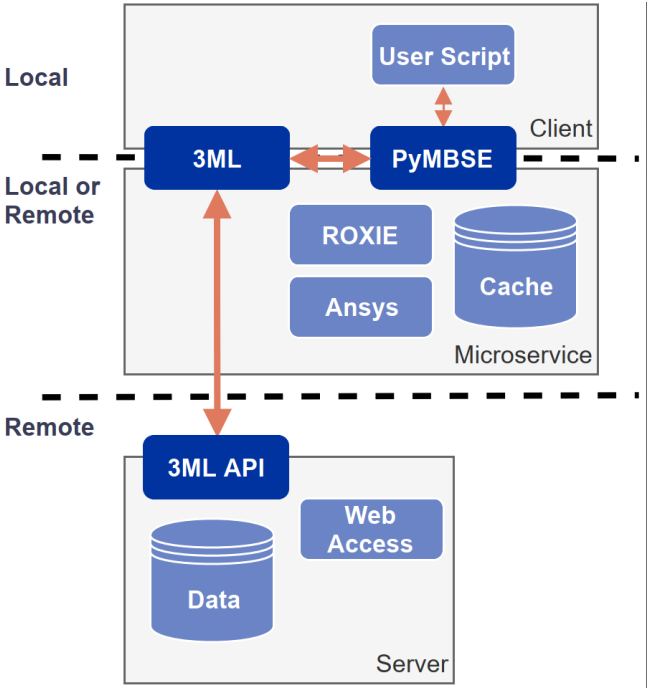
- Combine Models and Measurements
  - Avatar: measurement, simulation, inference based on Kirchhoff's theorem
  - Twin: Fusion of measurements, models, empirical rules, machine learning

- Track Avatars, Twins and applied algorithms
- in database
- Improve models with measurement data
- Better understanding of the magnet
- Feedback for new transducers
- or measurements

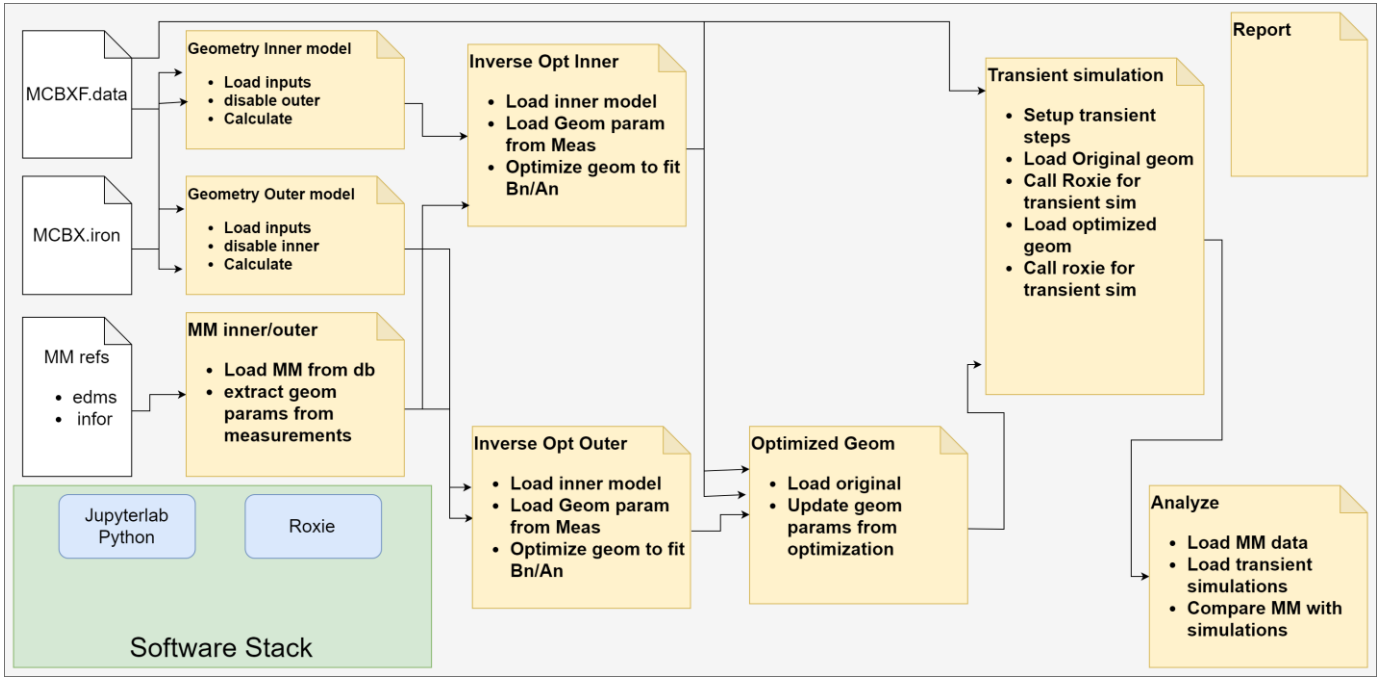


3ML – Magnetic Model Management Layer

- MBSE
  - Measurement models
  - Simulation models
  - Dependencies
  - Automated workflow
  - Result queries



3ML – Measurement Based Geometry Update



Conclusion

- Framework to build measurement applications
  - Built-in traceability and data storage
  - Measurement, postprocessing, analysis
- Integration in CERN IT infrastructure
- Link with simulations and models


CERN