DONDERS INSTITUTE

Radboud University

IN·DEI·NOMINE·FELICITER

Introduction to EEG, MEG and analysis with the FieldTrip toolbox

Robert Oostenveld

Donders Institute, Radboud University, Nijmegen, NL
NatMEG, Karolinska Institute, Stockholm, SE

KAROLINSKA INSTITUTET ANNO 1810

Karolinska Institutet

NatMEG

What is FieldTrip

a MATLAB toolbox for the analysis of MEG, EEG and animal electrophysiology data

can import data from many different file formats

contains algorithms for spectral analysis, source reconstruction, statistics, connectivity, …

Talk outline

What kind of signals are generated in the brain

How do we record those signals

Analyzing those signals with FieldTrip

Background on the FieldTrip toolbox

Talk outline

What kind of signals are generated in the brain

How do we record those signals

Analyzing those signals with FieldTrip

Background on the FieldTrip toolbox

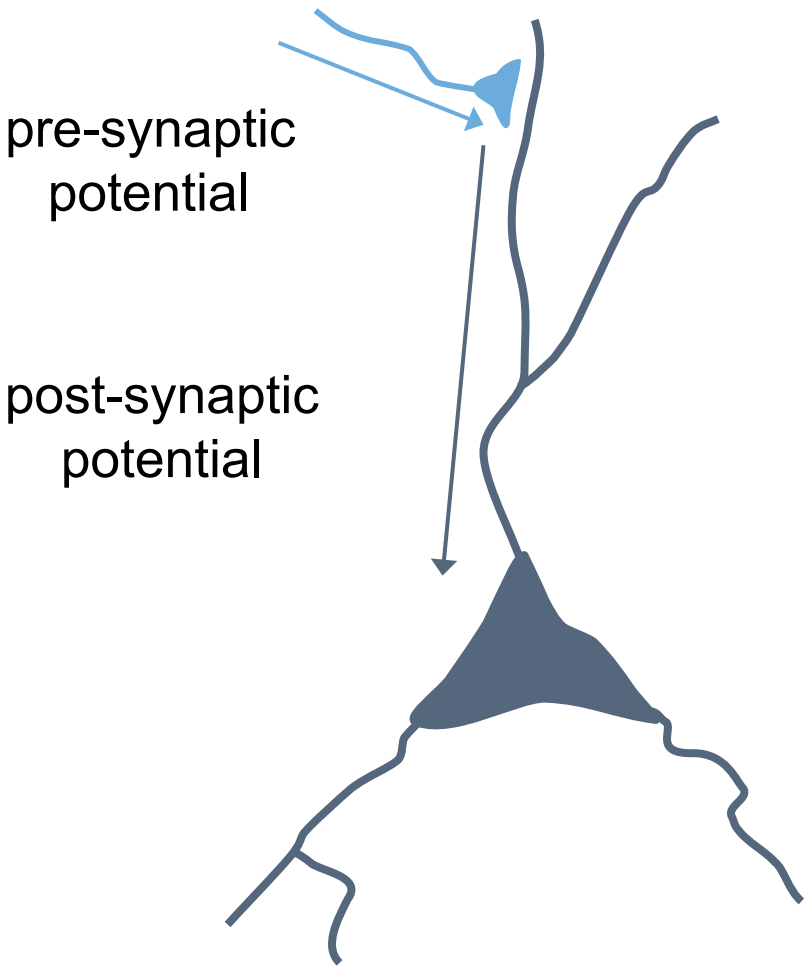
What kind of signals are generated in the brain

We measure the scalp potentials or field associated with post-synaptic potentials in pyramidal neurons

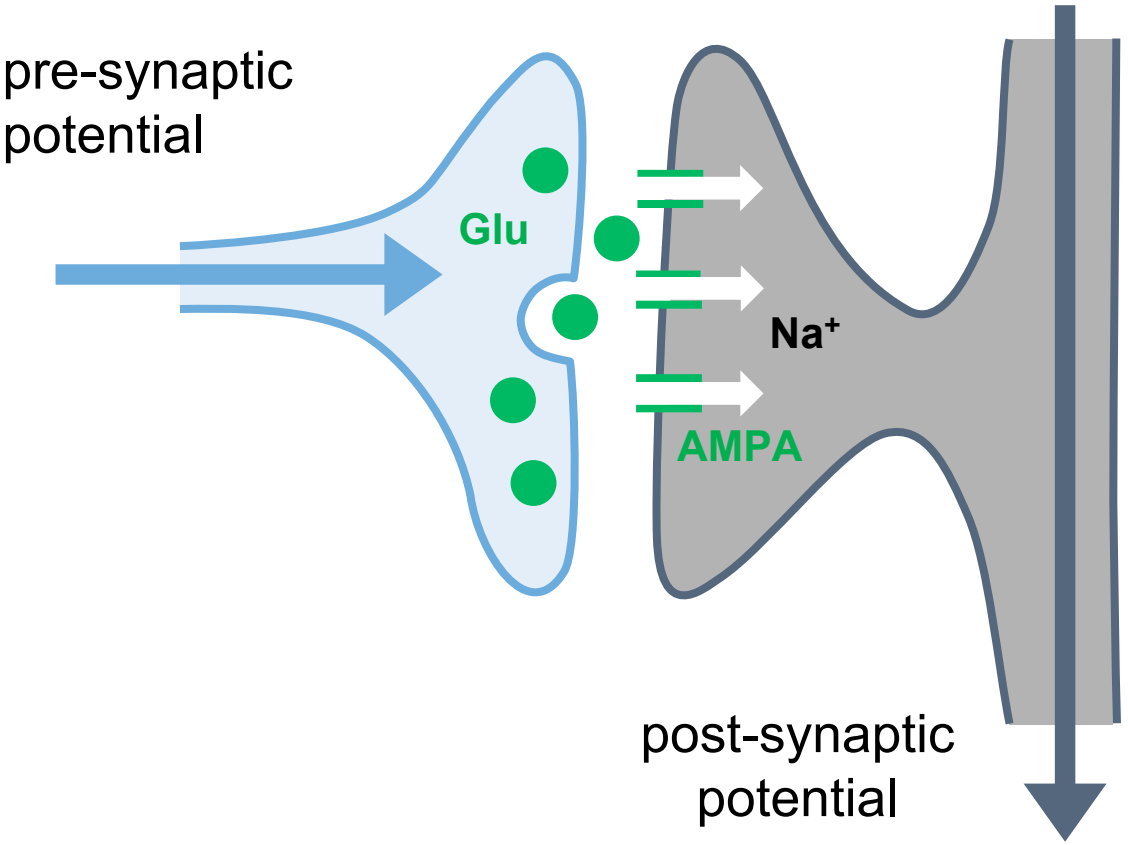
These PSPs represent the exitatory and inhibitory input that these neurons receive

Usually we study this neuronal input following the presentation of a stimulus or following a cognitive event

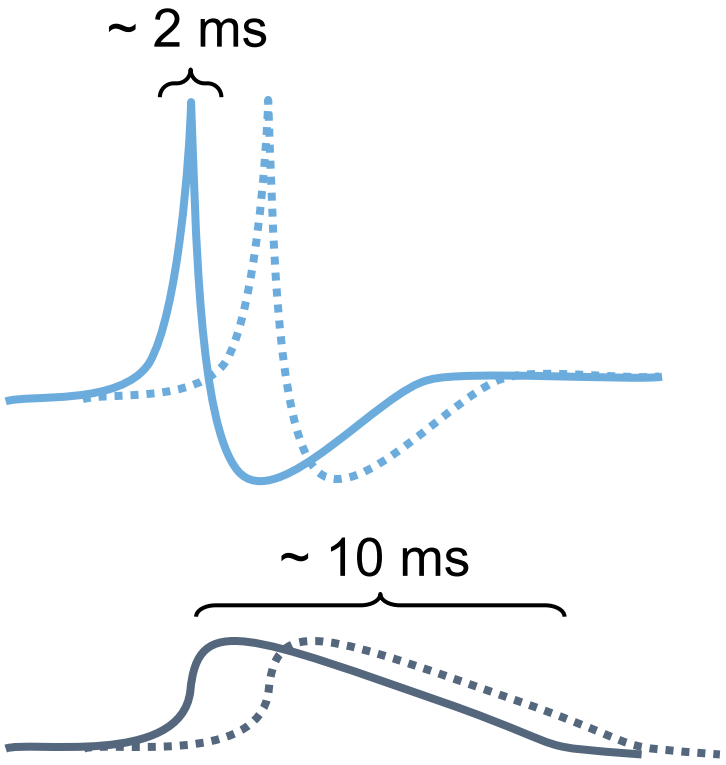
What generates the currents and fields



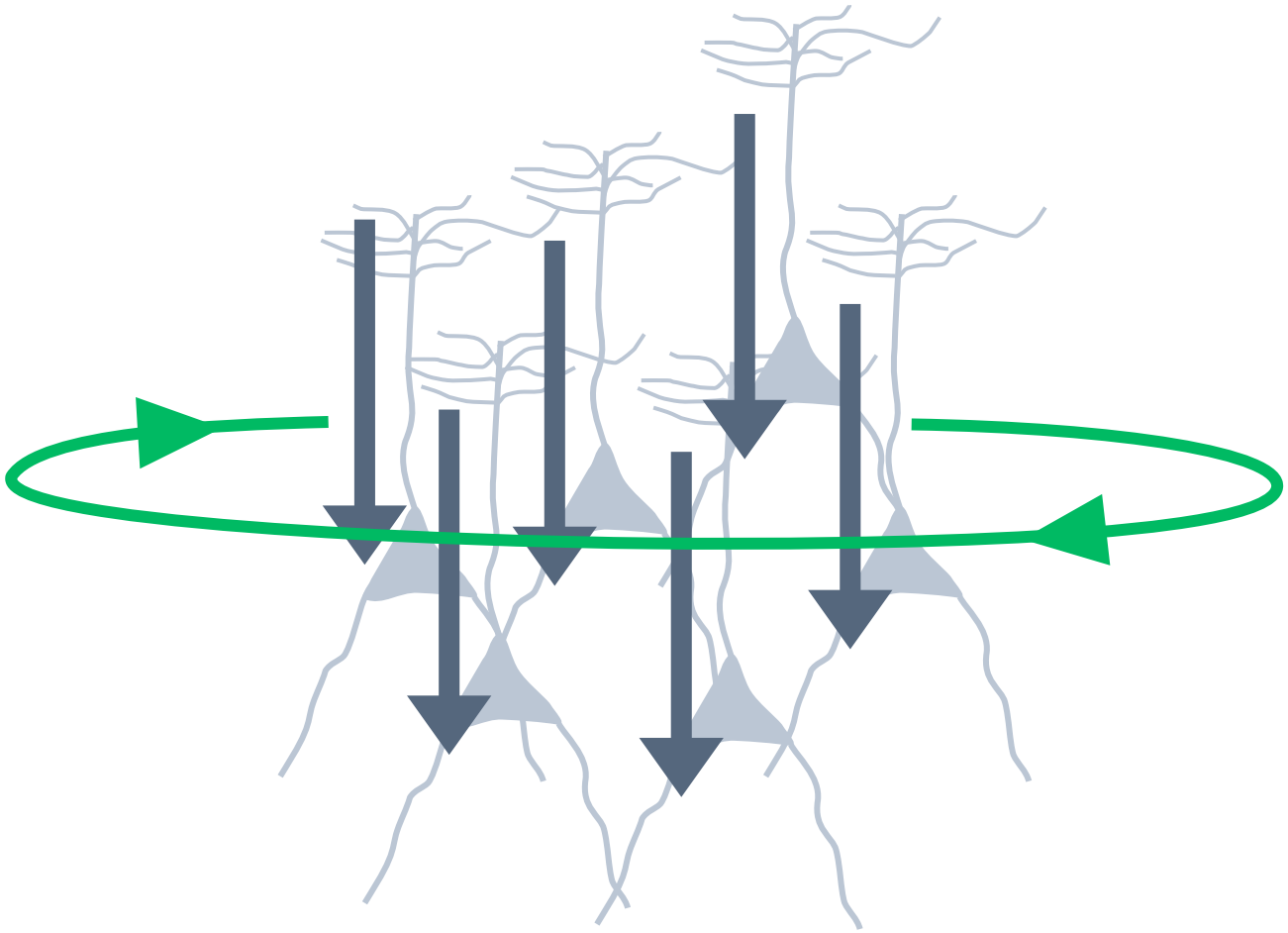
What generates the currents and fields



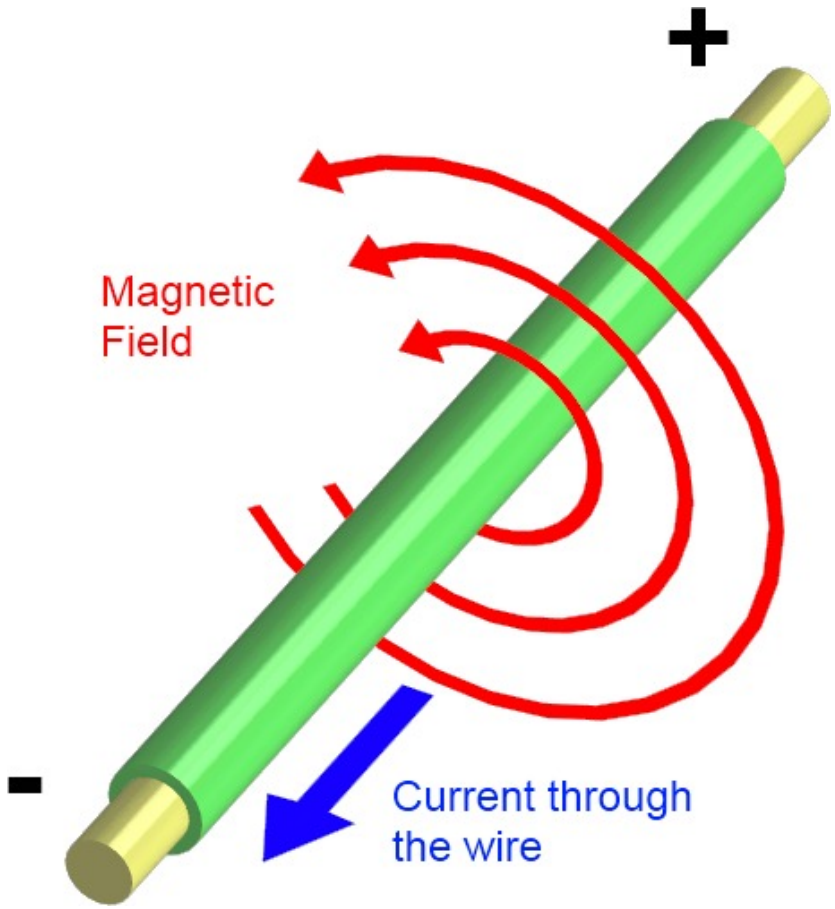
What generates the currents and fields



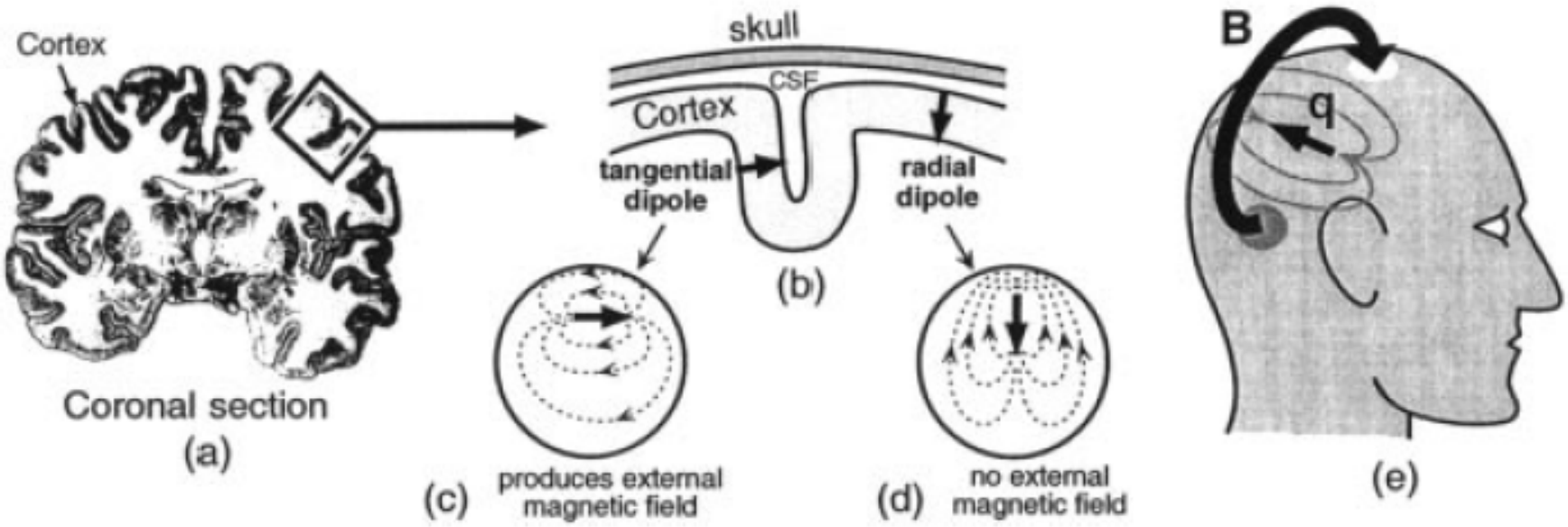
What generates the currents and fields



What generates the currents and fields

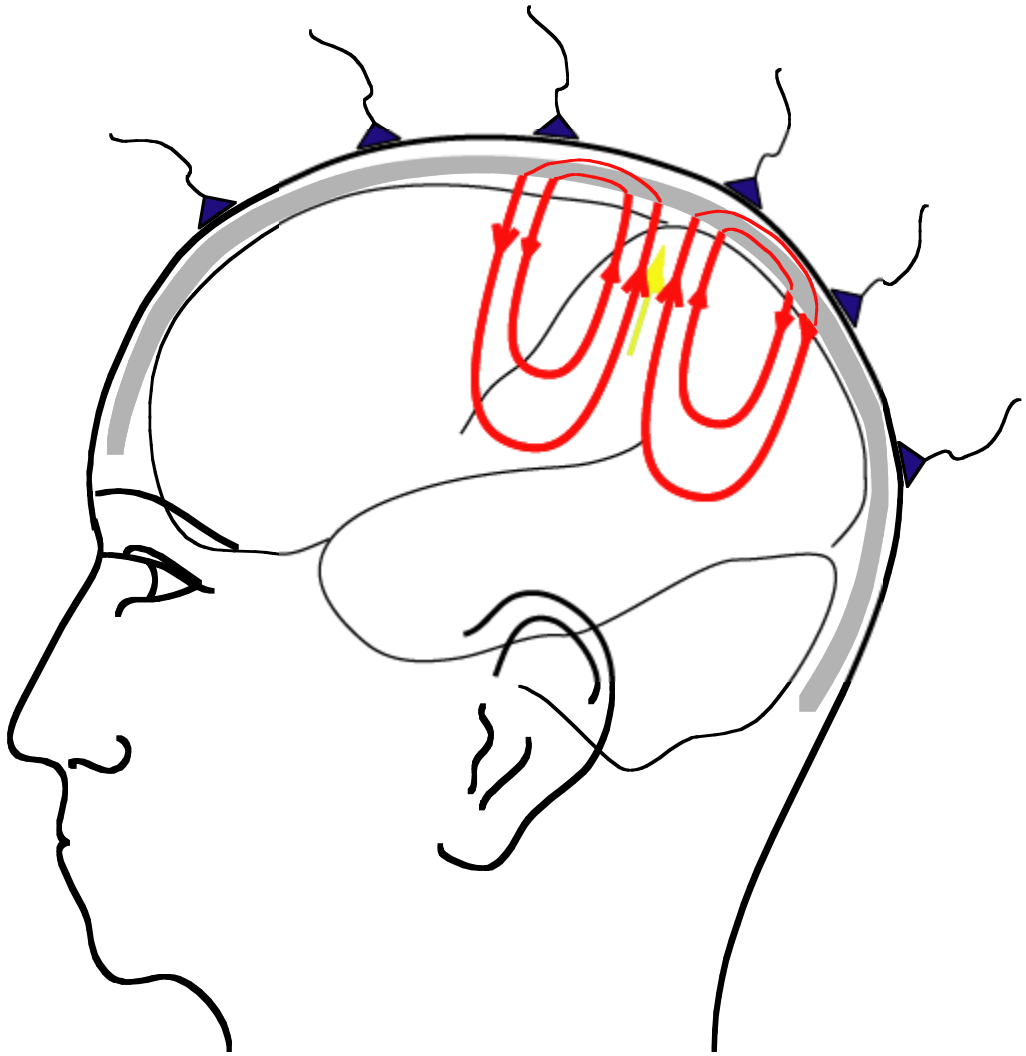


What generates the currents and fields

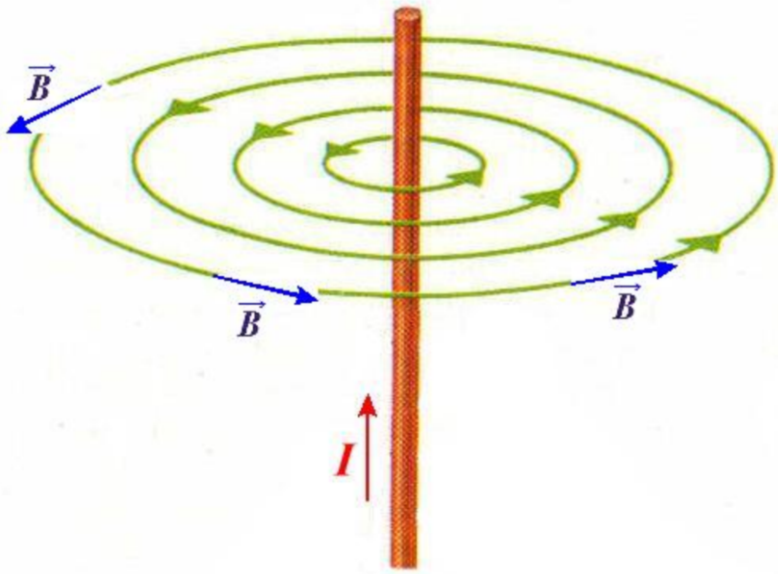


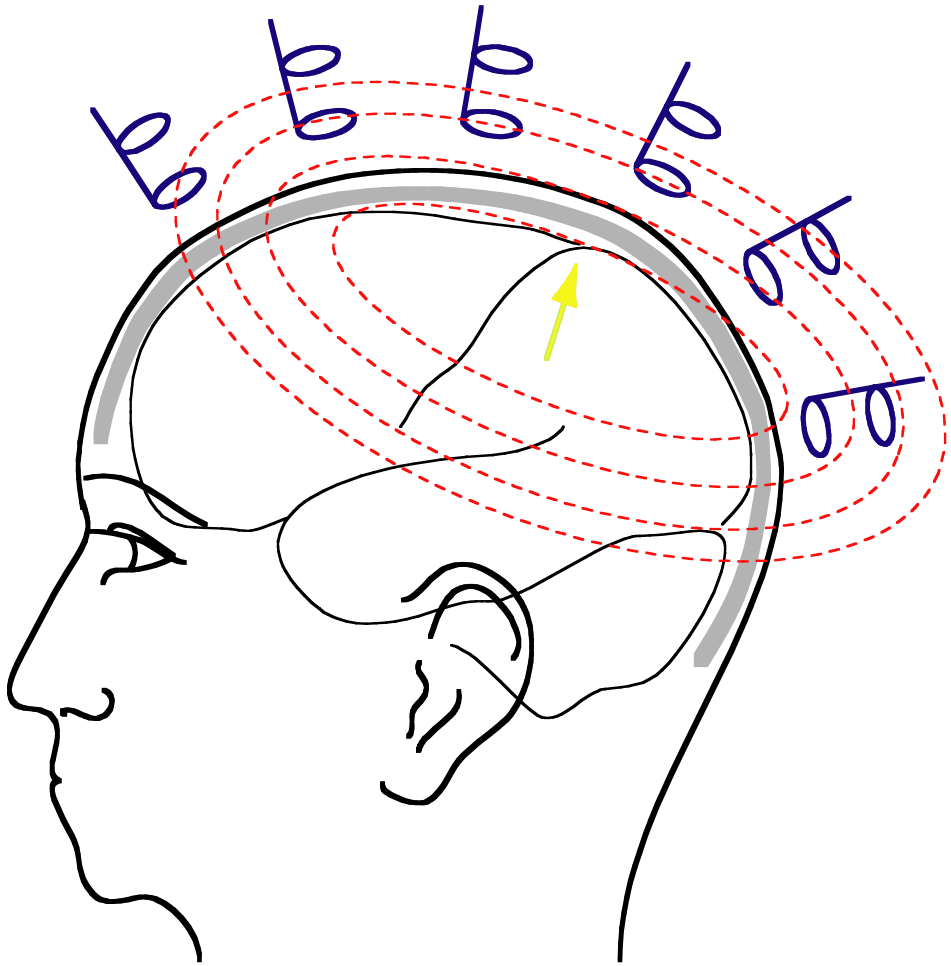
Vrba, 2001

EEG volume conduction



Electric current → magnetic field





Talk outline

What kind of signals are generated in the brain

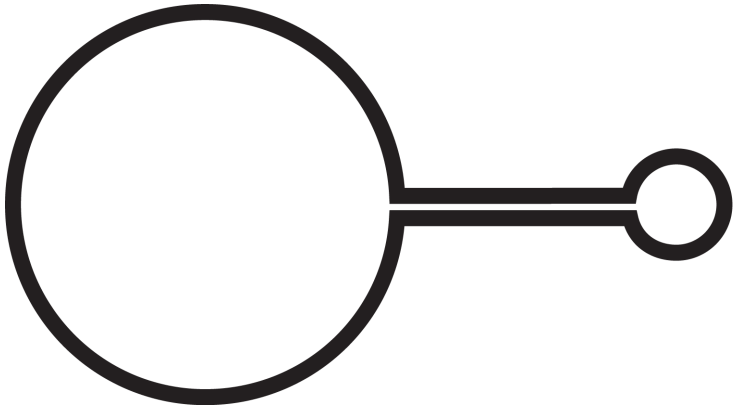
How do we record those signals

Analyzing those signals with FieldTrip

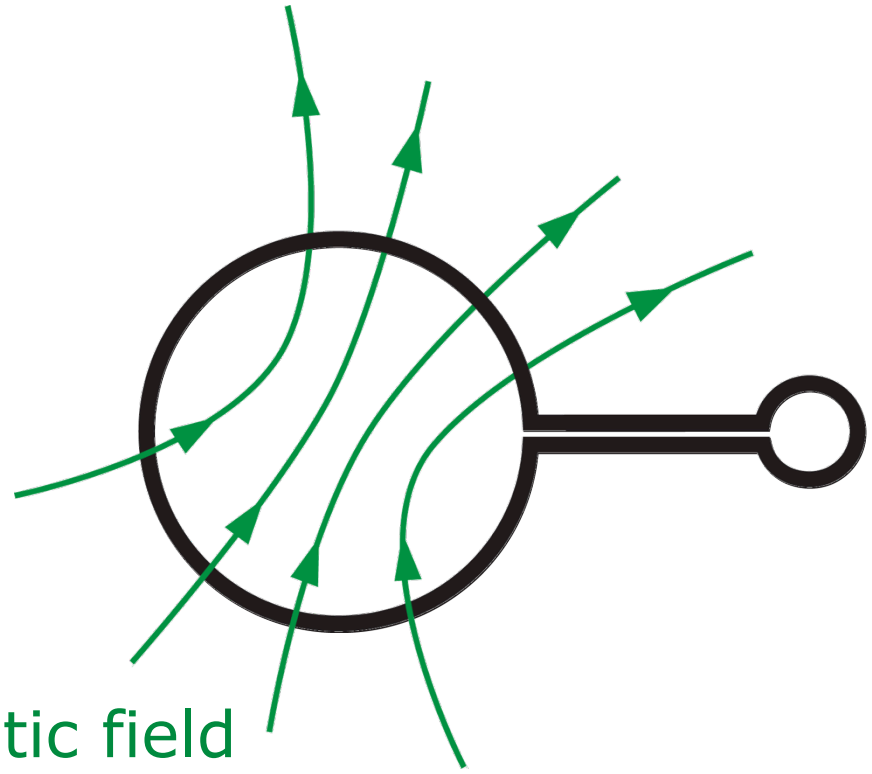
Background on the FieldTrip toolbox

How do we record these signals from the brain

Recording small magnetic fields

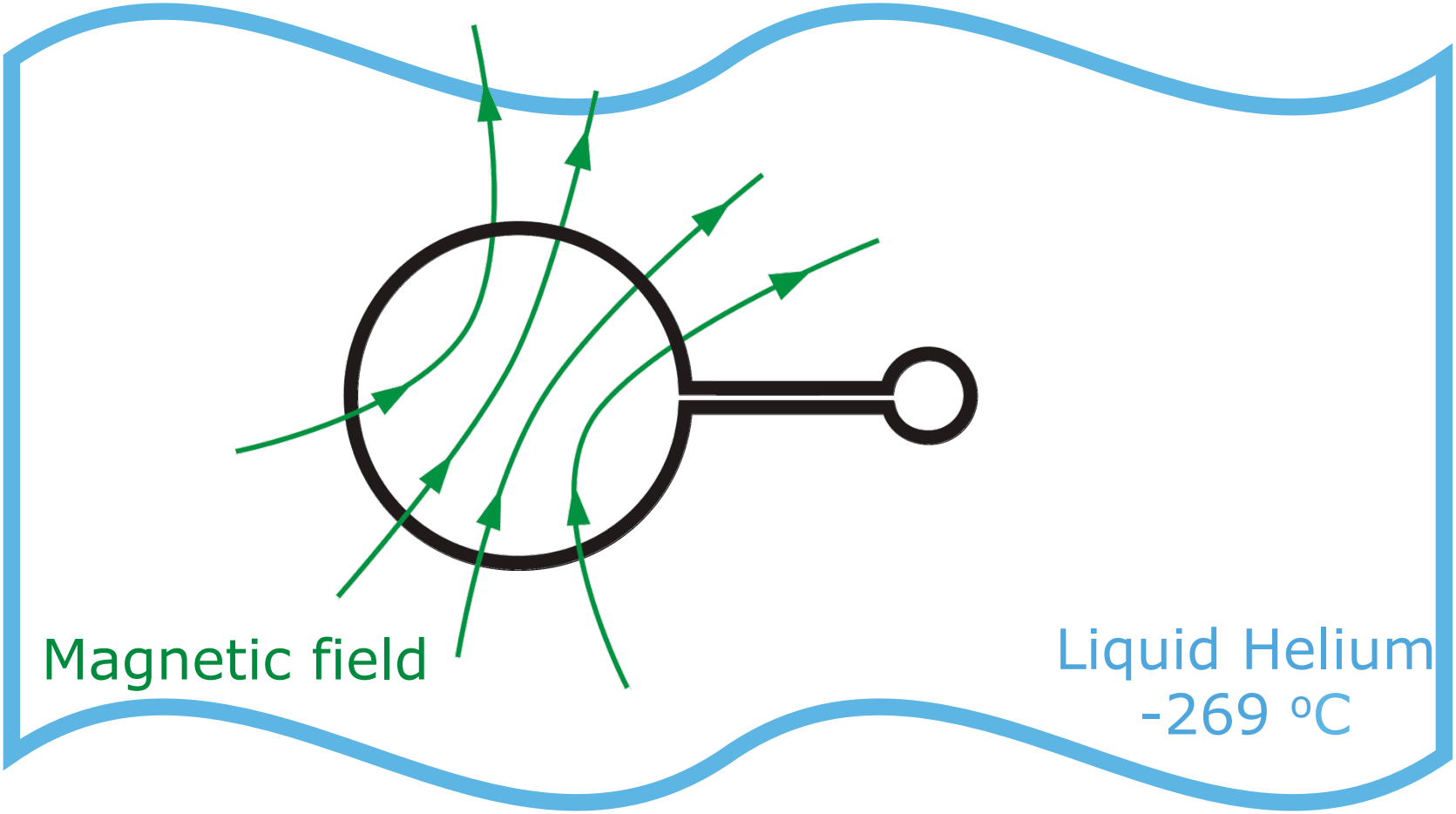


Recording small magnetic fields

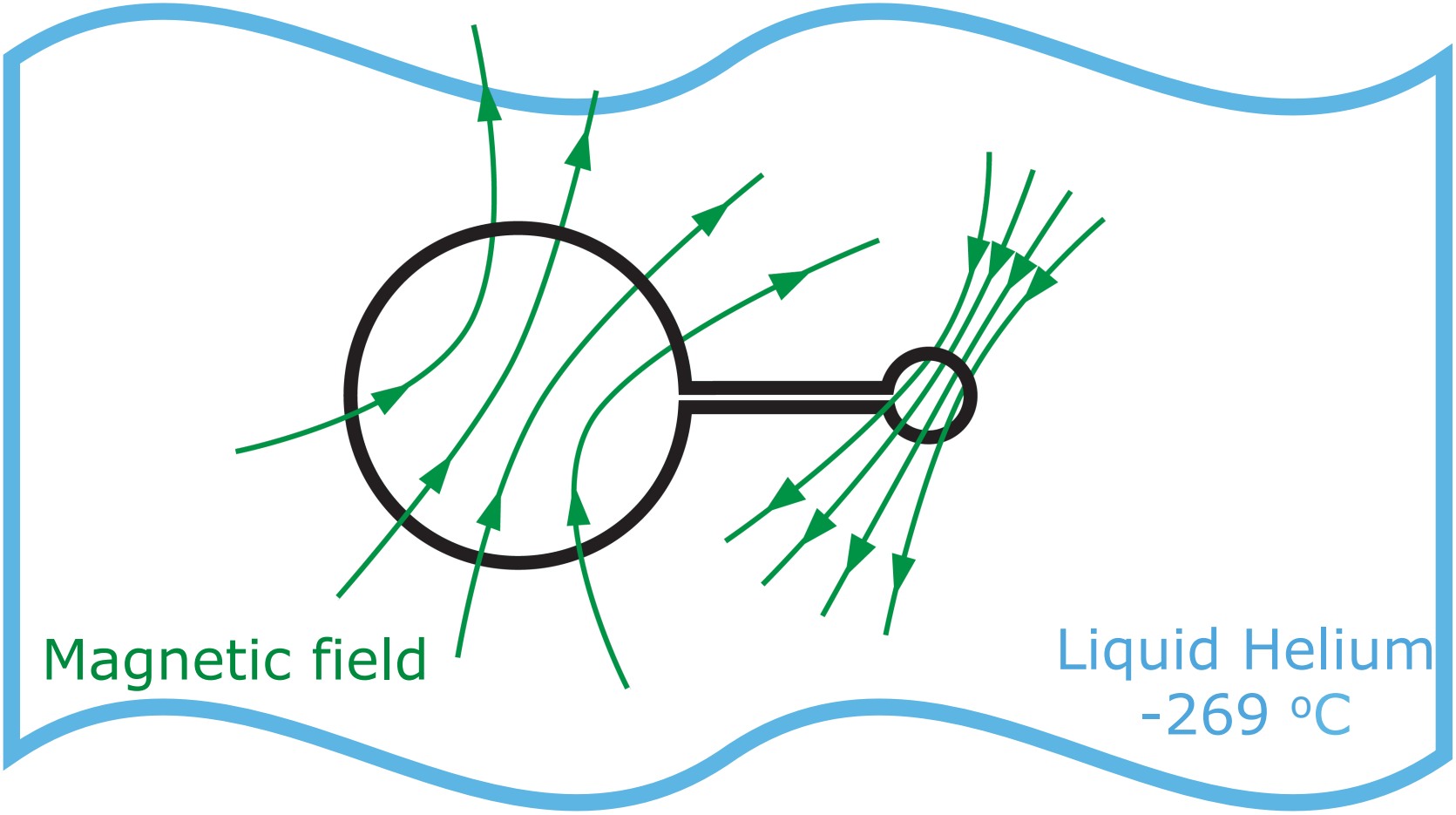


Magnetic field

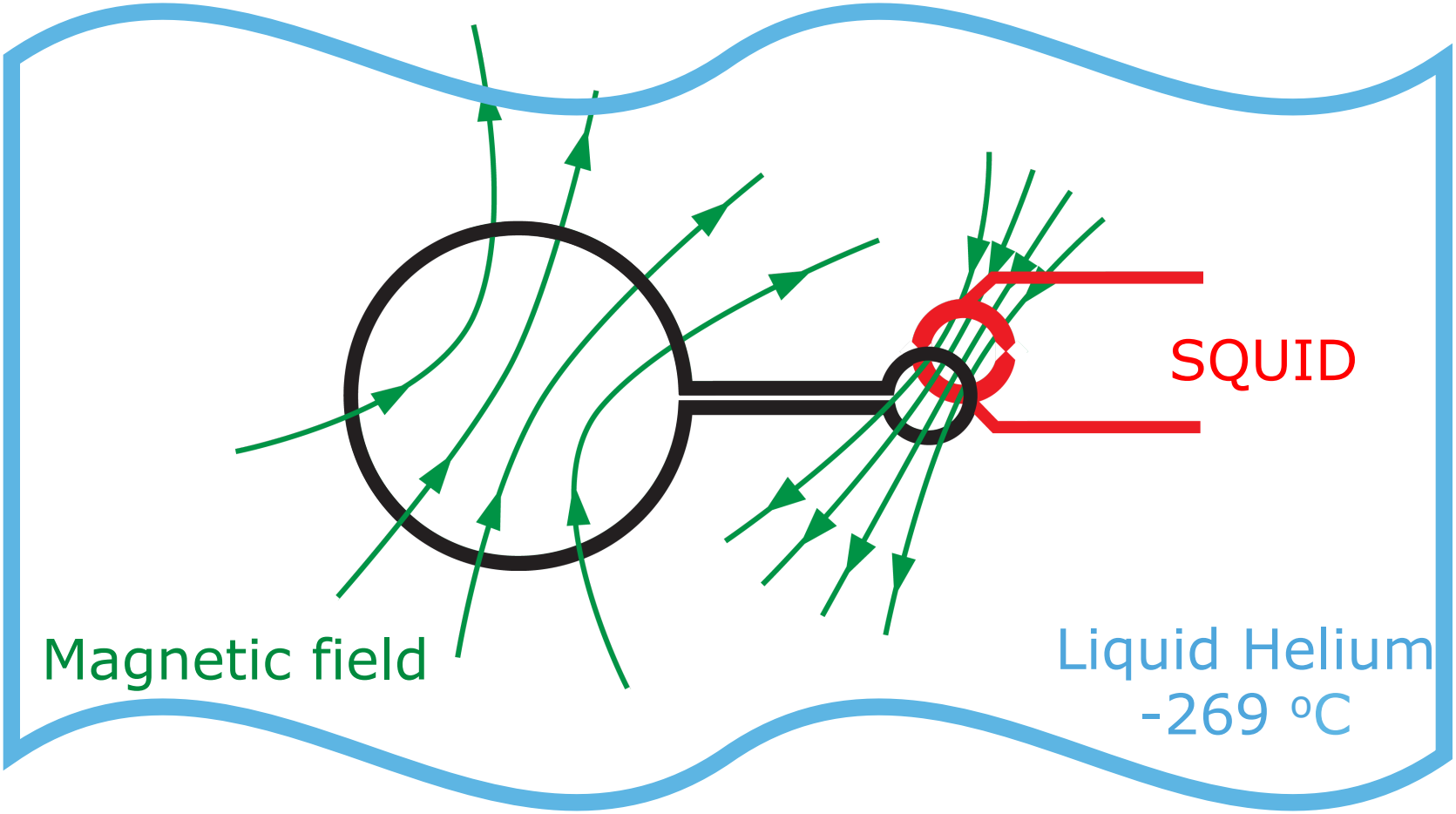
Recording small magnetic fields



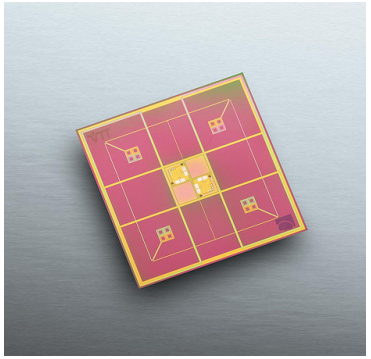
Recording small magnetic fields

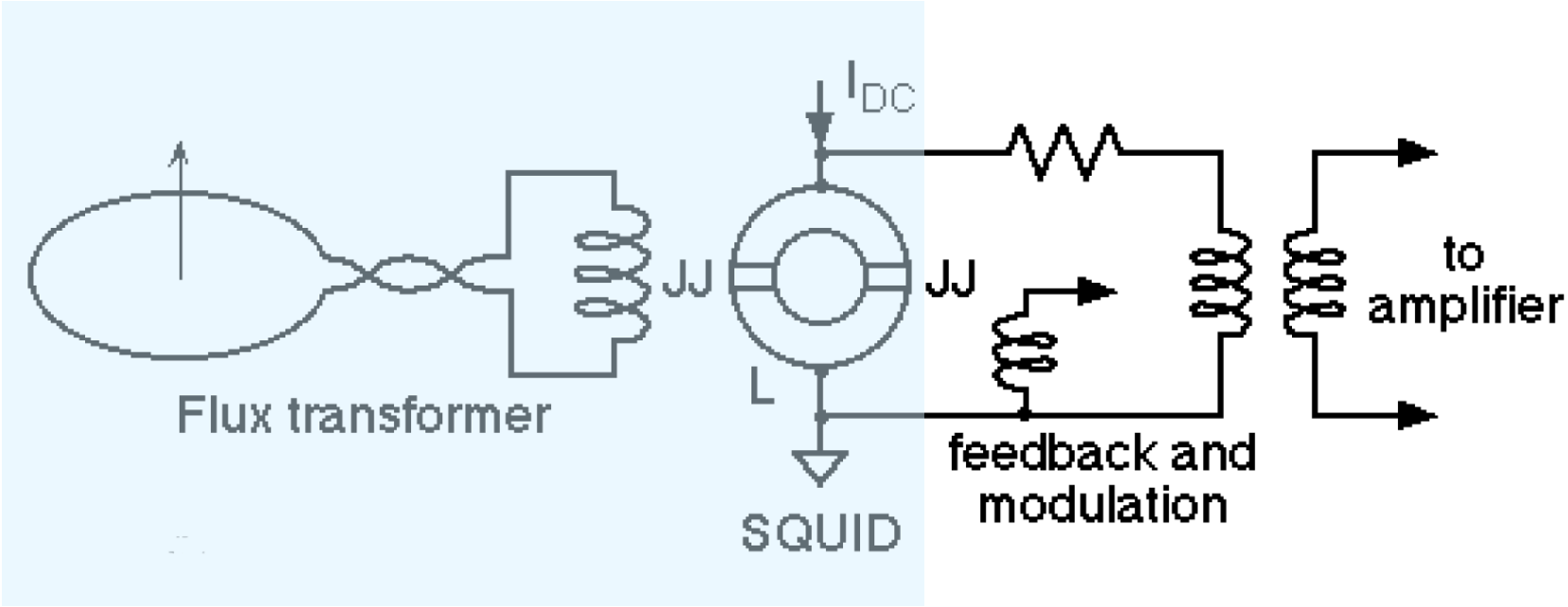


Recording small magnetic fields



Magnetic field detectors
Superconducting **QU**antum **I**nterference **D**evice





Magnetic field strength - compared

10^{-12}	0.1 - 1.0 picoTesla	human brain
10^{-9}	0.1 -10 nanoTesla	heliosphere
10^{-6}	24 microTesla	magnetic tape near tape head
10^{-5}	300-600 μT microTesla	earth's magnetic field
10^{-3}	5 milliTesla	typical refrigerator magnet
10^{0}	1.5 - 7 Tesla	MRI systems

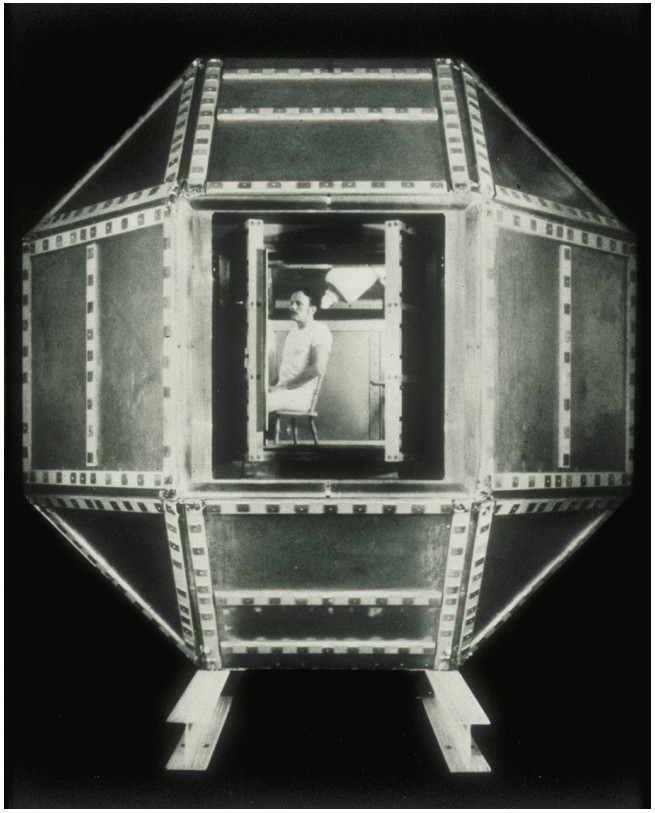
Technical challenges of MEG

Requires sensitive magnetic detectors

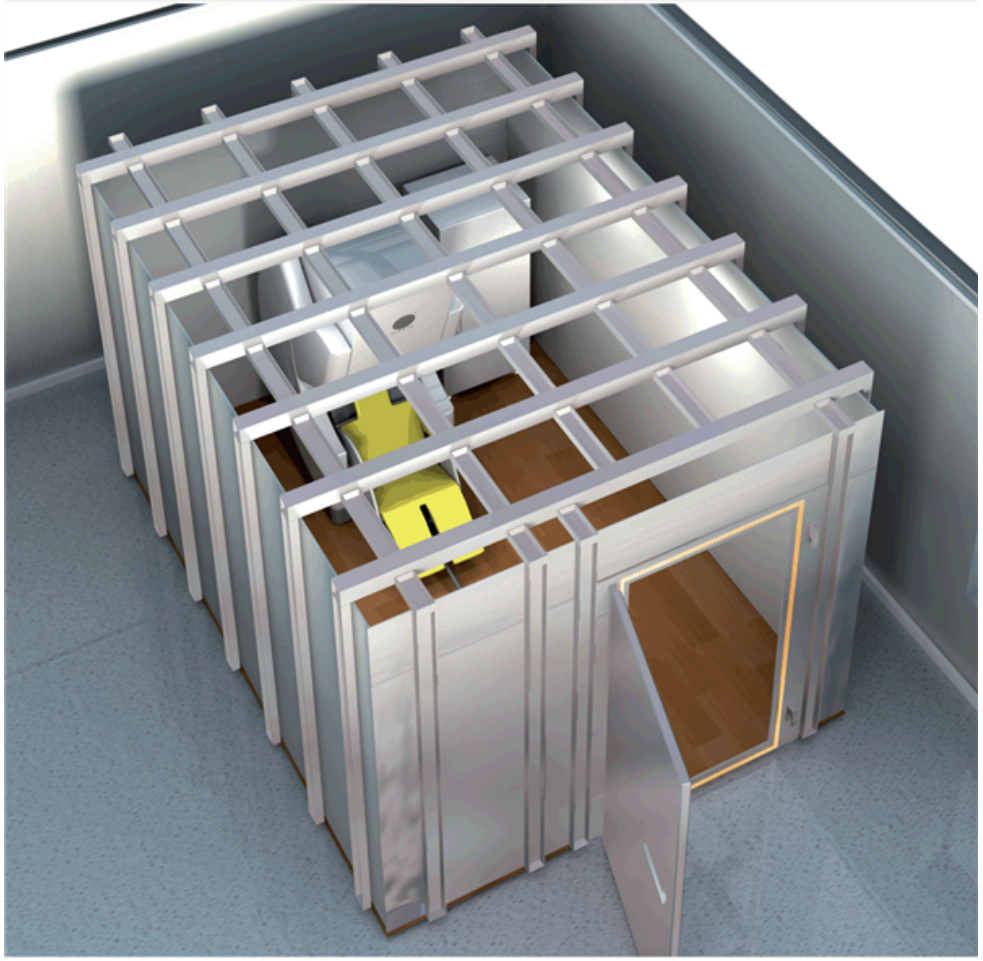
Deal with environmental noise

- shielding
- sensor design
- reference sensors for noise subtraction

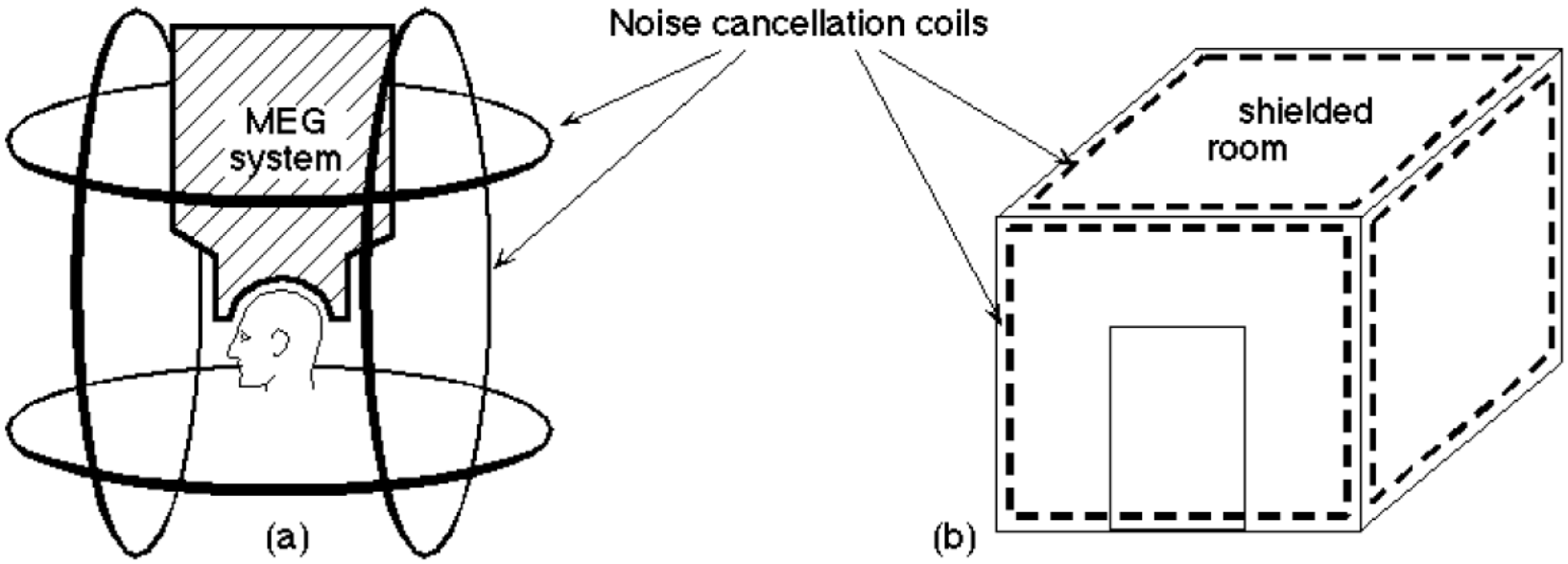
Shielding - passive



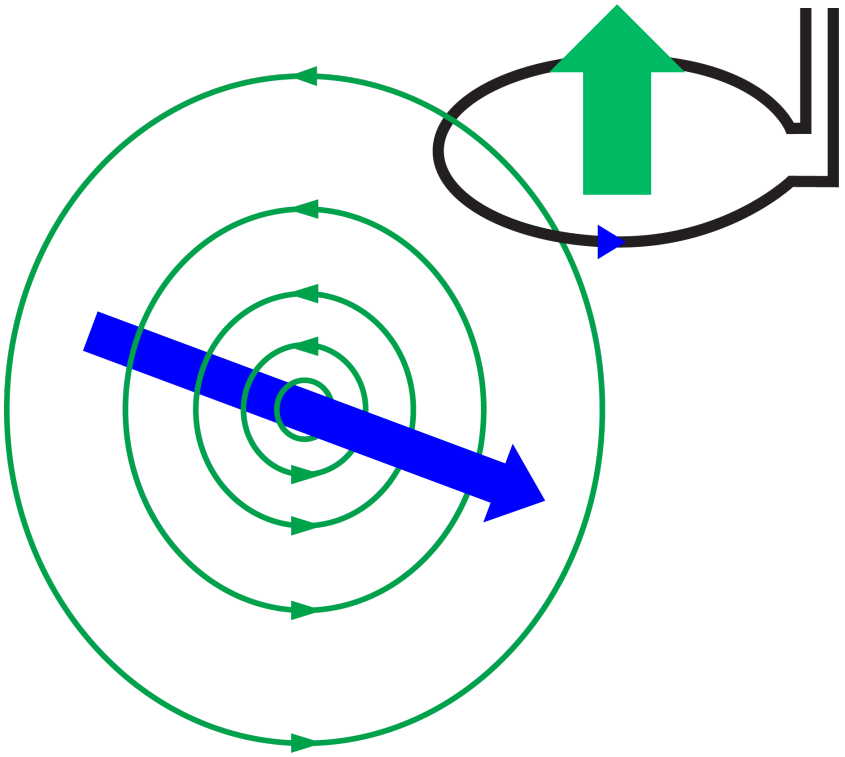
The magnetically shielded room built by David Cohen at MIT's Francis Bitter National Magnet Laboratory in 1969.



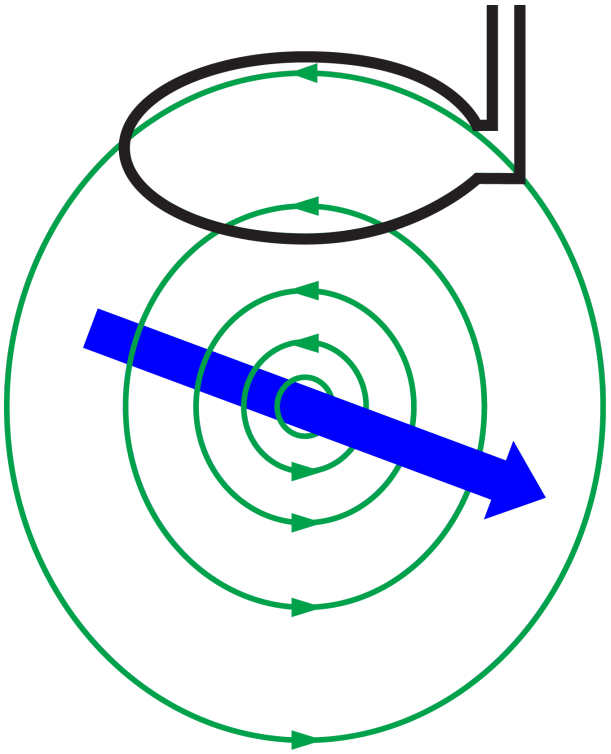
Shielding - active



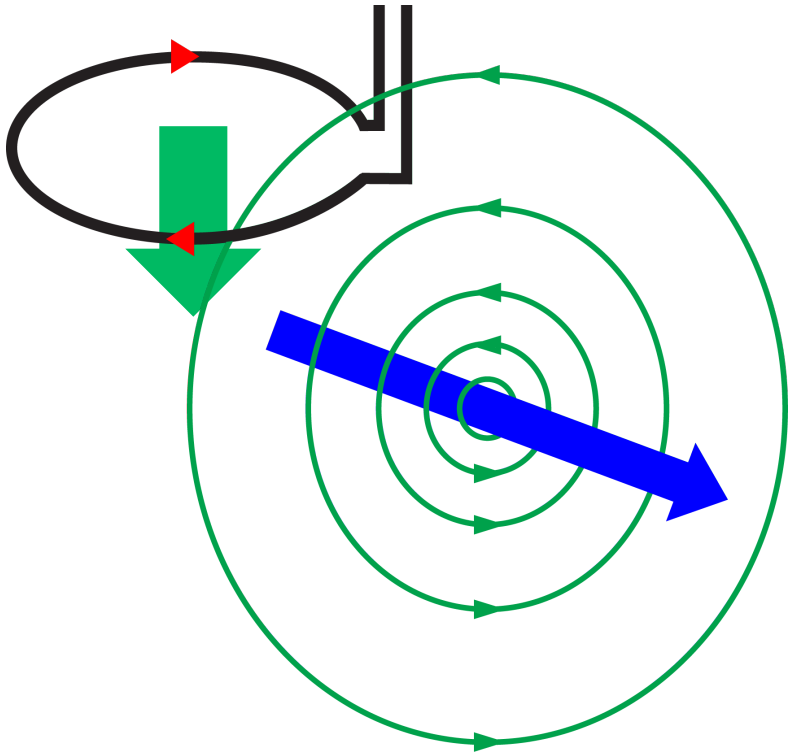
Magnetometer



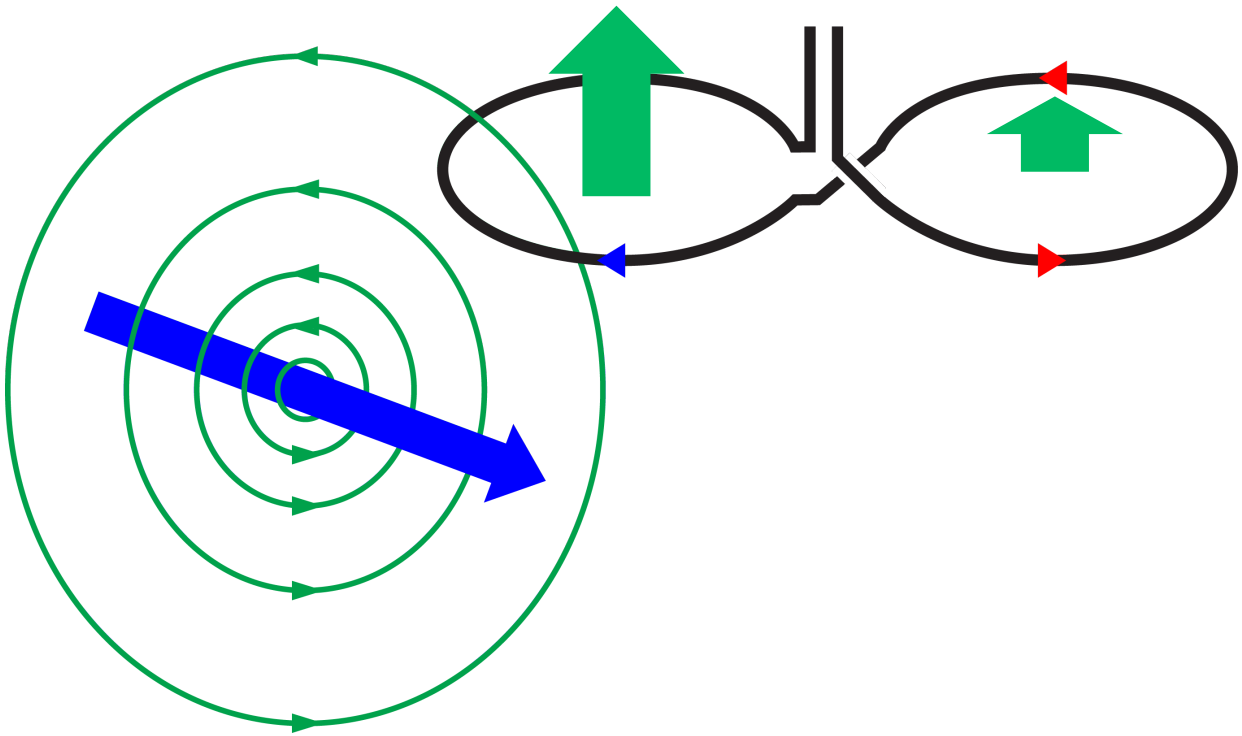
Magnetometer



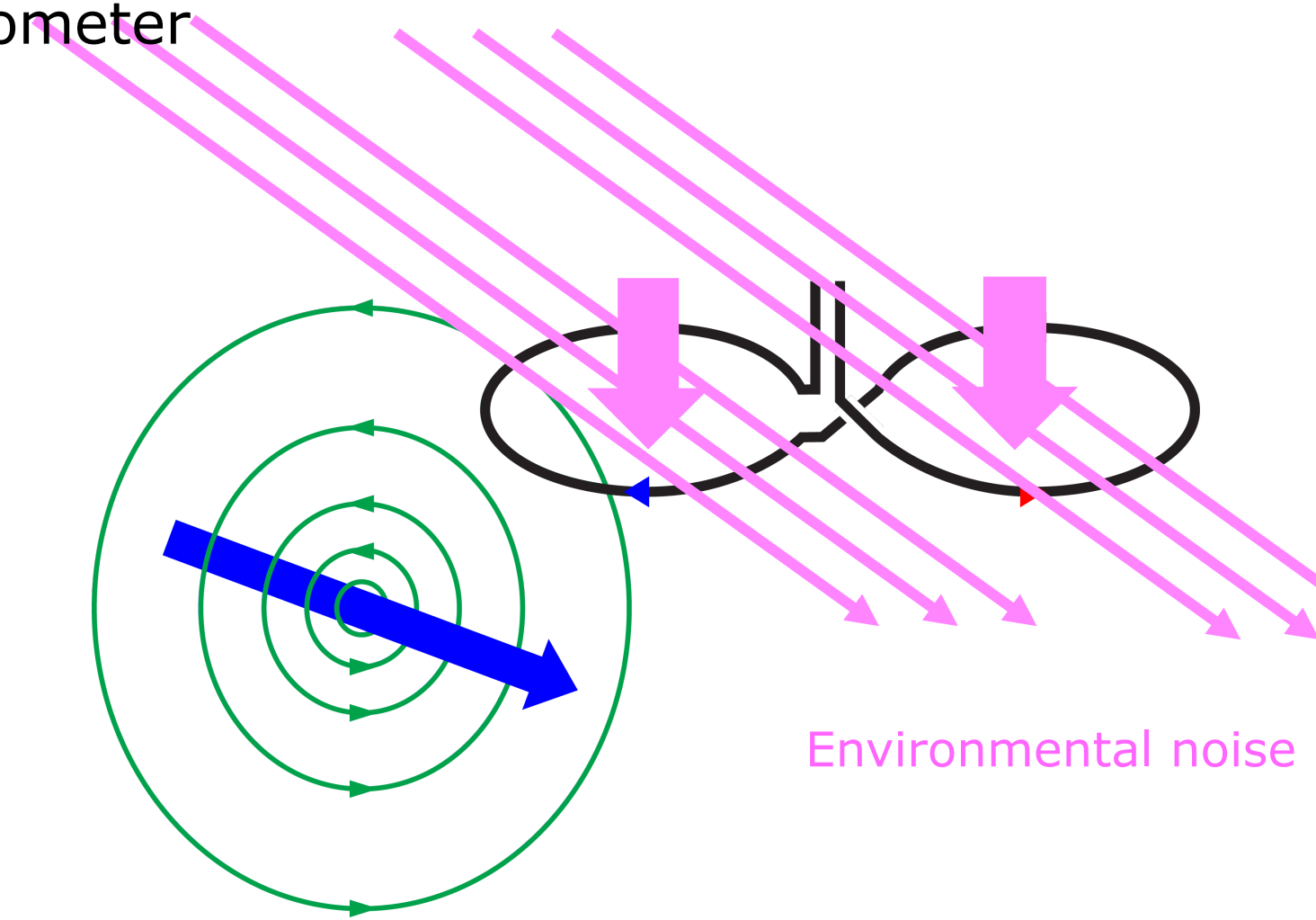
Magnetometer



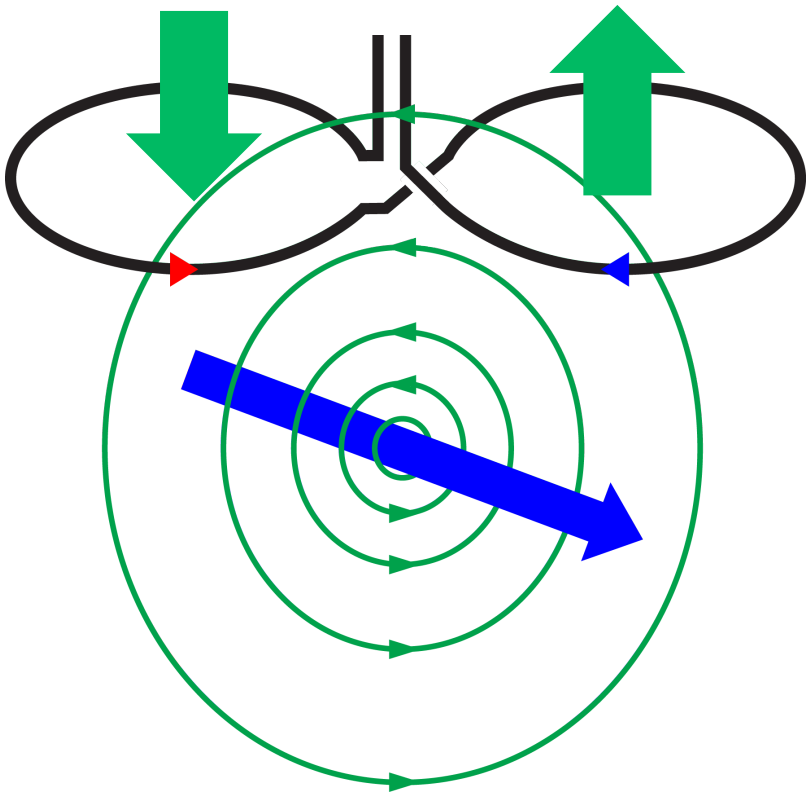
Planar gradiometer



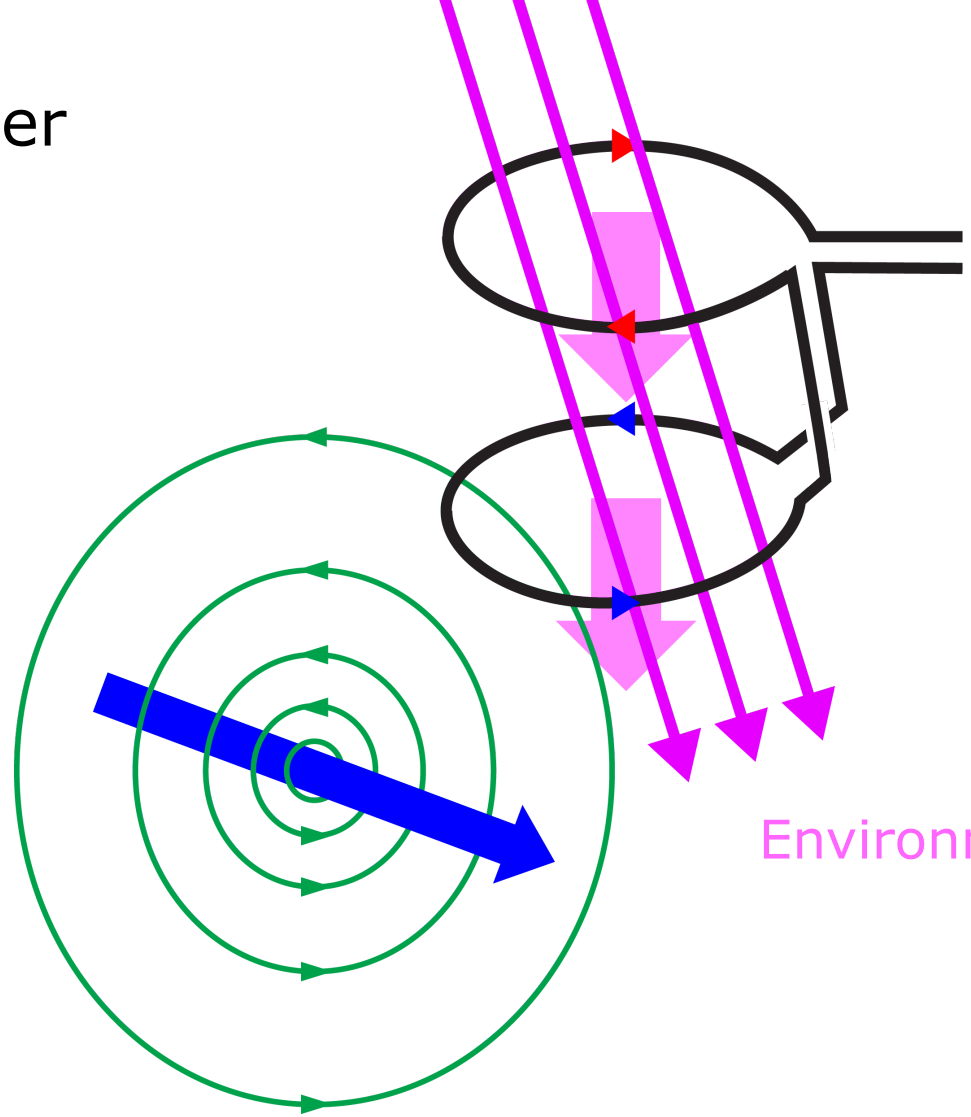
Planar gradiometer



Planar gradiometer

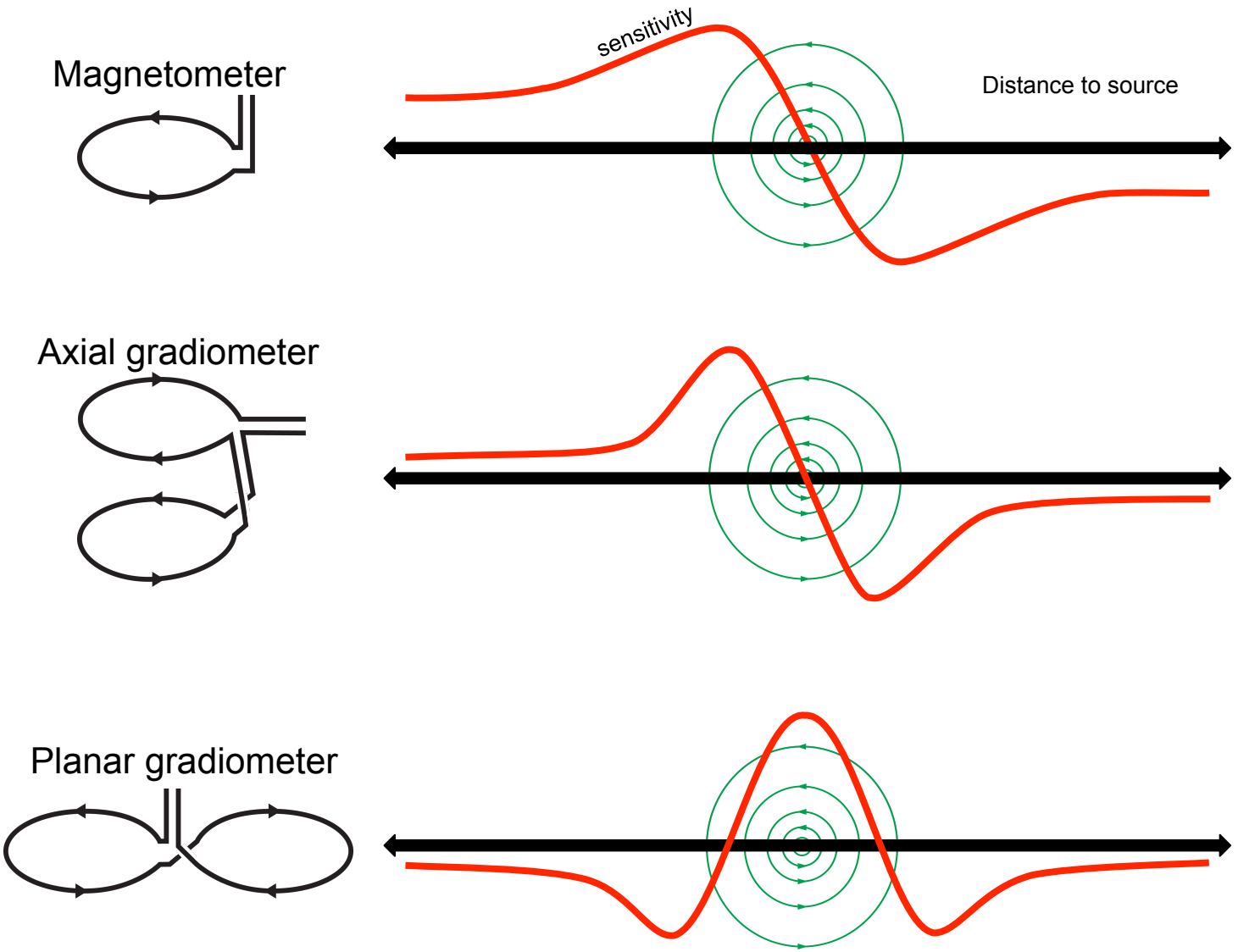


Axial gradometer



Environmental noise

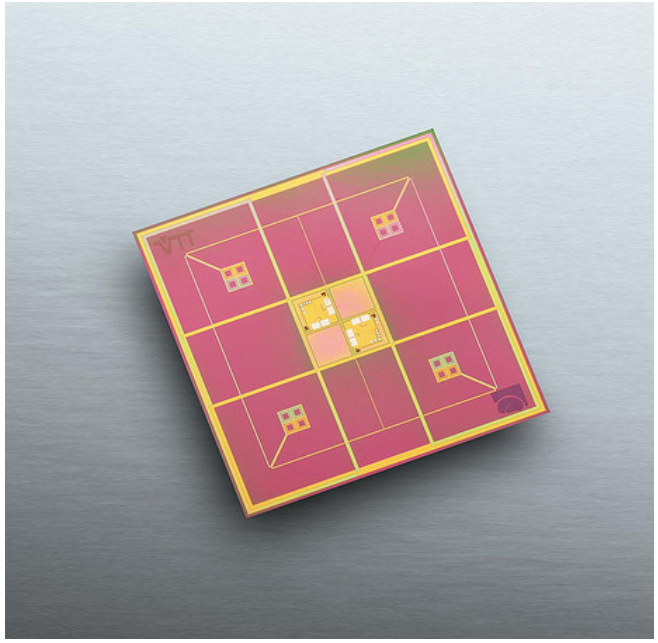
MEG sensor – sensitivity profile



MEG systems

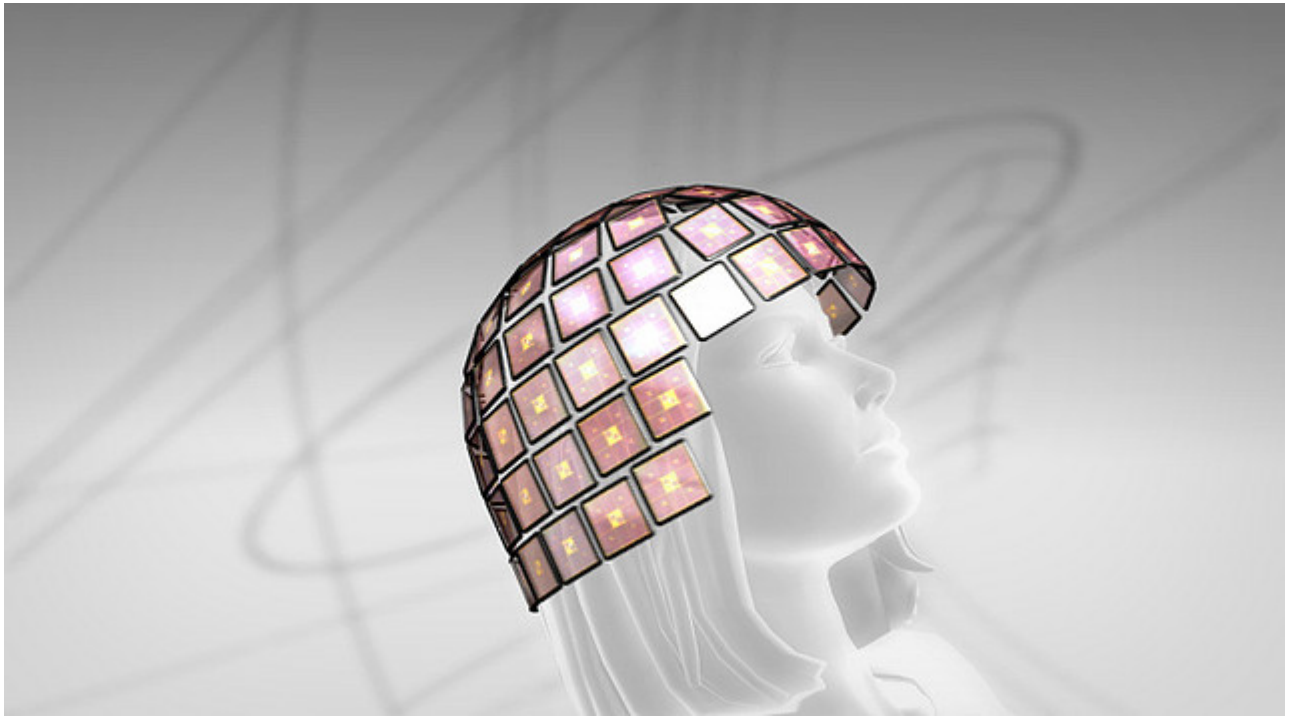
about 100-150 installations worldwide

two in the Netherlands

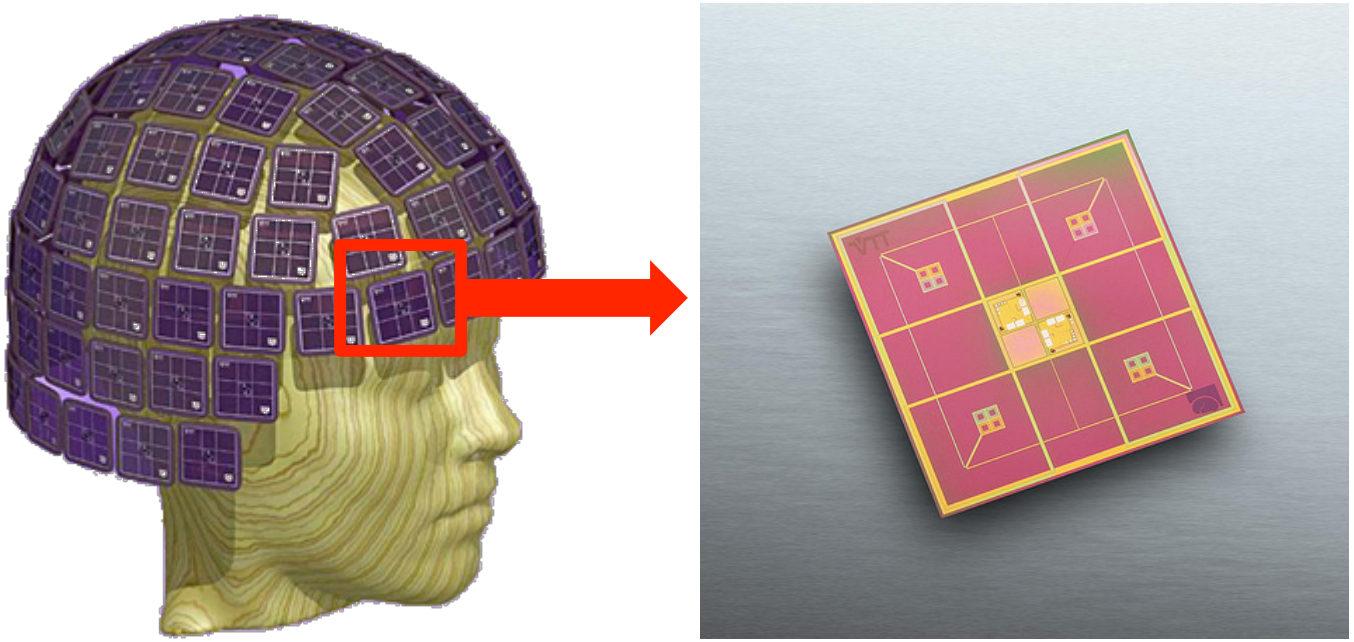


Elekta Neuromag

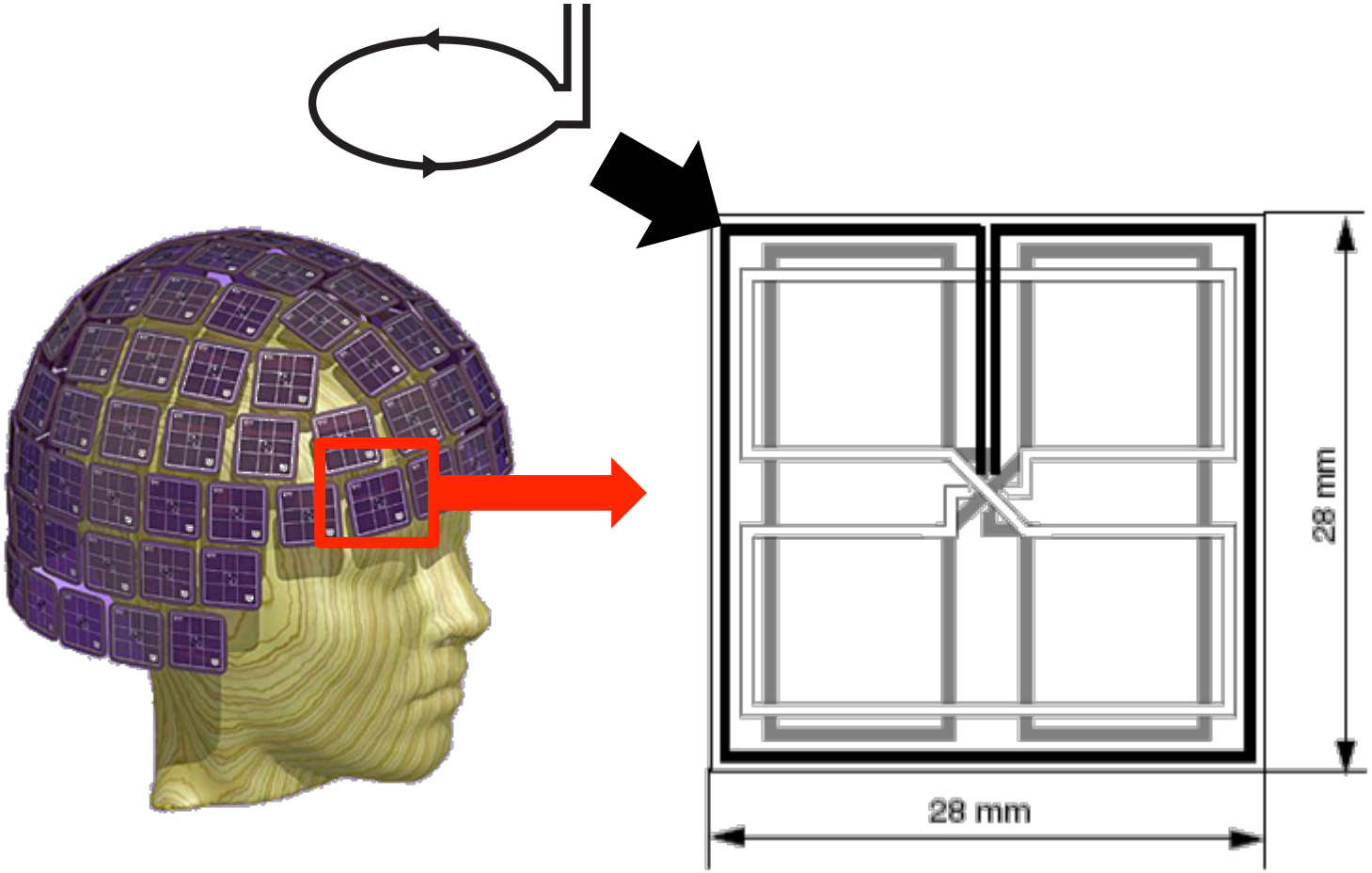
102 magnetometers
204 planar gradiometers
306 channels total



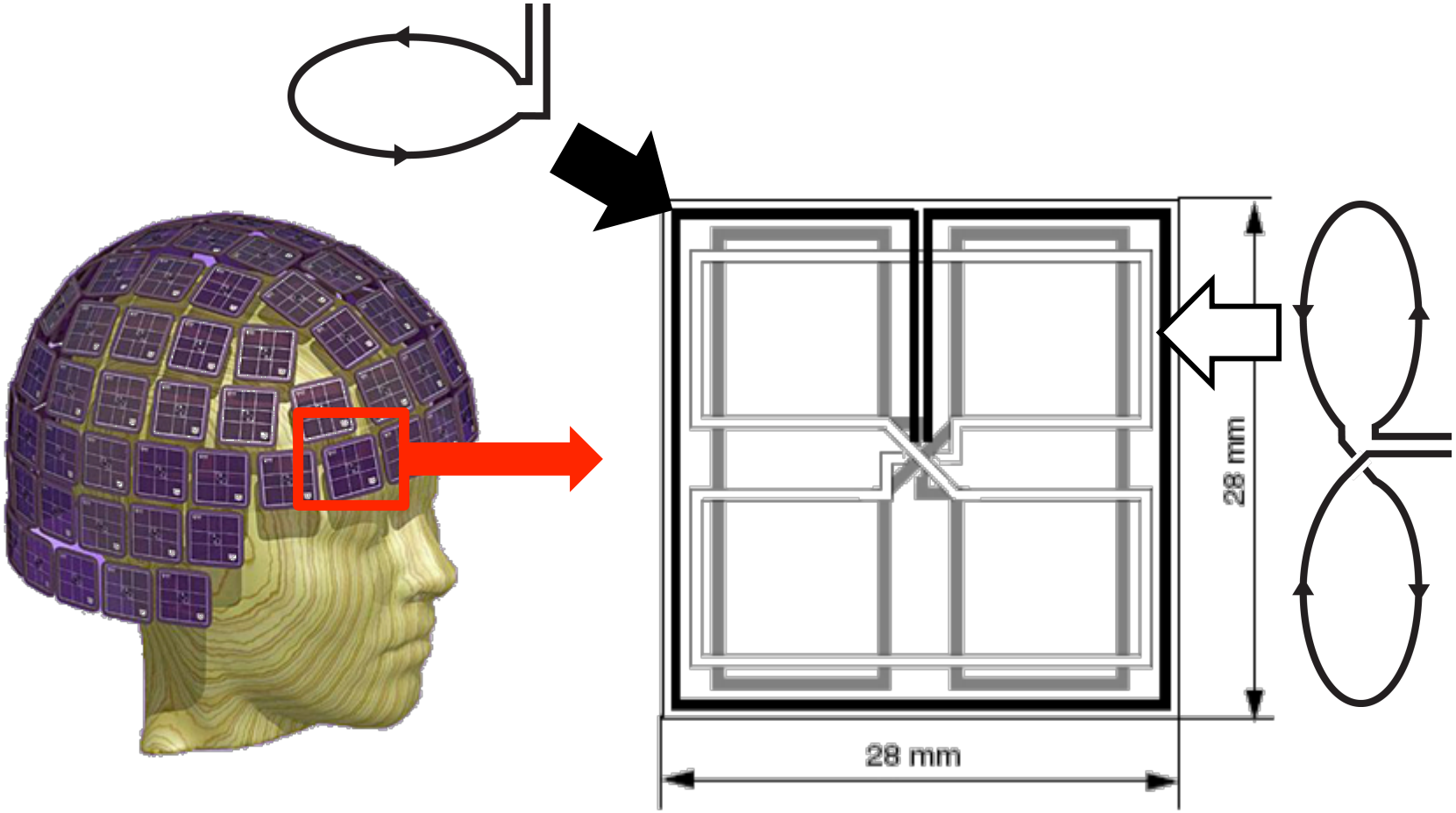
Elekta Neuromag



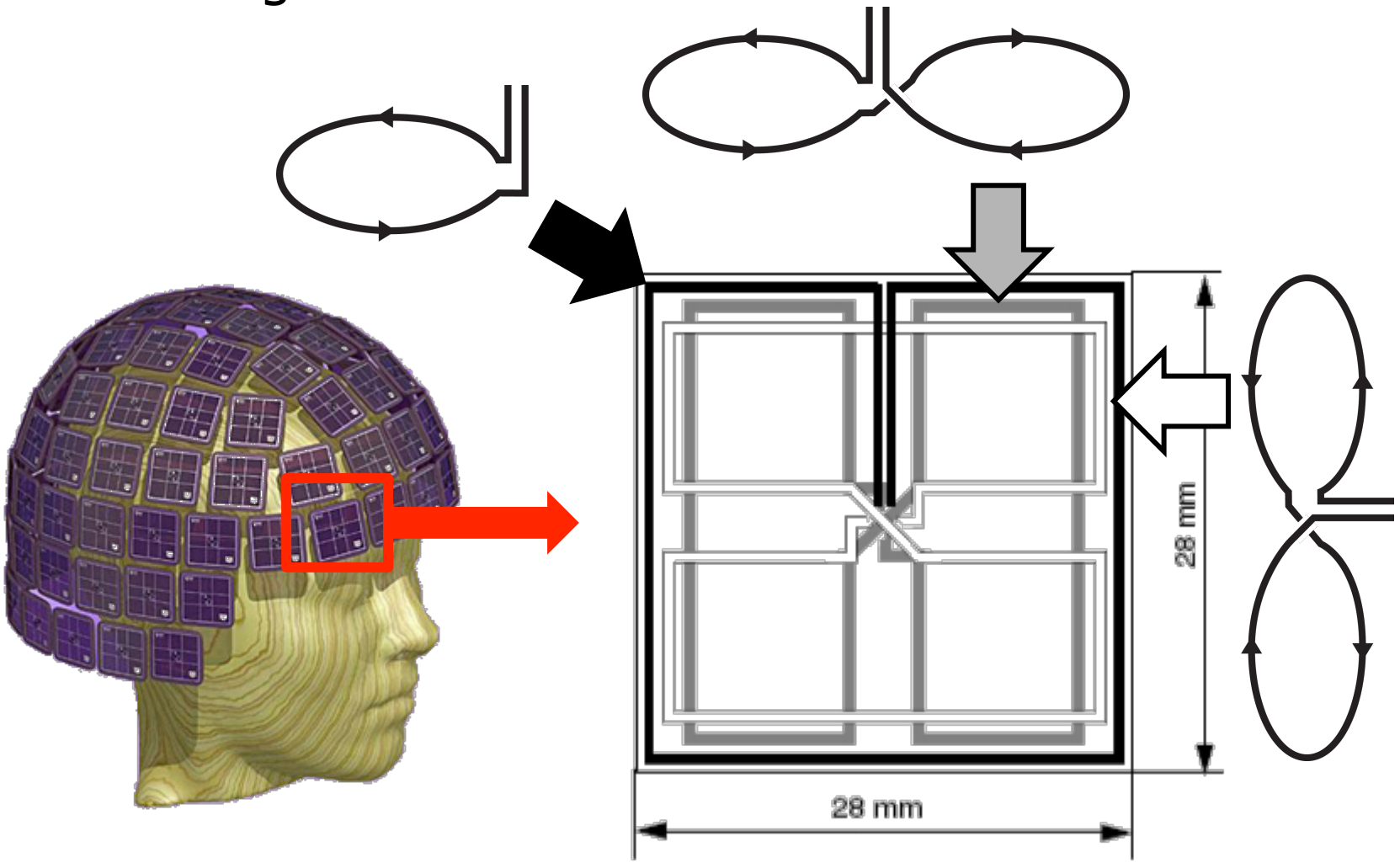
Elekta Neuromag



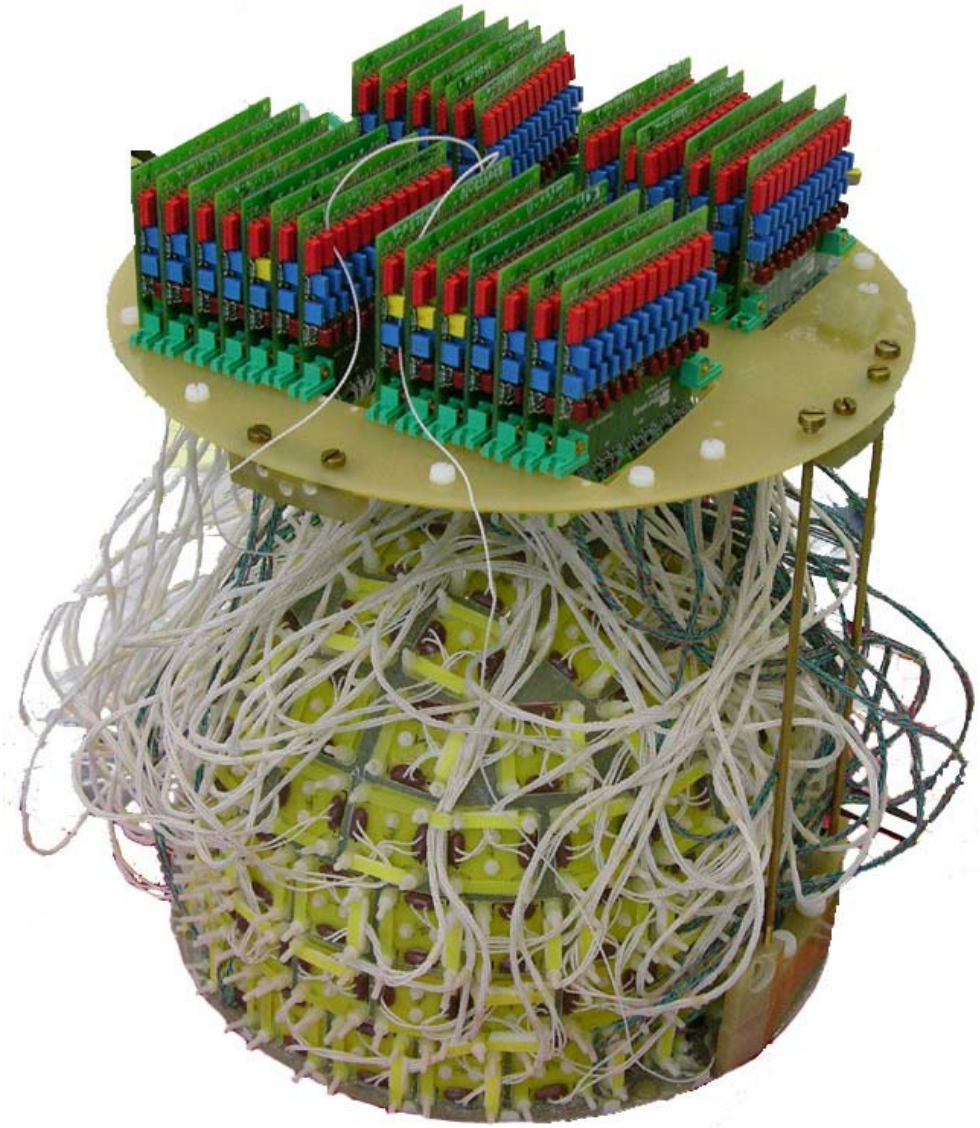
Elekta Neuromag



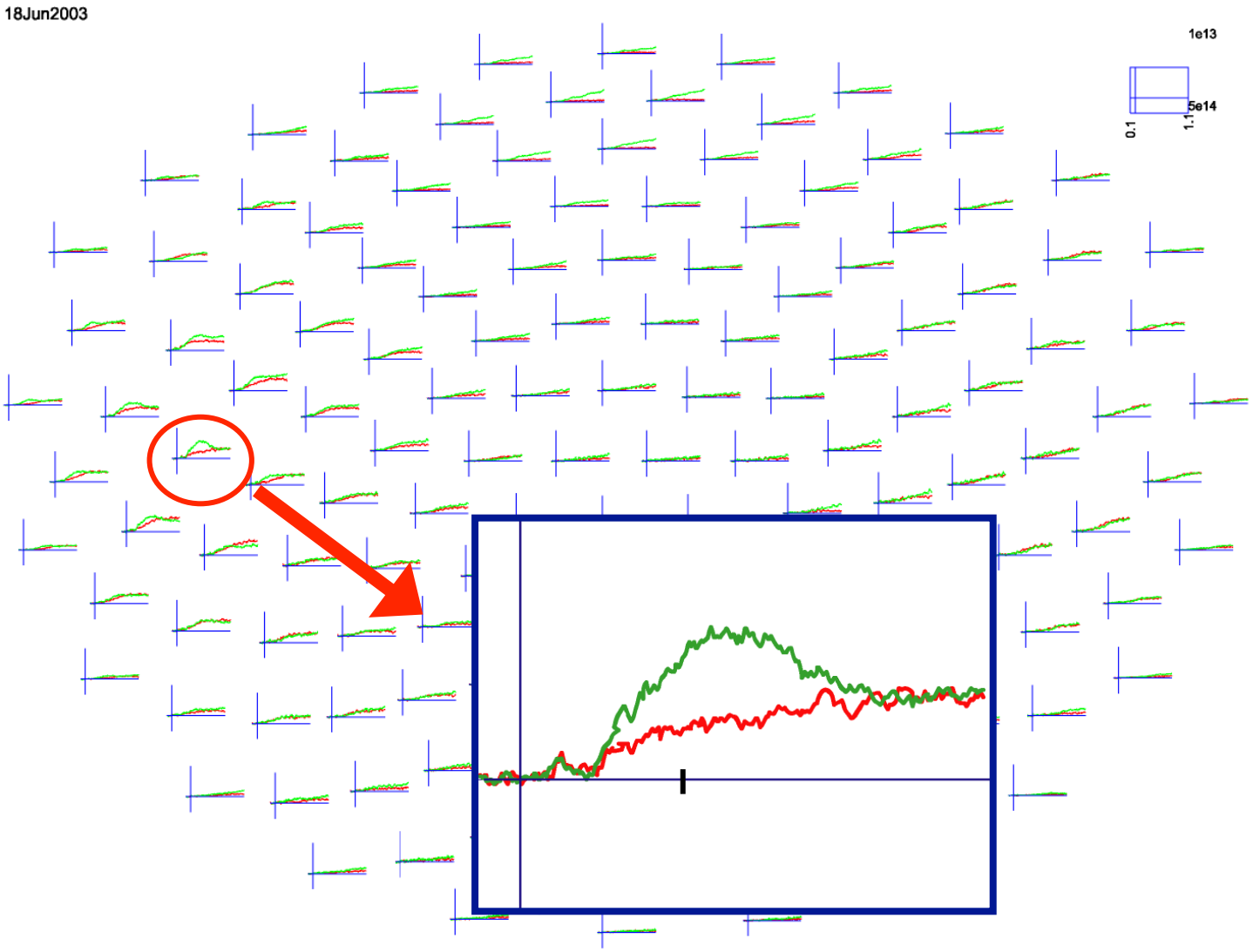
Elekta Neuromag



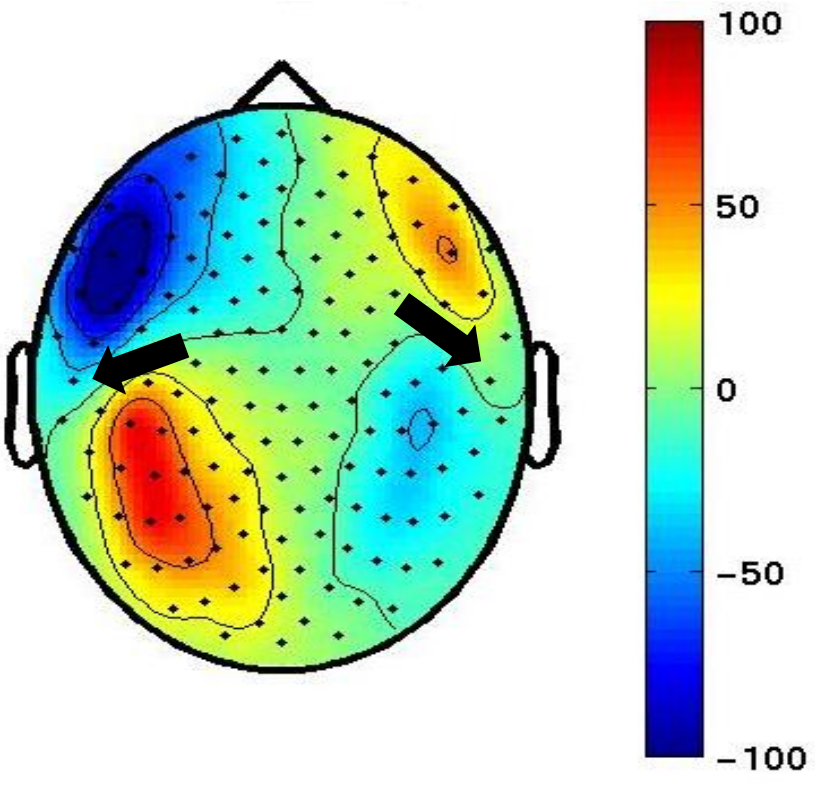
Elekta Neuromag



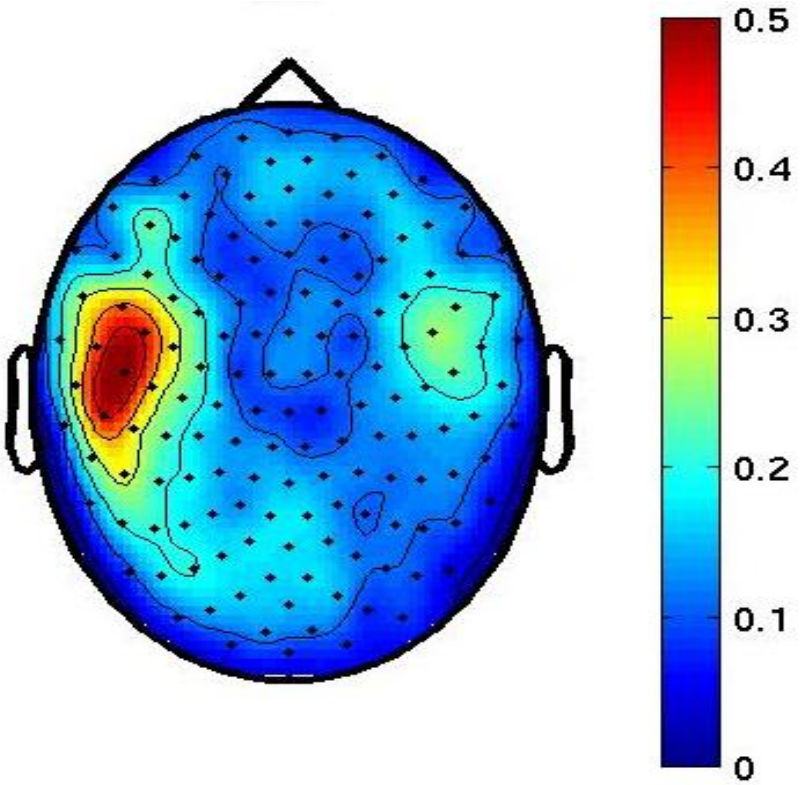
N400 response in MEG



N400 response - compared between MEG systems

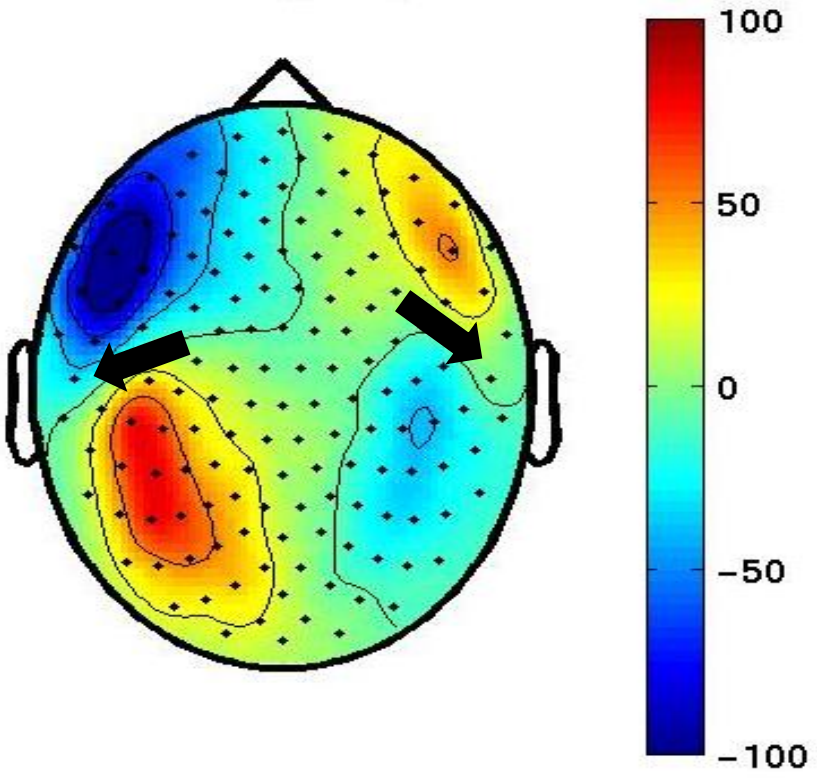


axial magnetic field

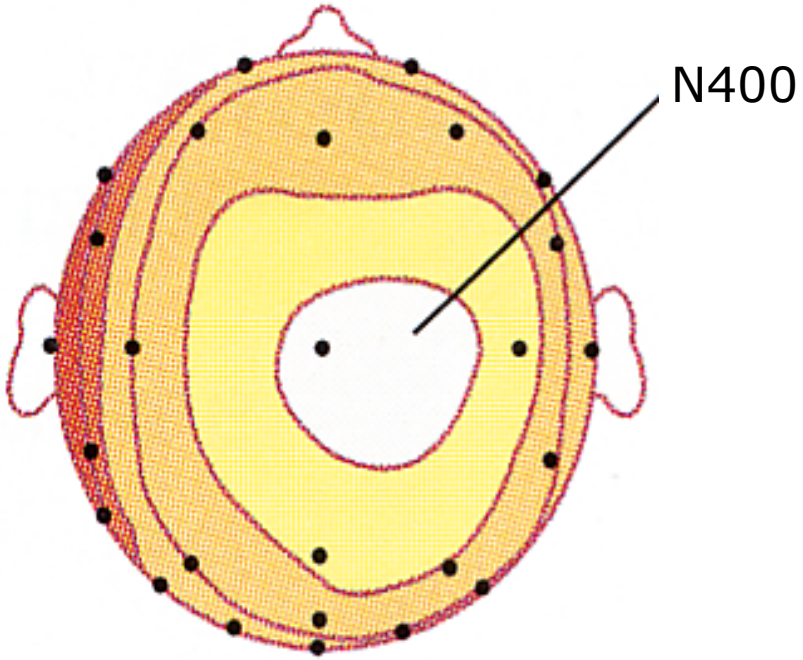


planar gradient

N400 response - compared between MEG and EEG



axial magnetic field



potential distribution

Talk outline

What kind of signals are generated in the brain

How do we record those signals

Analyzing those signals with FieldTrip

Background on the FieldTrip toolbox

M/EEG signal characteristics considered during analysis

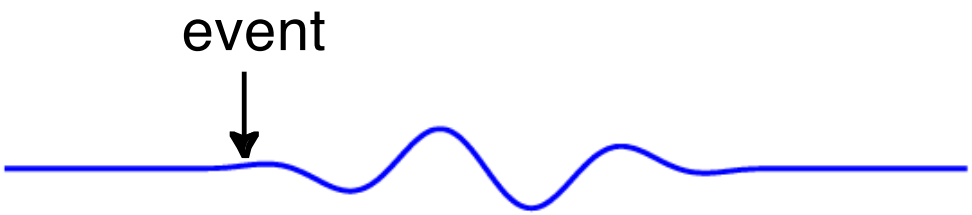
timecourse of activity
-> ERP

spectral characteristics
-> power spectrum

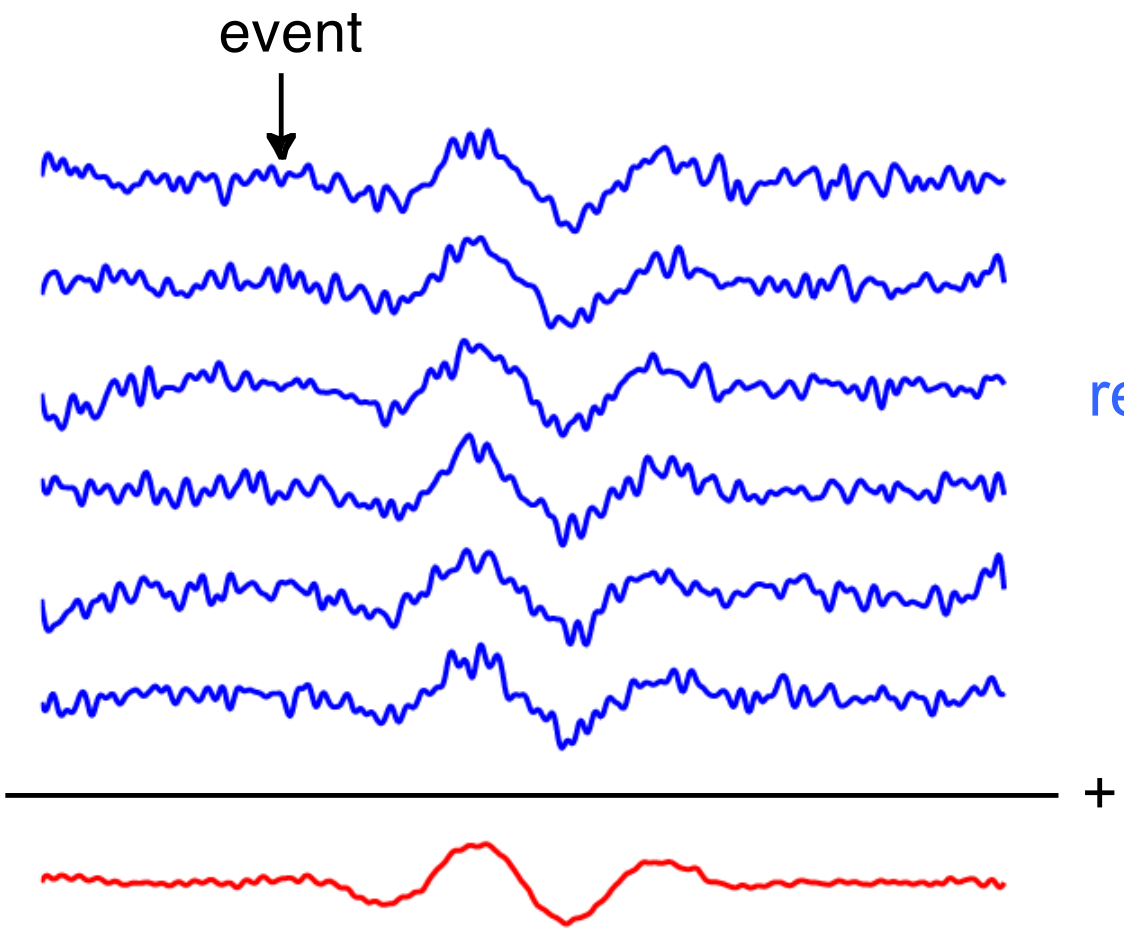
temporal changes in power
-> time-frequency response (TFR)

spatial distribution of activity over the head
-> source reconstruction

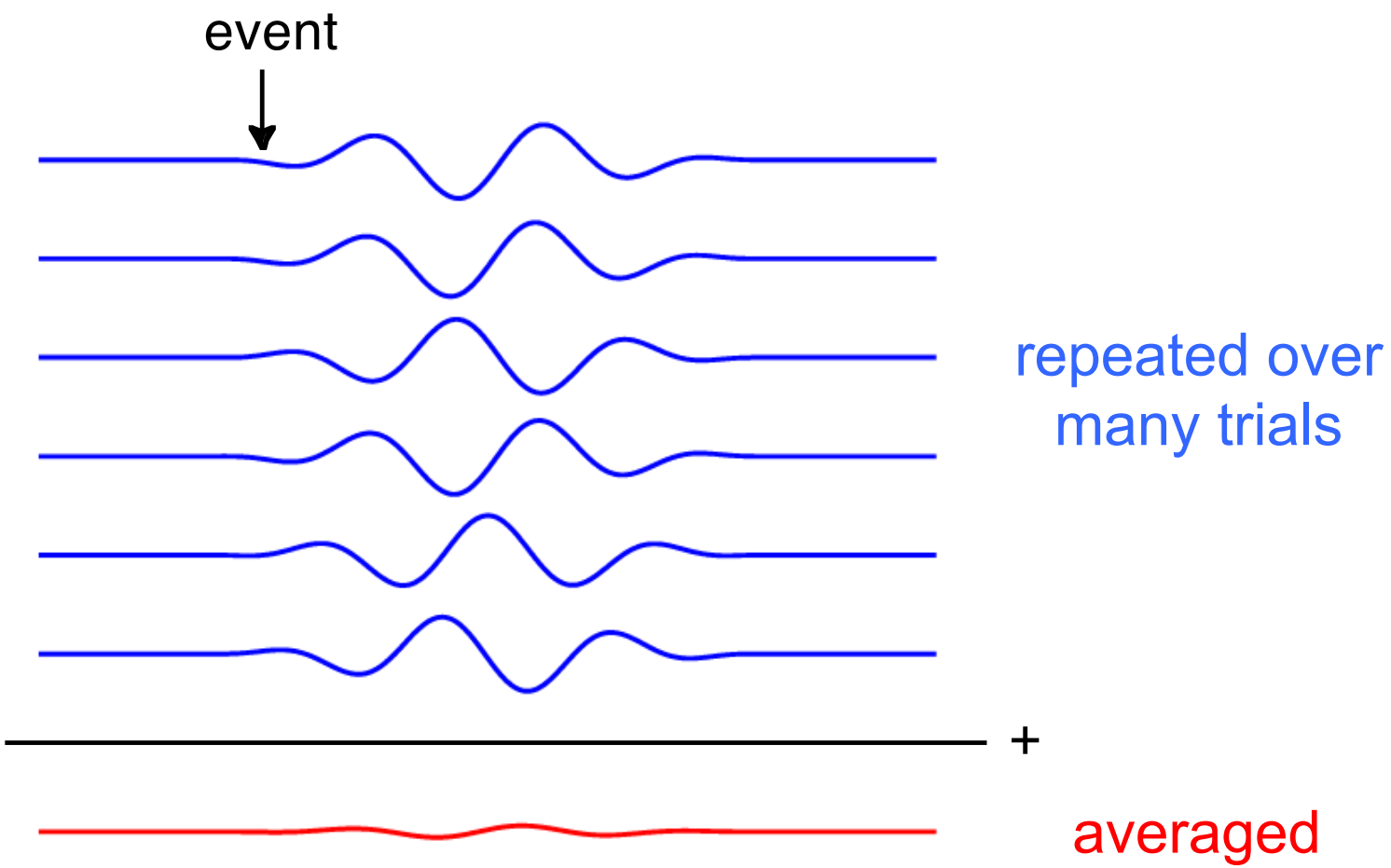
Evoked activity



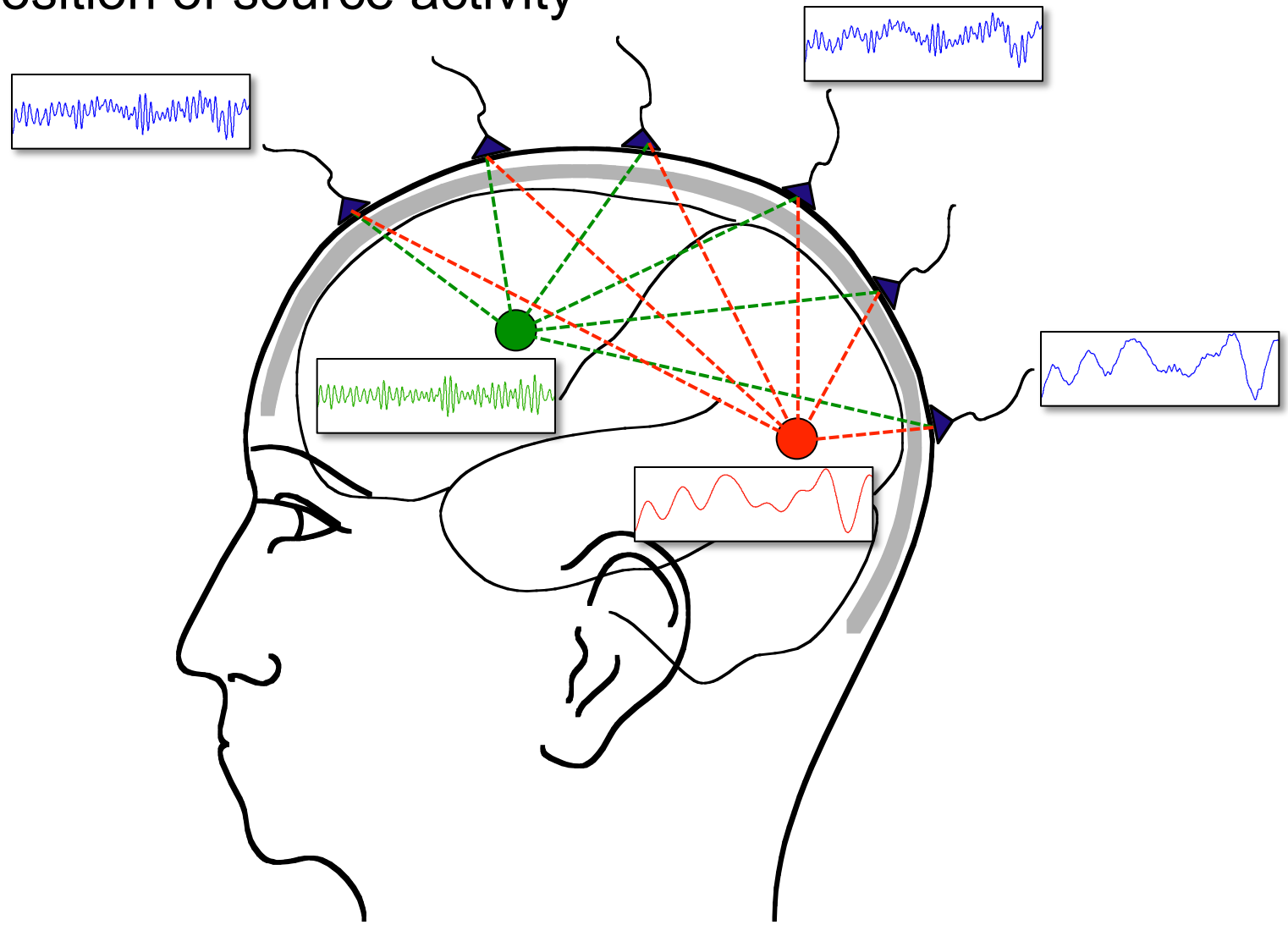
Evoked activity



Induced activity



Superposition of source activity



Separating activity of sources

- Use the temporal aspects of the data at the channel level
  - ERF latencies
  - ERF difference waves
  - Filtering the time-series
  - Spectral decomposition
- Use the spatial aspects of the data
  - Volume conduction model of head
  - Estimate source model parameters

Talk outline

What kind of signals are generated in the brain

How do we record those signals

Analyzing those signals with FieldTrip

Background on the FieldTrip toolbox

Some FieldTrip basics

dataout = functionname(cfg, datain, …)

functionname(

dataout = func

the “cfg” argument is a configuration structure, e.g.

```
cfg.channel = {‘C3’, C4’, ‘F3’, ‘F4’}
cfg.foilim = [1 70]
```

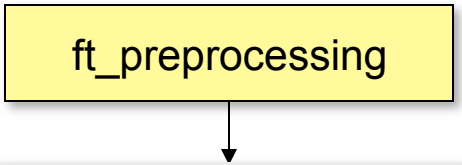
FieldTrip v.s. default Matlab

dataout = functionname(cfg, datain, …)

cfg.key1 = value1
cfg.key2 = value2

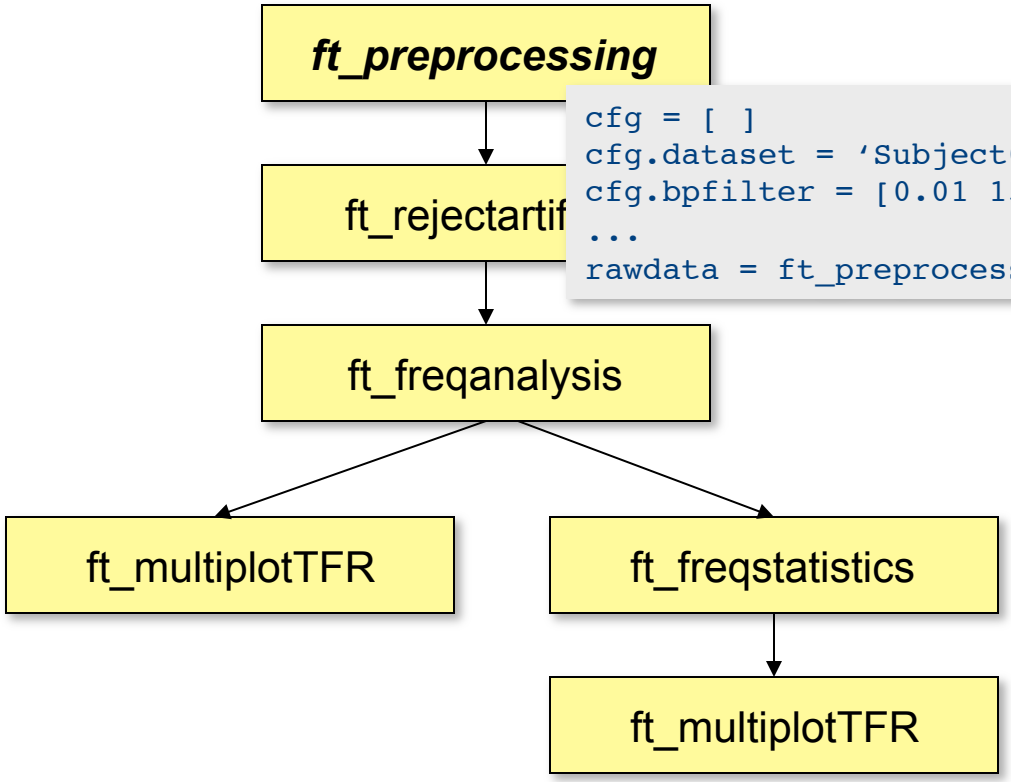
dataout = functionname(datain, ‘key1’, ‘value1’, …)

Using functions in an analysis protocol



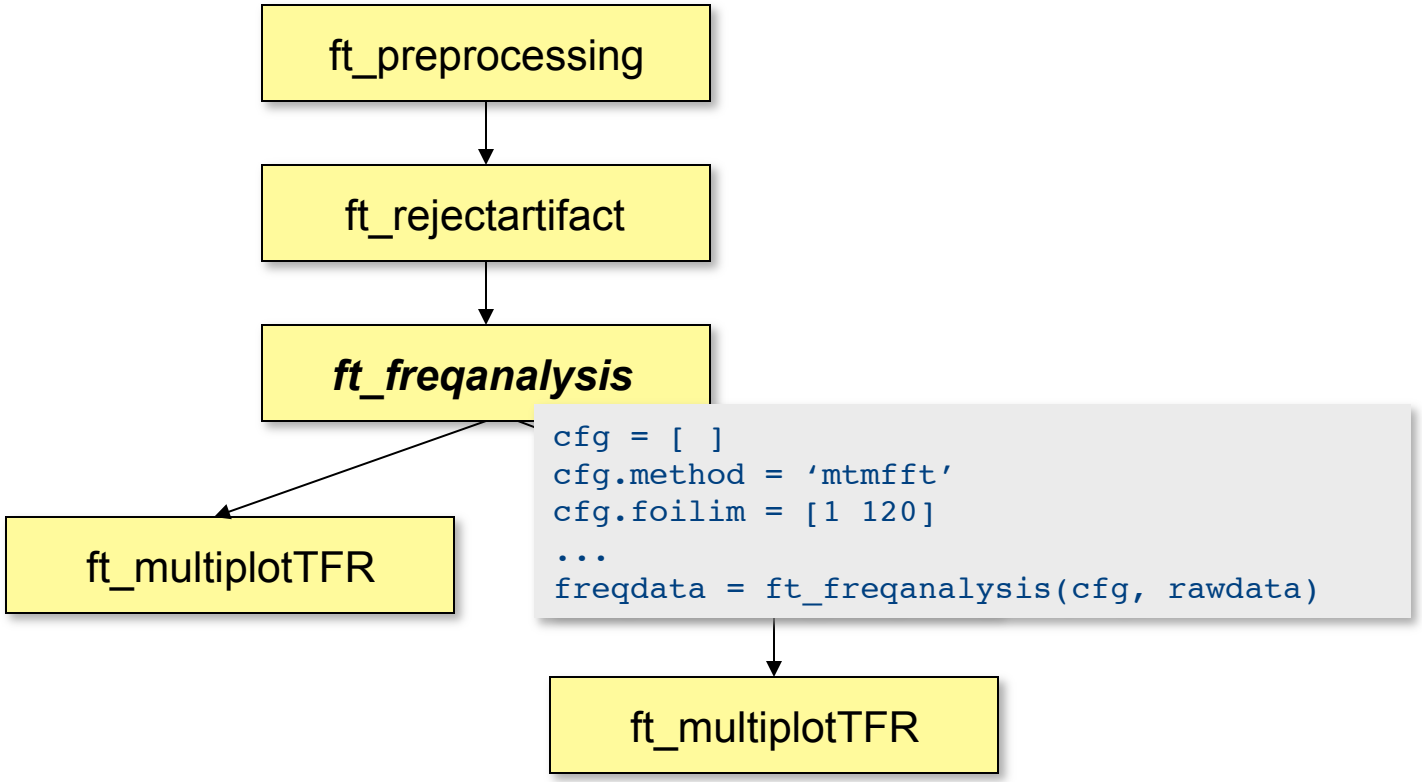

```
FT_PREPROCESSING reads MEG and/or EEG data according to user-specified
trials and applies several user-specified preprocessing steps to the
signals.

Use as
  [data] = ft_preprocessing(cfg)
or
  [data] = ft_preprocessing(cfg, data)

The first input argument "cfg" is the configuration structure, which
contains all details for the dataset filenames, trials and the
preprocessing options. You can only do preprocessing after defining the
segments of data to be read from the file (i.e. the trials), which is for
example done based on the occurence of a trigger in the data.

...
```

Using functions in an analysis protocol




```
cfg = [ ]
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)
```

Using functions in an analysis protocol



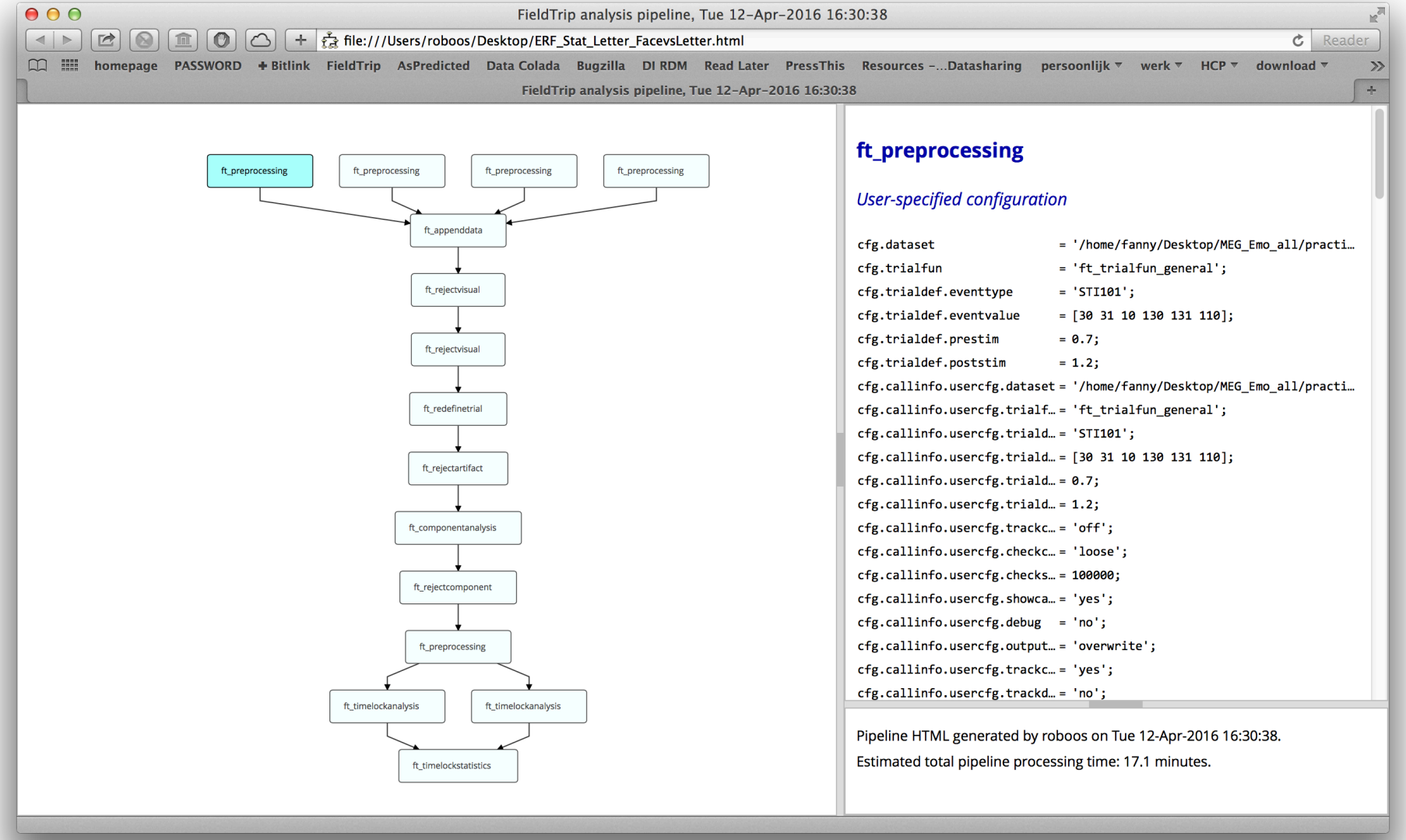
Raw data structure

```
rawData =
      label: {151x1 cell}
      trial: {1x80 cell}
       time: {1x80 cell}
    fsample: 300
        hdr: [1x1 struct]
        cfg: [1x1 struct]
```

Event related response

```
timelockData =
             label: {151x1 cell}
               avg: [151x900 double]
               var: [151x900 double]
              time: [1x900 double]
            dimord: 'chan_time'
               cfg: [1x1 struct]
```

Keeping track of your analysis

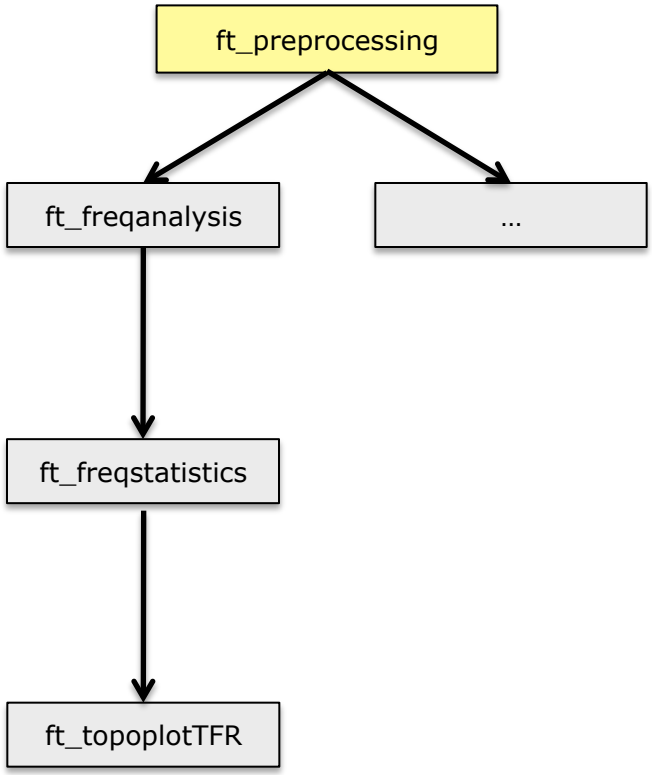


Example use in scripts

```
cfg = []
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)
```

```
cfg = []
cfg.method = 'mtmfft'
cfg.foilim = [1 120]
...
freqdata = ft_freqanalysis(cfg, rawdata)
```

```
cfg = []
cfg.method = 'montecarlo'
cfg.statistic = 'indepsamplesT'
cfg.design = [1 2 1 2 2 1 2 1 1 2 ... ]
...
freqstat = ft_freqstatistics(cfg, freqdata)
```

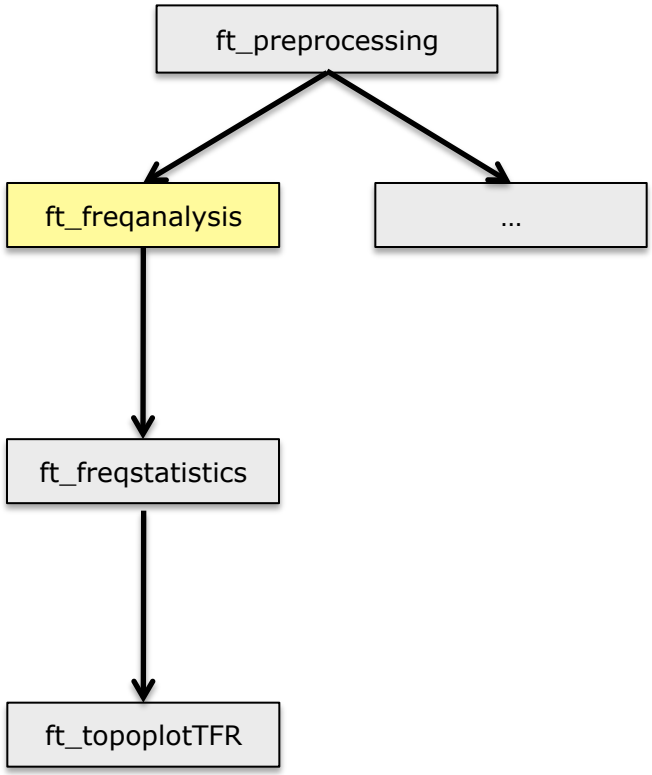


Example use in scripts

```
cfg = []
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)
```

```
cfg = []
cfg.method = 'mtmfft'
cfg.foilim = [1 120]
...
freqdata = ft_freqanalysis(cfg, rawdata)
```

```
cfg = []
cfg.method = 'montecarlo'
cfg.statistic = 'indepsamplesT'
cfg.design = [1 2 1 2 2 1 2 1 1 2 ... ]
...
freqstat = ft_freqstatistics(cfg, freqdata)
```

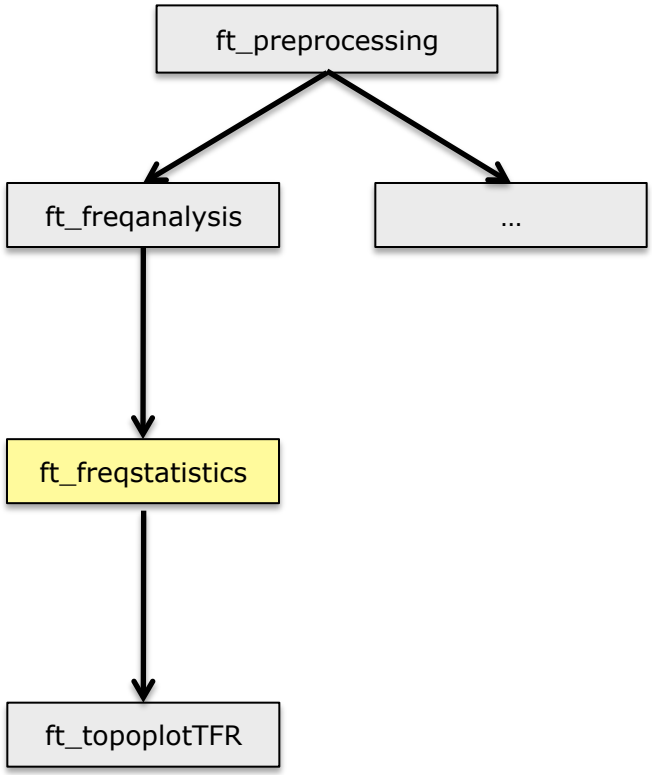


Example use in scripts

```
cfg = []
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)
```

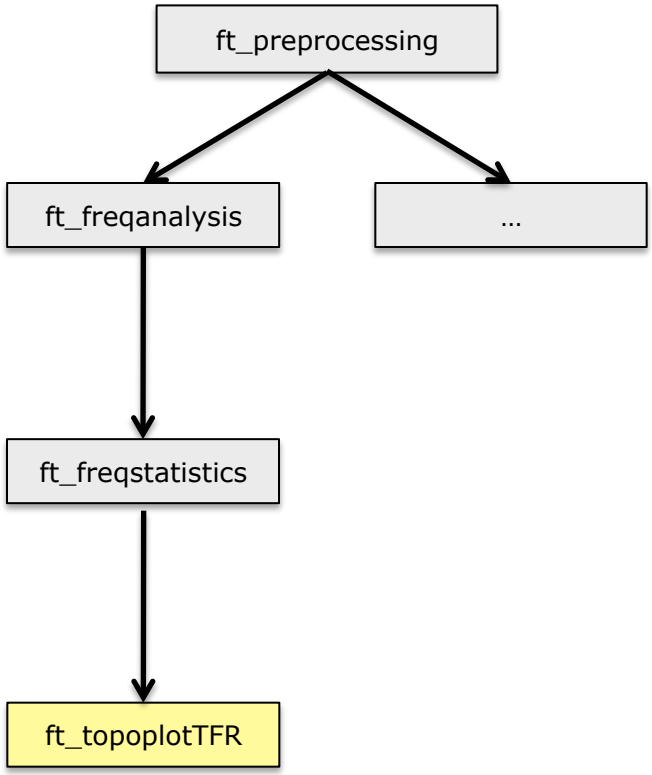
```
cfg = []
cfg.method = 'mtmfft'
cfg.foilim = [1 120]
...
freqdata = ft_freqanalysis(cfg, rawdata)
```

```
cfg = []
cfg.method = 'montecarlo'
cfg.statistic = 'indepsamplesT'
cfg.design = [1 2 1 2 2 1 2 1 1 2 ... ]
...
freqstat = ft_freqstatistics(cfg, freqdata)
```



Example use in scripts

```
cfg = []
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)


cfg = []
cfg.method = 'mtmfft'
cfg.foilim = [1 120]
...
freqdata = ft_freqanalysis(cfg, rawdata)


cfg = []
cfg.method = 'montecarlo'
cfg.statistic = 'indepsamplesT'
cfg.design = [1 2 1 2 2 1 2 1 1 2 ... ]
...
freqstat = ft_freqstatistics(cfg, freqdata)
```

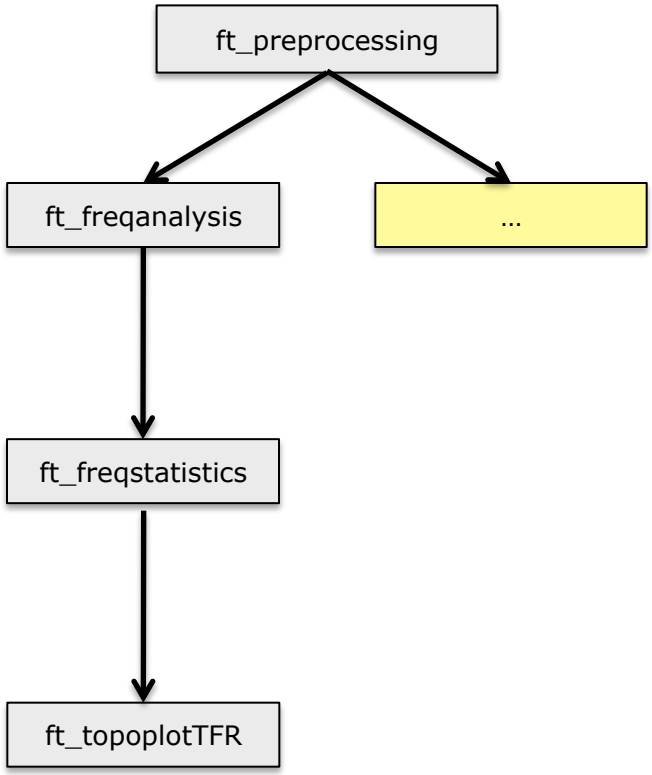


Example use in scripts

```
cfg = []
cfg.dataset = 'Subject01.ds'
cfg.bpfilter = [0.01 150]
...
rawdata = ft_preprocessing(cfg)
```

```
cfg = []
cfg.method = 'mtmfft'
cfg.foilim = [1 120]
...
freqdata = ft_freqanalysis(cfg, rawdata)
```

```
cfg = []
cfg.method = 'montecarlo'
cfg.statistic = 'indepsamplesT'
cfg.design = [1 2 1 2 2 1 2 1 1 2 ... ]
...
freqstat = ft_freqstatistics(cfg, freqdata)
```



Example use in scripts

```
subj = {'S01.ds', 'S02.ds', …}
trig = [1 3 7 9]

for s=1:nsubj
for c=1:ncond

  cfg = []
  cfg.dataset  = subj{s}
  cfg.trigger  = trig(c)
  rawdata{s,c} = ft_preprocessing(cfg)

  cfg = []
  cfg.method    = 'mtmfft'
  cfg.foilim    = [1 120]
  freqdata{s,c} = ft_freqanalysis(cfg, rawdata{s,c})

end
end
```

Example use in scripts

```
subj = {'S01.ds', 'S02.ds', …}
trig = [1 3 7 9]

for s=1:nsubj
for c=1:ncond

  cfg = []
  cfg.dataset  = subj{s}
  cfg.trigger  = trig(c)
  rawdata      = ft_preprocessing(cfg)


  filename = sprintf('raw%s_%d.mat', subj{s}, trig(c));
  save(filename, 'rawdata')


end
end
```

Example use in distributed computing

```
subj = {'S01.ds', 'S02.ds', …}
trig = [1 3 7 9]

for s=1:nsubj
for c=1:ncond

  cfgA{s,c} = []
  cfgA{s,c}.dataset    = subj{s}
  cfgA{s,c}.trigger    = trig(c)
  cfgA{s,c}.outputfile = sprintf('raw%s_%d.mat', subj{s}, trig(c))

  cfgB{s,c} = []
  cfgB{s,c}.dataset    = subj{s}
  cfgB{s,c}.trigger    = trig(c)
  cfgB{s,c}.inputfile  = sprintf('raw%s_%d.mat', subj{s}, trig(c));
  cfgB{s,c}.outputfile = sprintf('freq%s_%d.mat', subj{s}, trig(c));

end
end

dfeval(@ft_preprocessing, cfgA)
dfeval(@ft_freqanalysis,  cfgB)
```

Example use in distributed computing

```
subj = {'S01.ds', 'S02.ds', …}
trig = [1 3 7 9]

for s=1:nsubj
for c=1:ncond

  cfgA{s,c} = []
  cfgA{s,c}.dataset    = subj{s}
  cfgA{s,c}.trigger    = trig(c)
  cfgA{s,c}.outputfile = sprintf('raw%s_%d.mat', subj{s}, trig(c))

  cfgB{s,c} = []
  cfgB{s,c}.dataset    = subj{s}
  cfgB{s,c}.trigger    = trig(c)
  cfgB{s,c}.inputfile  = sprintf('raw%s_%d.mat', subj{s}, trig(c));
  cfgB{s,c}.outputfile = sprintf('freq%s_%d.mat', subj{s}, trig(c));

end
end

qsubcellfun(@ft_preprocessing, cfgA)
qsubcellfun(@ft_freqanalysis,  cfgB)
```

FieldTrip is a toolbox

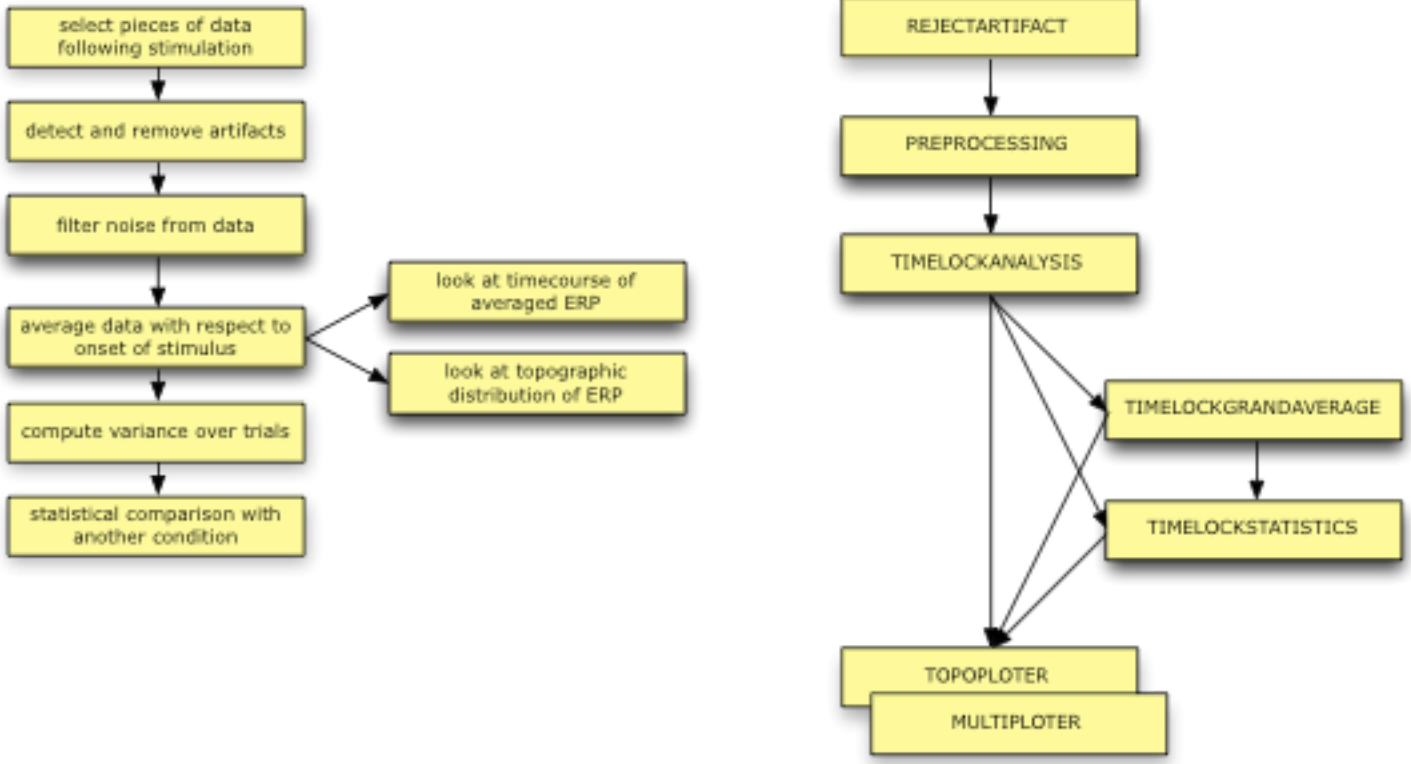
the data and the separate functions are in your hands

the scripts depend on the data properties, your computer and on your programming skills and style

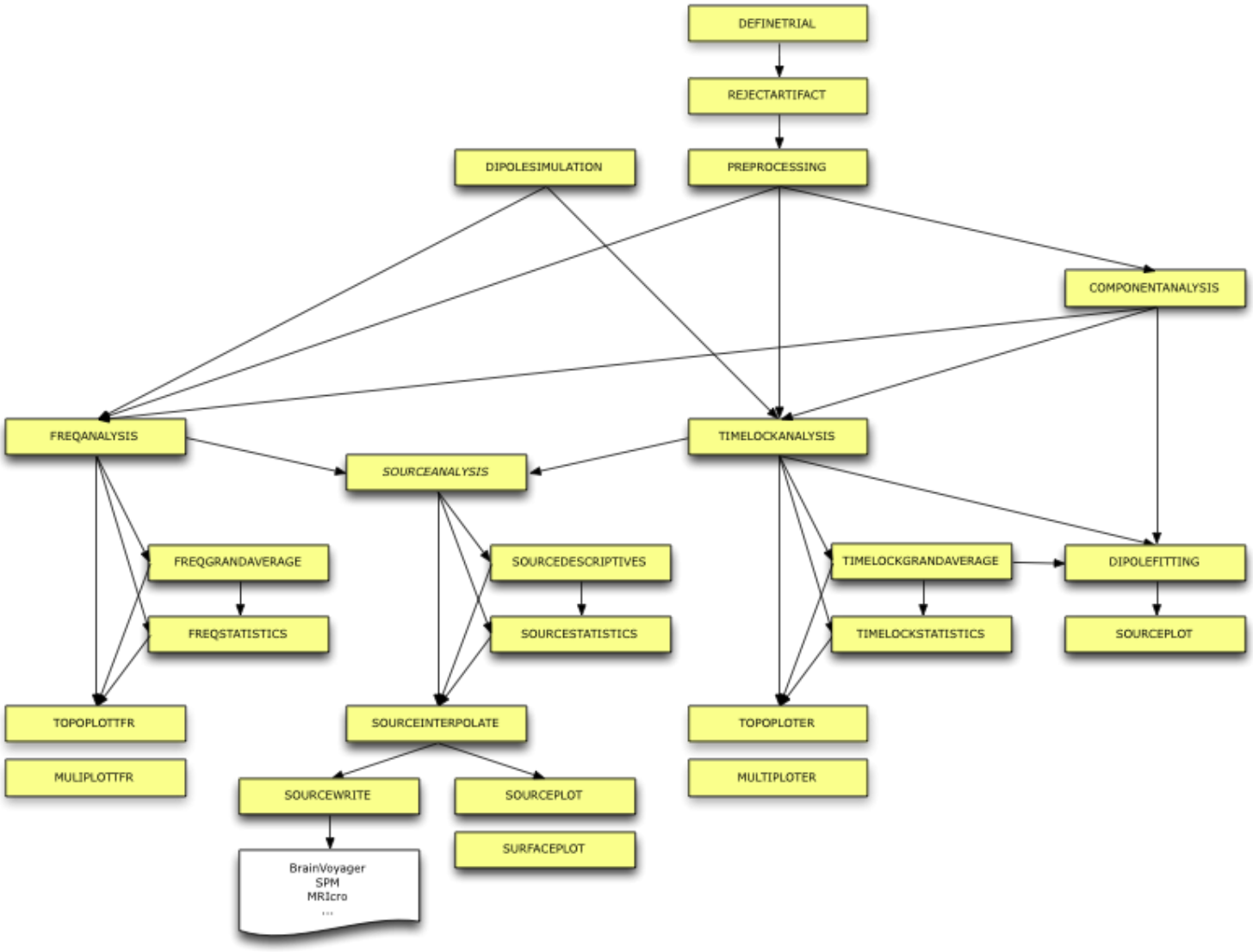
scripts correspond to analysis protocols

- scripts can be reviewed by supervisors
- scripts are often shared with colleagues
- scripts can be published/released

One-to-one mapping between analysis steps and toolbox functions



Overview of main functions



Finding your way around in the FieldTrip toolbox

Matlab

help functionname

edit functionname

Website

http://www.fieldtriptoolbox.org

Email discussion list

Expertise in your local group

Talk outline

What kind of signals are generated in the brain

How do we record those signals

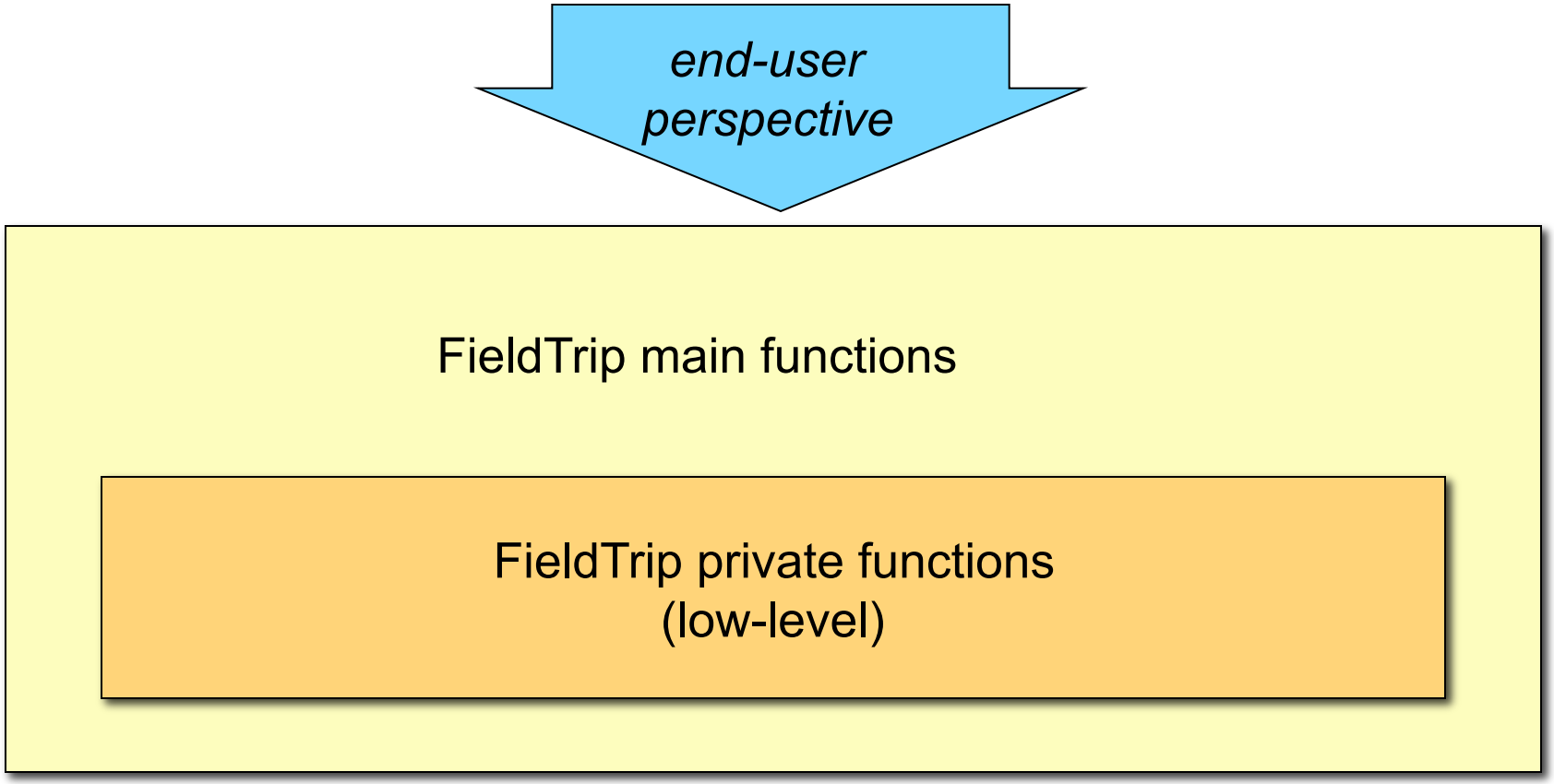
Analyzing those signals with FieldTrip

Background on the FieldTrip toolbox

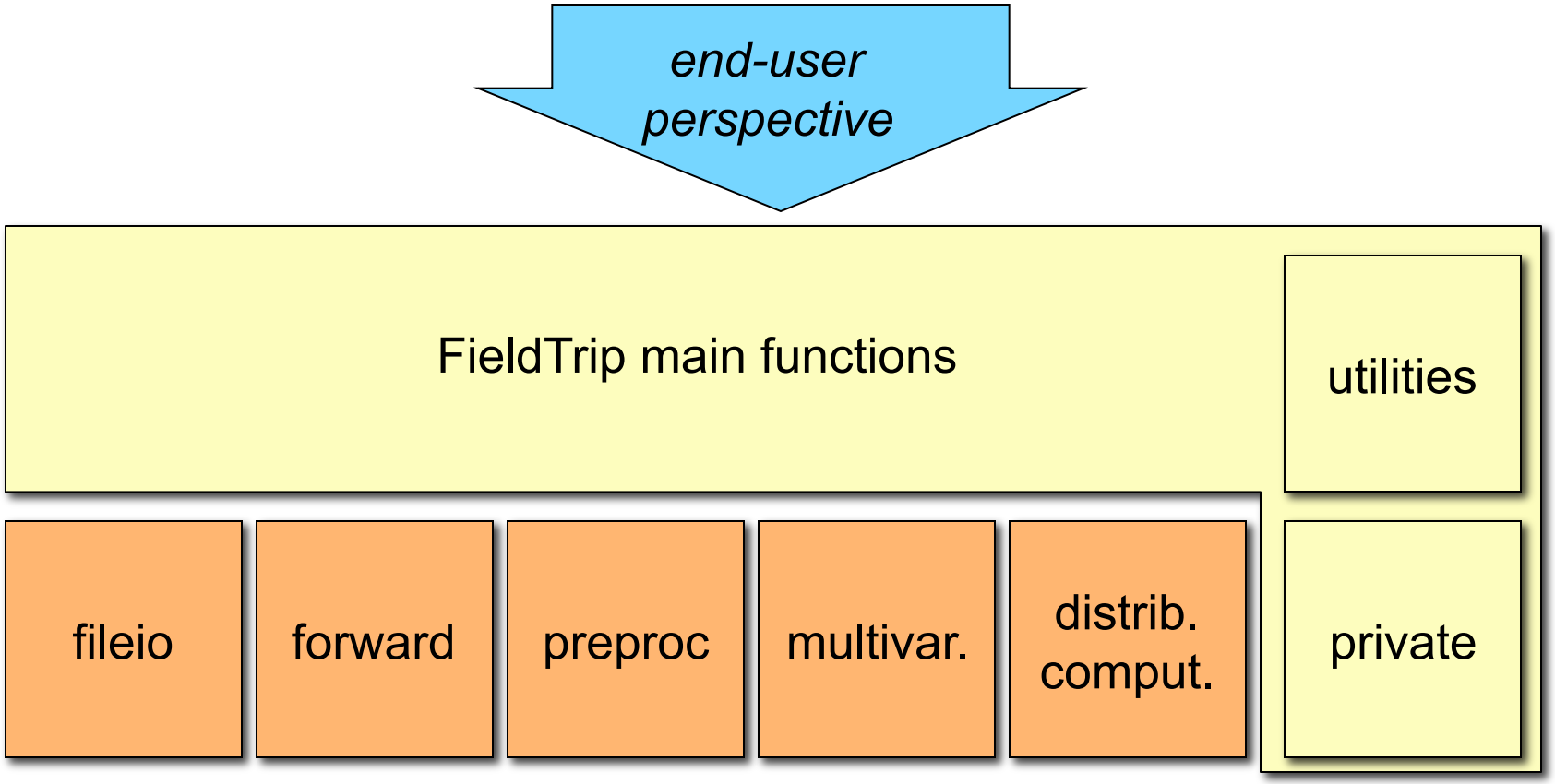
Who is the audience?

- experimental neuroscientists
  - no graphical user interface
  - more dedicated and ambitious researchers
- developers of other software packages
  - SPM
  - EEGLAB
  - BESA
  - BCI2000
- developers of analysis tools and methods
  - SIMBIO
  - OpenMEEG

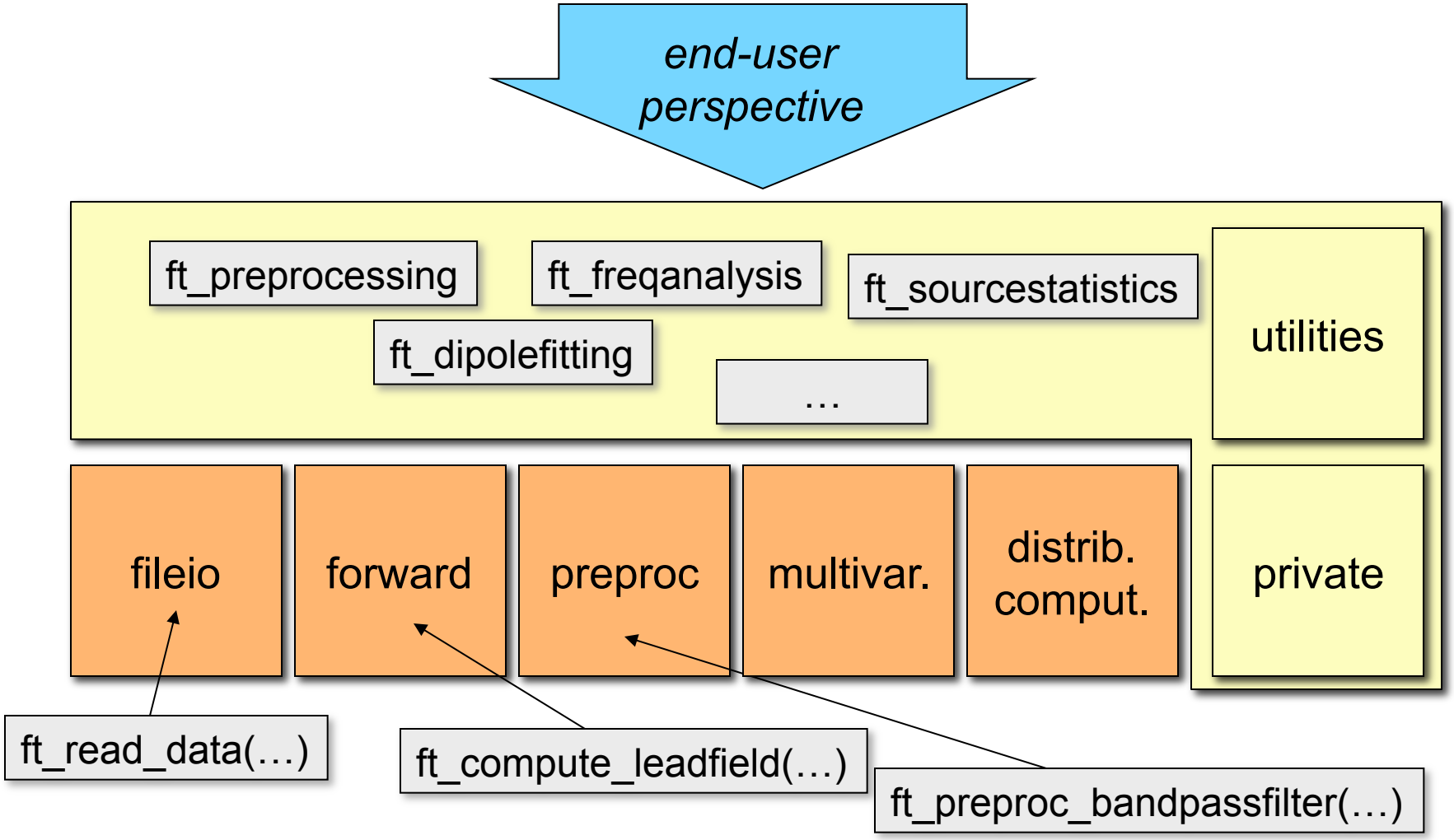
FieldTrip toolbox structure - at a glance



FieldTrip toolbox structure - a closer look



FieldTrip toolbox structure - a closer look



Summary

- What kind of signals are generated in the brain
- How do we record those signals
- Analyzing those signals with FieldTrip
- Background on the FieldTrip project

After lunch: hands-on

- Selecting segments of data
- Reading and preprocessing
- Averaging
- Plotting