

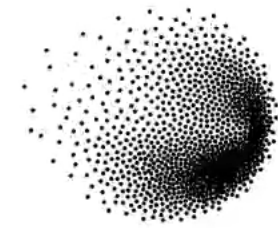
OPALX Retreat

CIEMAT Madrid

September 1 2026

OPALX - Retreat - Welcome

- Thanks to the LOC in particular to Pedro
- Format
 - we start the morning (9.00) with wrap-ups of the day before
 - then discussion / presentations
 - lunch (12.00)
 - working groups (open end)



PSI

OPALX Retreat Madrid September 1 - 4



	Tuesday	Wednesday	Thrusday	Friday
9-12	Welcome / Into :: (Pedro & AA) Presentation of goals :: group leader Input :: Quo-Vadis H5Hut & I/O in General (Sri) Input :: CI/CD (John) Input :: Misc - pyOPALTools, LCO (AA)	Input :: OPALX Paper (Miachil) Input :: Solver redesign (Alex) Input :: Manual / AI Assistant (AA)	Status Linac (Michail) Status Ring (Jon) Status Secondaries (Ryan)	working on projects Wrap-up :: presentation of achievements (Sri)
12-13.30	Lunch	Lunch	Lunch	Lunch
13.30 - xx.xx	working on projects	working on projects Lab tour t.b.d	working on projects	Input :: Status and Future of OPAL (Pedro) Input :: OPALX release roadmap (AA) Wrap-up :: presentation of achievements (group leaders) 15.00 adjourn
	Group I/O : Sri (lead), Severin, John, AA Group CI/CD : John Group Solver : Alex (lead), Severin, AA Group Linac : Michail Group Ring : Jon (lead), Chris Group Secondaries : Ryan Input :: max 20' - seed discussion		20.00 Dinner @ Terracotta Plaza de españa 	

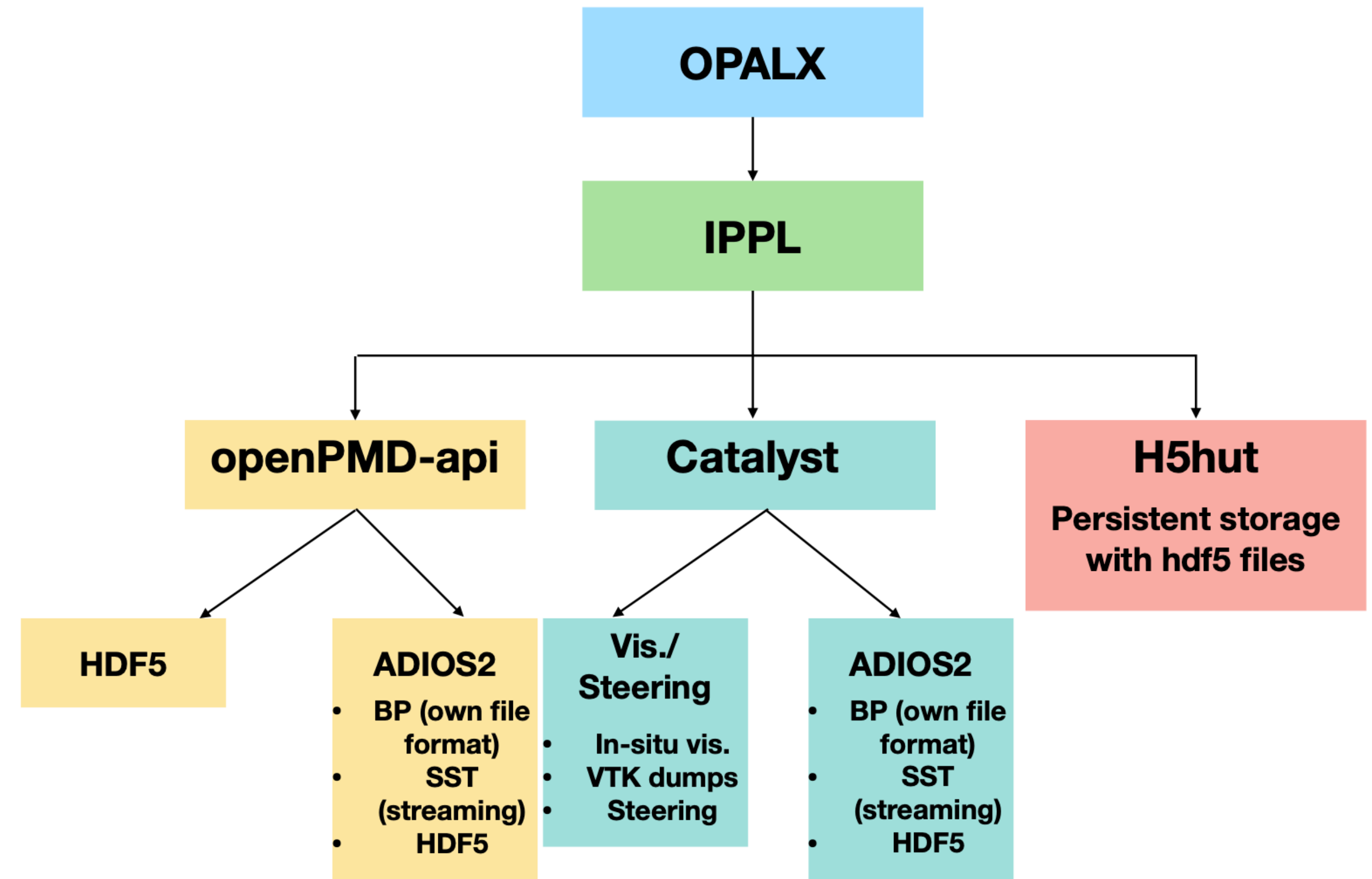
AA - Goals

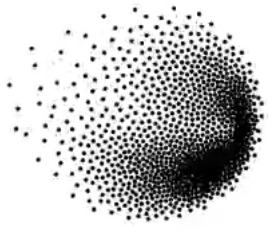
- Overarching
 - OPALX V 1.0 - timeline and who is doing what
 - OPALX Paper - timeline and who is doing what
 - What data do we need for production-proposals at CSCS/LUMI
 - OPALX QA (solver-design, CI/CD, I/O)
- my personal
 - Housekeeping (pyOPALTools, runOPAL, other converters) & low-level QA
 - Manual and/or AI prompt **opalx.cscs.ch**

I/O for OPALX

Goals for the retreat

1. With a minimal IPPL example (e.g. TestScatter) investigate three possible routes for the persistent I/O in IPPL/OPALX
2. **H5hut:** The legacy option in OPAL (John/Andreas)
3. **Catalyst->ADIOS2->HDF5/BP:** Based on the existing catalyst in-situ interface in IPPL (Severin)
4. **openPMD-api->ADIOS2->HDF5/BP:** Interface openPMD-api with IPPL (Sri)
5. We are interested in: Performance, scalability, dependencies and maintenance





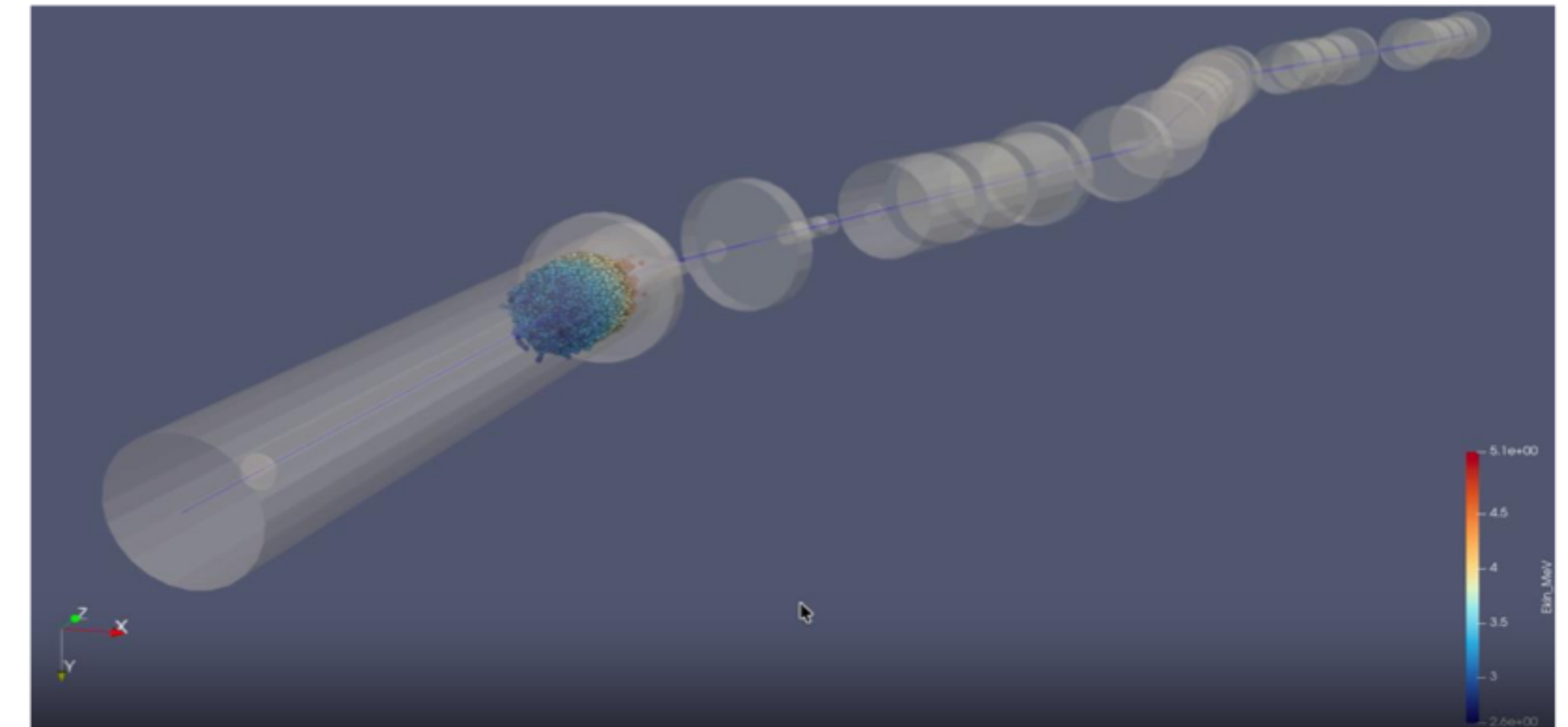
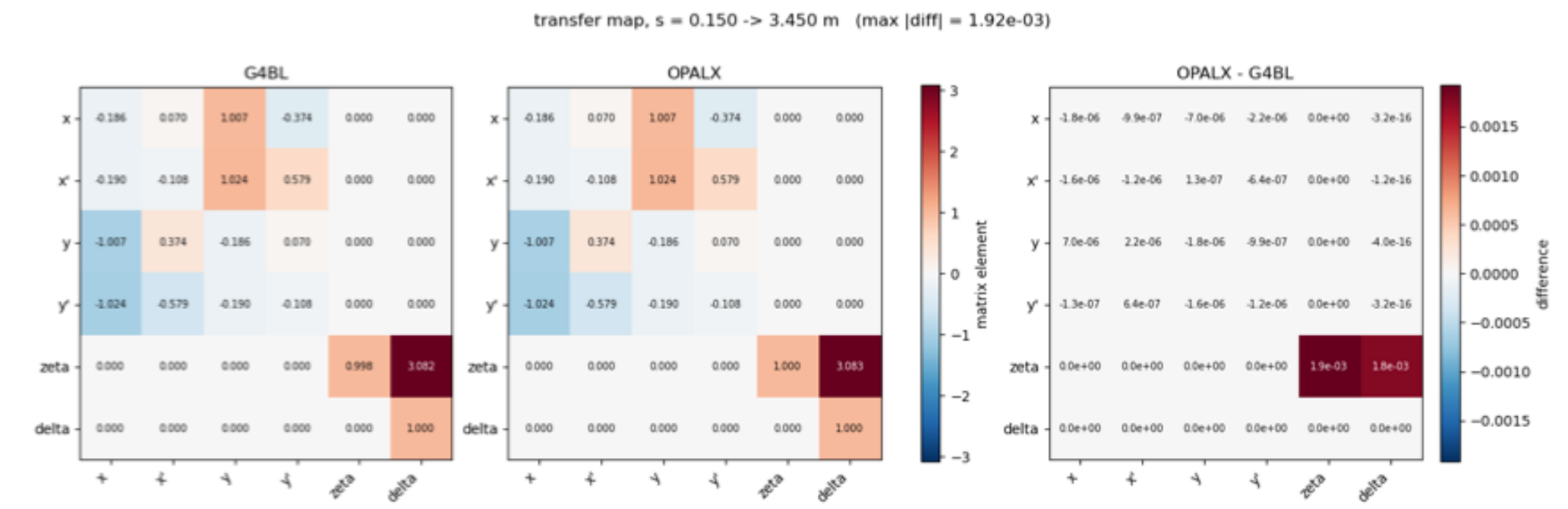
PSI

Unified Element Framework



Goals

- Unify Element Handling
 - MultipoleT (Jonathan, Chris)
 - BeamBeam (Andy, Alex)
 - RF Cavity (Michalis, Alex)
 - Ring (Jonathan, Chris)
 - Orbitthreading through an element multiple times
- Transfer Matrices
 - Generate transfer matrices of all element during the orbitthreading
- Visualization of the Elements (Severin, Andy, Sri)
 - Generate nice paraview enabled vtk output

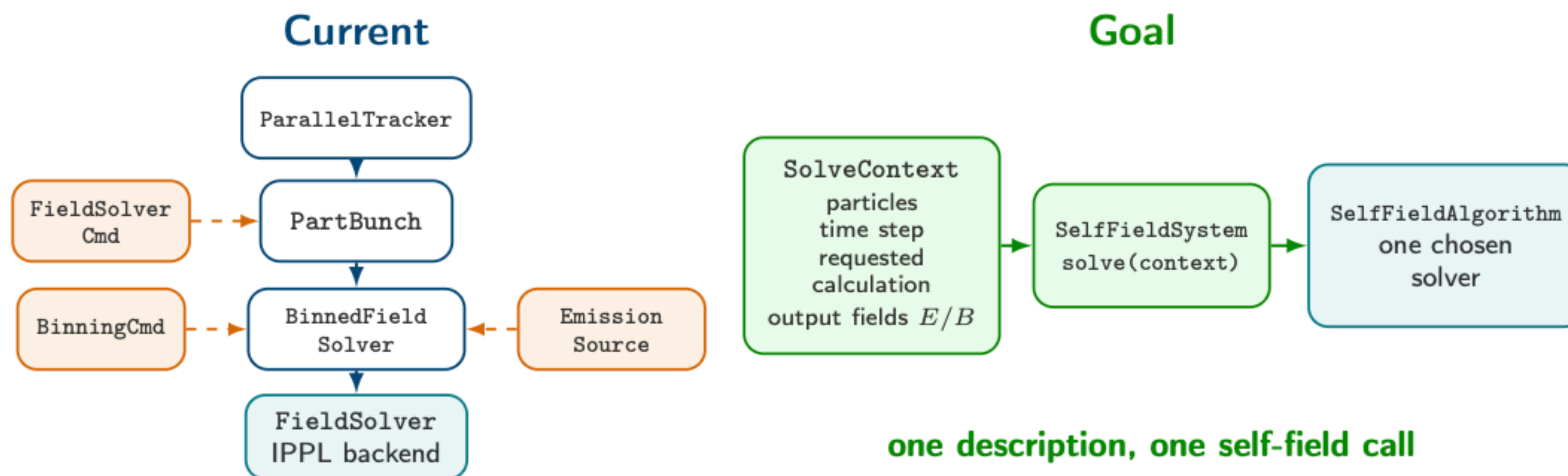


Team Ryan

Task: Ring

- Discovering what is available with respect to element layout and filling in anything missing.
- Extending the layout system to the MultipoleT element if necessary.
- Testing the 2.5D solver in this scenario.
- Laying out the ISIS ring and modelling the acceleration cycle (I have this for old Opal).
- Port FFA elements from old Opal.
- Re-implement Python interface without using Boost.
- Add rectangular grounded walls boundary condition 2D Poisson solver (using discrete sine transform?) for the 2.5D solver.

One context defines each self-field calculation



Takeaway: Each self-field calculation is described by one `SolveContext`. This design lets us add new solvers and specify the features each one supports.

Continuous Benchmarking for IPPL/OpalX

Catch performance regressions before they merge

- Run a small, representative benchmark on every PR across the three CSCS platforms.
- Submit results to **bencher.dev** and alert when a kernel regresses beyond a statistical threshold.
- Status: implemented on `continuous-benchmarking` ; data is flowing from Eiger, Daint GH200, and Beverin MI300.