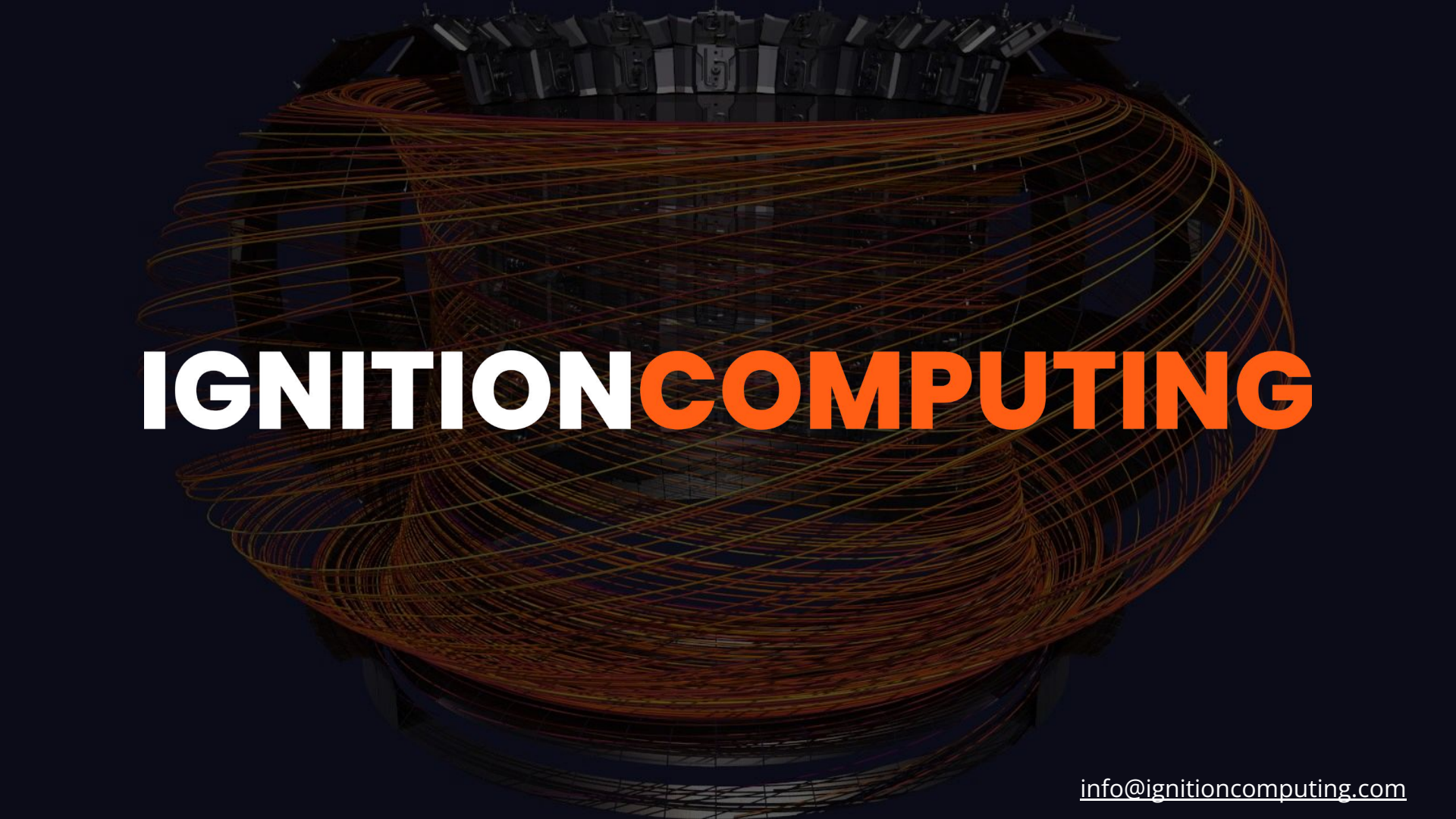


info@ignitioncomputing.com

IMAS-compatible Digital twins

2025-12-10

IGNITION
COMPUTING

About Us

Ignition Computing was founded in 2020 in Eindhoven, The Netherlands, to build the software powering science

- Boutique consulting firm for high-tech and scientific software
- Extensive track record with simulation development, numerical methods, ML, HPC, visualization and performance optimization
- Team of 9 with backgrounds in Fusion, Applied Physics, CS or maths
  - Now hiring!

iter
china eu india japan korea russia usa

PACIFIC FUSION

CelSian
Glass & Solar

SURF

Plasma Matters.

ProJoules

EUROfusion

DIFFER

Areas of work

- ITER Projects
  - Pulse Design Simulator
  - IMAS environment
- Tokamak Digital Twin
- Agriculture optimization
- FLASH MHD code improvement
- ML simulation acceleration

iter
china eu india japan korea russia usa

cea

UNIVERSITÉ CÔTE D'AZUR

EUROfusion

ProJoules

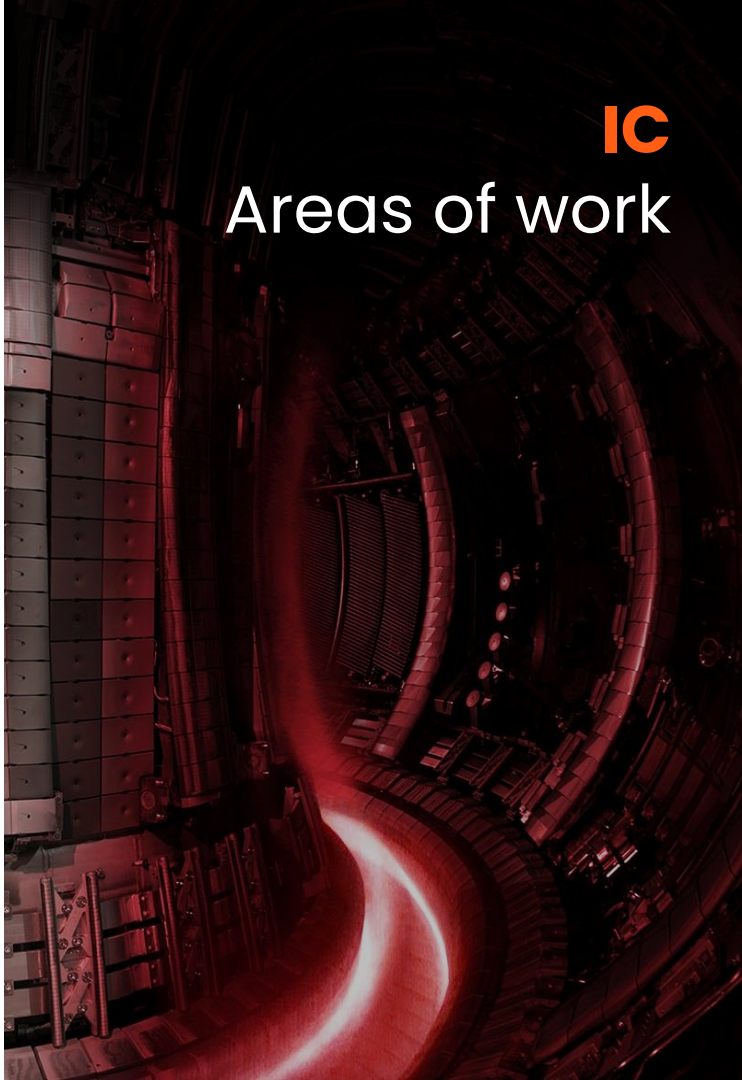
PACIFIC FUSION

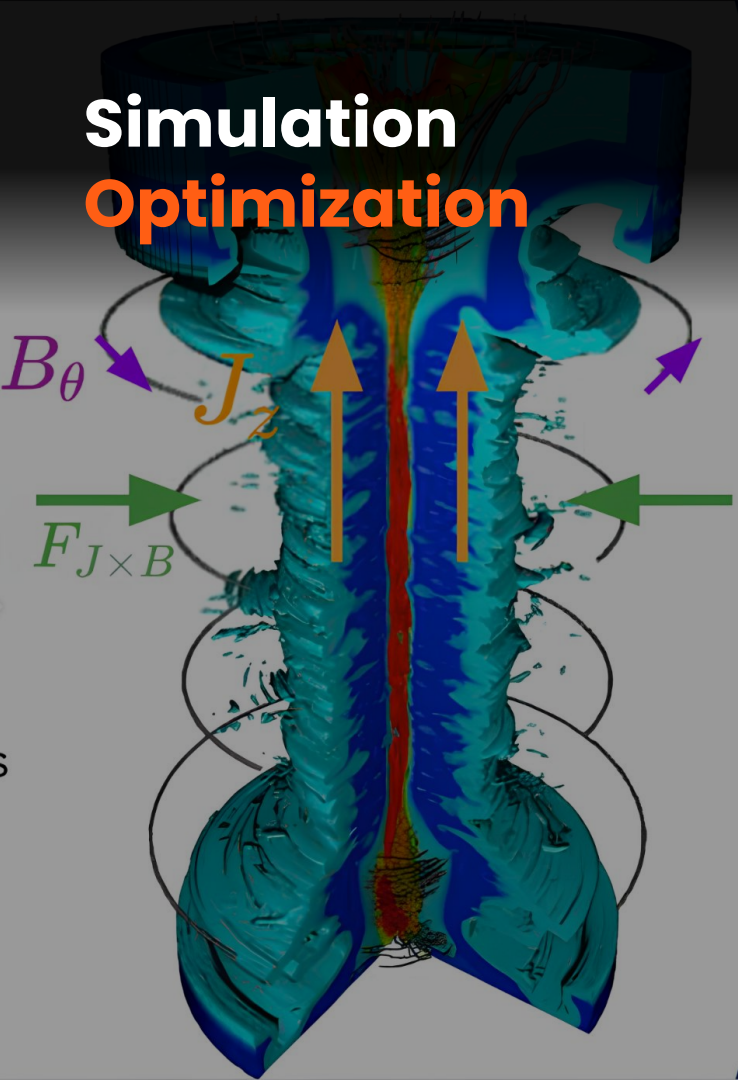
OpenFOAM

CelSian

Diabatix

Plasma Matters.





FLASH code improvements

Correctness

- Coarse-fine boundaries in **adaptive meshes**
- Improvements to **finite volume methods**
- Magnetic divergence cleaning methods

Stability

- **Backup options** for failed solves
- **Direct solver** implementation

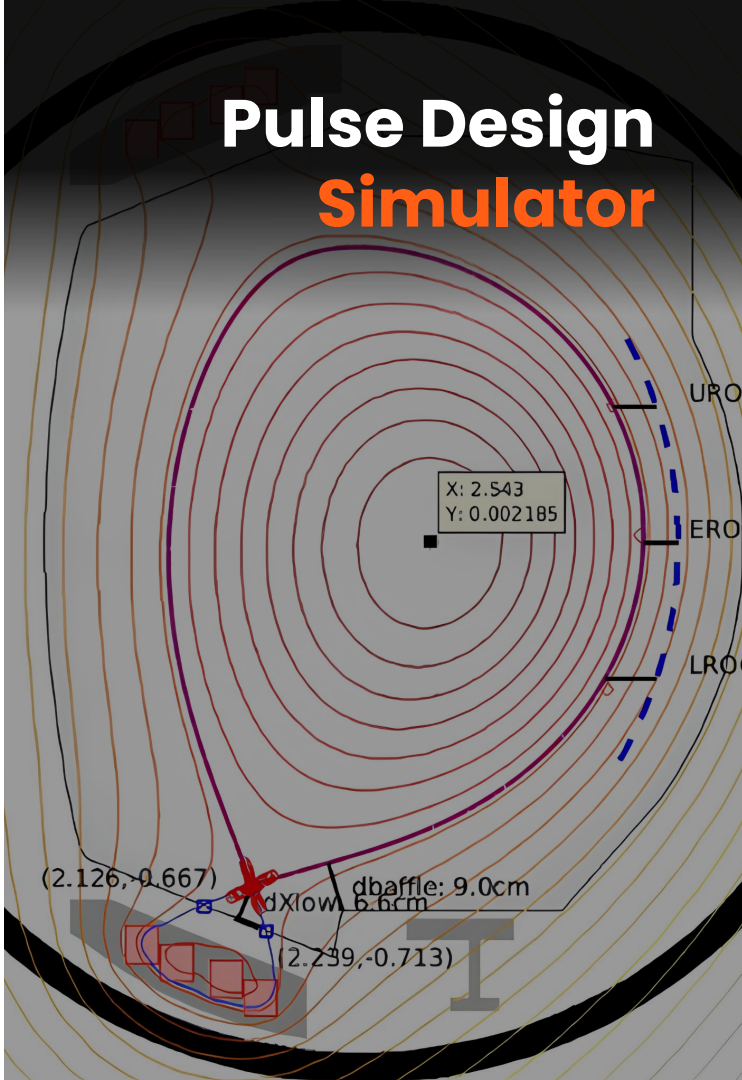
Performance

- Matrix solver optimization (~1.2x speedup)
- Preconditioner improvements (~1.4x+ speedup)
- EOS parallelization, improvements (~1.4x speedup)
- Code optimization (~1.3x speedup)
- Minor changes (~1.2x speedup)

PACIFIC FUSION

FLASH CENTER

- Combine fast, simple **physics simulators** with **control system**
- Design pulse schedule & control targets
- Transport (METIS and TORAX)
- Forward & inverse equilibrium (NICE)
- ITER PCSSP integration
- Compact waveform editor

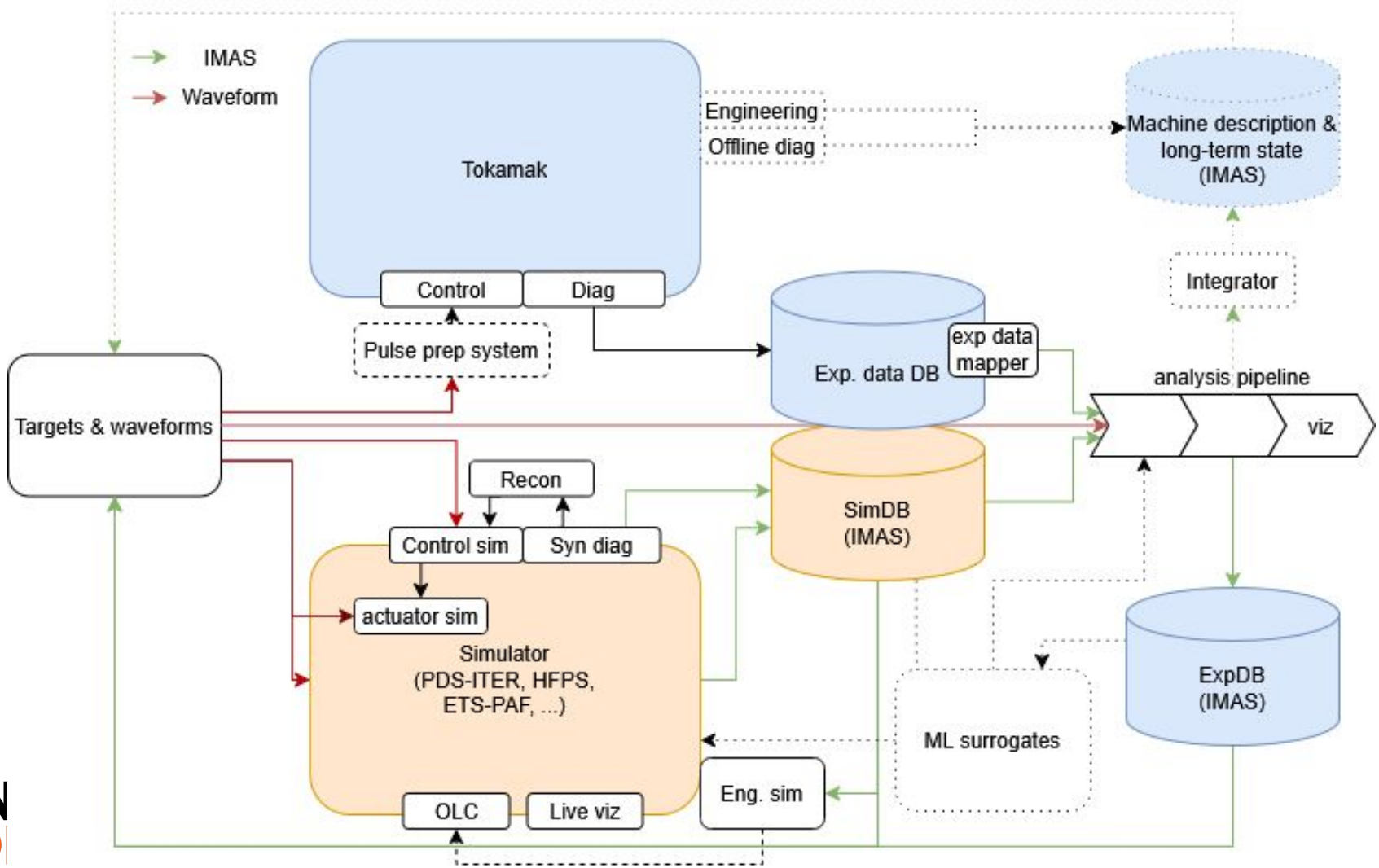


IMAS
Integrated Modeling and Analysis Suite

Contribute to IMAS standardized data model for fusion data

- Data Dictionary (DD) support, documentation
  - Array-based storage (xarray, netCDF)
- IMAS-Python access layer
- Serialisation & deserialisation
- IMAS-Validator
- IMAS-ParaView

iter
china eu india japan korea russia usa


IMAS
Waveform
Tokamak
Engineering
Offline diag
Machine description & long-term state (IMAS)
Integrator
Control
Diag
Pulse prep system
Exp. data DB
exp data mapper
analysis pipeline
Targets & waveforms
viz
Recon
SimDB (IMAS)
Control sim
Syn diag
actuator sim
Simulator (PDS-ITER, HFPS, ETS-PAF, ...)
ExpDB (IMAS)
ML surrogates
Eng. sim
OLC
Live viz

Waveform files – Example

- YAML tree structure
- Naming and structure of groups is arbitrary and may be decided by the user

- The waveforms names should refer to an existing IDS node

- Either a shorthand form may be used for constant waveforms:
  - `waveform: 12`
- Or a full waveform description:
  - `waveform:`
    `-{type: sine, amplitude: 10}`
    `-{from: 10, to: 0}`

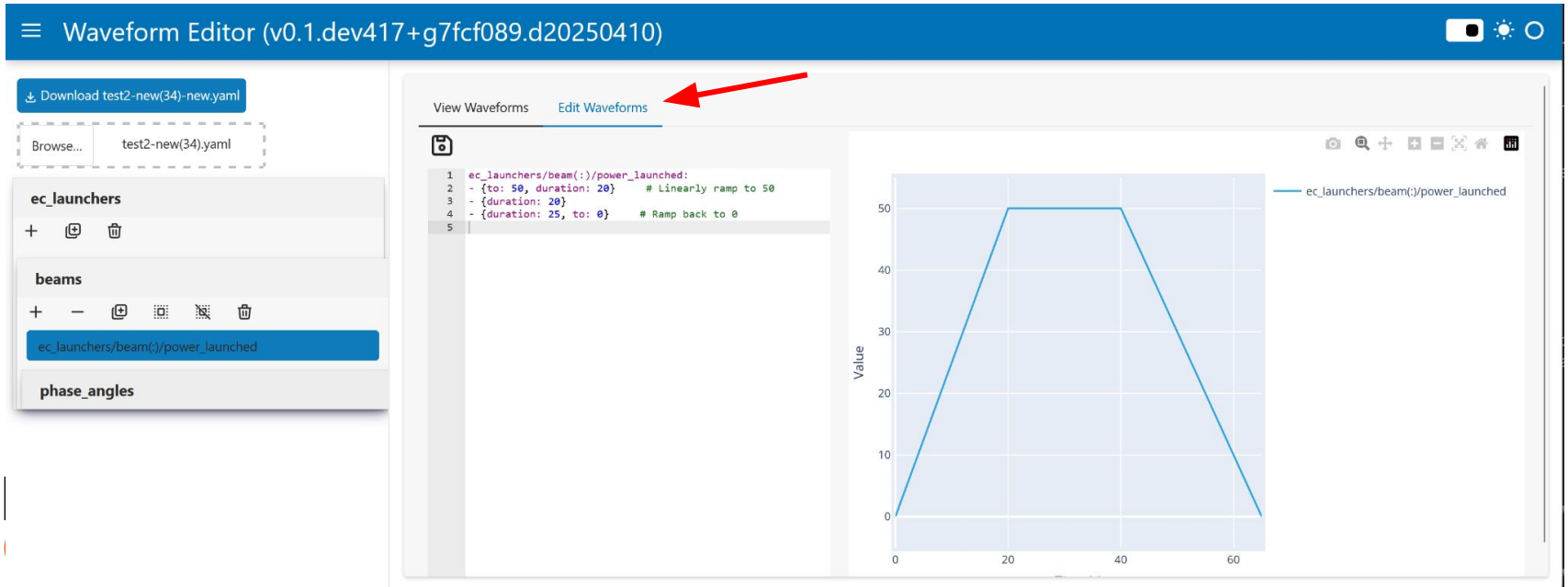
```yaml
ec_launchers:
  ec_launchers/beam(:)/frequency: 1.7e11
  beams:
    phase_angles:
      ec_launchers/beam(1)/phase/angle: -1.65898
      ec_launchers/beam(2)/phase/angle: -1.65379
      ec_launchers/beam(3)/phase/angle: -1.64701
    power:
      ec_launchers/beam(1:24)/power_launched:
      - {to: 8.33e5, duration: 20} # implicit linear ramp
      - {duration: 20} # constant at 8.33e5
      - {duration: 25, to: 0} # implicit linear back to 0
```

- Groups in green
- Waveforms in blue

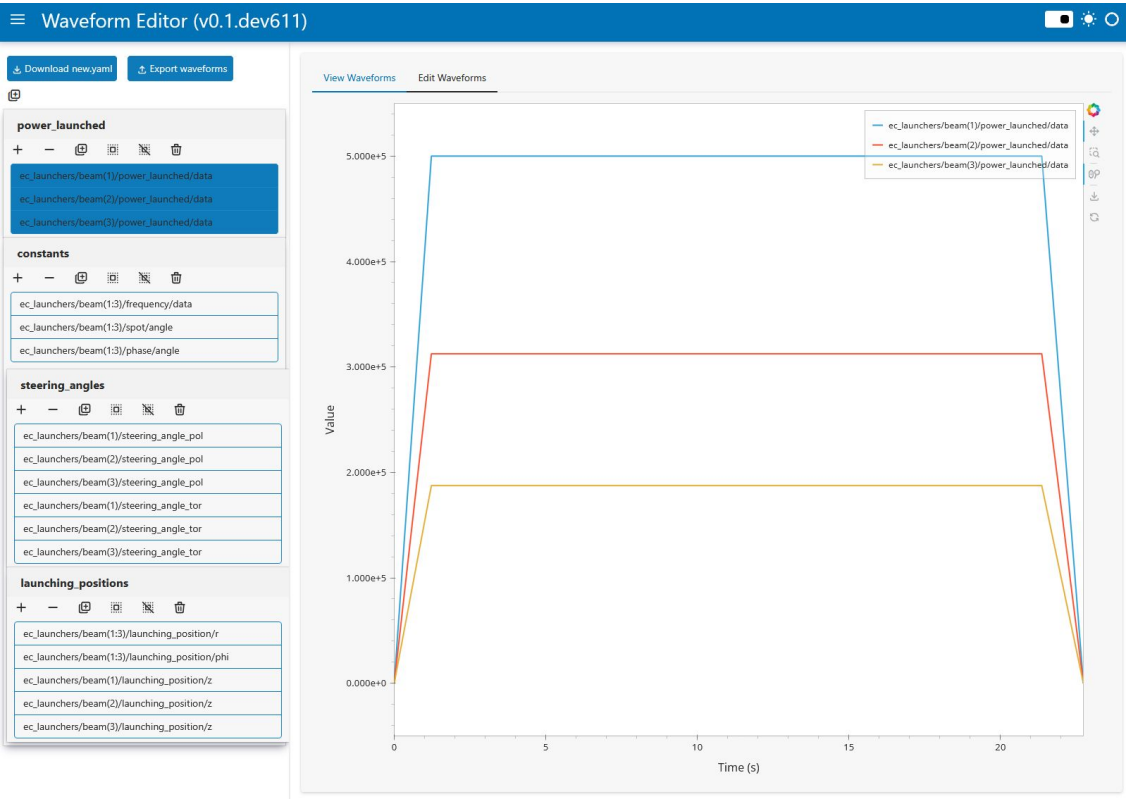
IGNITION COMPUTING

UI implementation – Editing

- Waveforms can be edited by selecting the “Edit Waveform” tab
- Waveforms will show updates in real-time
- When finished editing, press the -button to save the waveform



Example: ECRH for WEST




```yaml
globals:
  dd_version: 4.0.0
  machine_description:
    ec_launchers: imas:hdf5?path=/work/imas/shared/ima
power_launched:
  ec_launchers/beam(1)/power_launched/data:
  - {duration: 1.22528}
  - {type: constant, value: 5e+5, end: 2.136e1}
  - {to: 0, end: 2.27322e1}
  ec_launchers/beam(2)/power_launched/data:
  - {duration: 1.22528}
  - {type: constant, value: 3.125e+5, end: 2.136e1}
  - {to: 0, end: 2.27322e1}
  ec_launchers/beam(3)/power_launched/data:
  - {duration: 1.22528}
  - {type: constant, value: 1.875e+5, end: 2.136e1}
  - {to: 0, end: 2.27322e1}
constants:
  steering_angles:
    ec_launchers/beam(1)/steering_angle_pol: -0.1
    ec_launchers/beam(2)/steering_angle_pol: 0.0
    ec_launchers/beam(3)/steering_angle_pol: 0.15
    ec_launchers/beam(1)/steering_angle_tor: 0.2
    ec_launchers/beam(2)/steering_angle_tor: 0.29
    ec_launchers/beam(3)/steering_angle_tor: 0.02
  launching_positions:
    ec_launchers/beam(1:3)/launching_position/r: 3.53
    ec_launchers/beam(1:3)/launching_position/phi: 0.0
    ec_launchers/beam(1)/launching_position/z: 0.0
    ec_launchers/beam(2)/launching_position/z: 0.2
    ec_launchers/beam(3)/launching_position/z: -0.2
  ec_launchers/beam(1:3)/frequency/data: 1.05e11
  ec_launchers/beam(1:3)/spot/angle: 0.0
  ec_launchers/beam(1:3)/phase/angle: 0.0
```

Advanced Scenario Planning

- Take ‘impact’ into account in making scenarios
  - Neutron budget
  - Divertor heat load
- Risk-averaged values for nonlinear cost functions
  - Divertor temperature?
- Define ‘target’ / loss functions
  - Semi-manual (full one needed for machine optimization)
- Think about distributions, path in configuration space

Practical steps

- Harmonize and crosslink experimental data, IMAS expdb, Simdb and waveforms
- Extract reconstruction chains from simulators
  - Allow reuse, standardize on measurement definitions
- Standardize actuators and control system interfaces
  - Essential for PDS simulator - PCSSP interface
- Standardize controller configuration, scenario definition

IGNITION
COMPUTING

Long-term state

- Integrate over time, all pulses
  - Like total neutron load estimates
  - Divertor tile health
  - Mirror damage
- Offline diagnostics provide integrated measurements
- Engineering changes
  - Here is overlap with machine description database

IGNITION
COMPUTING

Multiple simulators

- PDS
- ETS-PAF?
- JINTRAC?
- FRAME?

Anything IMAS+MUSCLE3 is easy, non-muscle3 may be possible?

IGNITION
COMPUTING

Experiment preparation with the waveform editor

D. van Vugt, M. Schneider
S. Blokhuizen, M. Sebregts

IGNITION
COMPUTING

Goals

- Pulse configuration in a convenient way
  - Good user experience
  - Re-use, remix, share
- Review pulse schedules before running
  - Code review, for pulse schedules!
  - Run automated checks, CI test!
- Use same configuration and tools between simulation and experiment
- Easy to train people for SL
- Reproducibility of past pulses
  - Deterministic configuration

IGNITION
COMPUTING

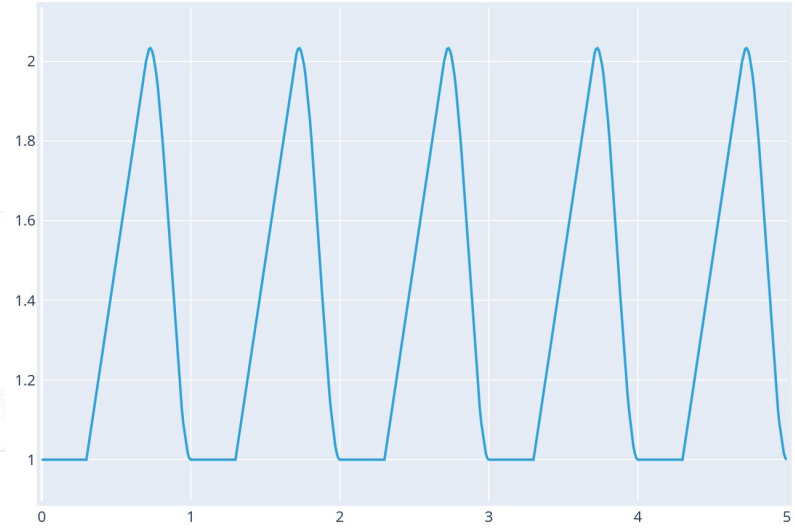
yaml-based waveform files

- New format built in PDS Waveform editor project
- Tendency-based and compact:
  - linear, sine, smooth
- Calculations
  - Derive waveforms from eachother
- Text-based!
  - Line-based diffs
  - LLM generatable?
  - Copy-paste
  - Share by chat

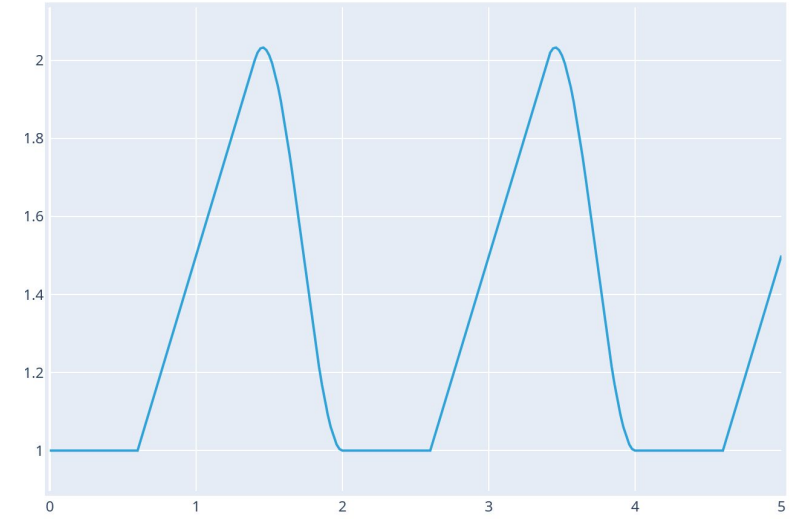
IGNITION
COMPUTING

Scaling waveforms – Example

```
1  waveform:
2  - type: repeat
3    duration: 5
4    waveform:
5    - {type: constant, value: 1, duration: 0.3}
6    - {type: linear, from: 1, to: 2, duration: 0.4}
7    - {type: smooth, duration: 0.3}
```




```
1  waveform:
2  - type: repeat
3    duration: 5
4    period: 2
5    waveform:
6    - {type: constant, value: 1, duration: 0.3}
7    - {type: linear, from: 1, to: 2, duration: 0.4}
8    - {type: smooth, duration: 0.3}
```



IGNITION
COMPUTING

Internships available! Topics roughly:

- Preconditioner reusability
- MUSCLE3 high-order time stepping
- Pulse Design Simulator extensions
- Live visualisation pipelines
- …

Additional internship & travel support available through FuseNet!

Contact us at
careers@ignitioncomputing.com

FuseNet
The European Fusion Education Network