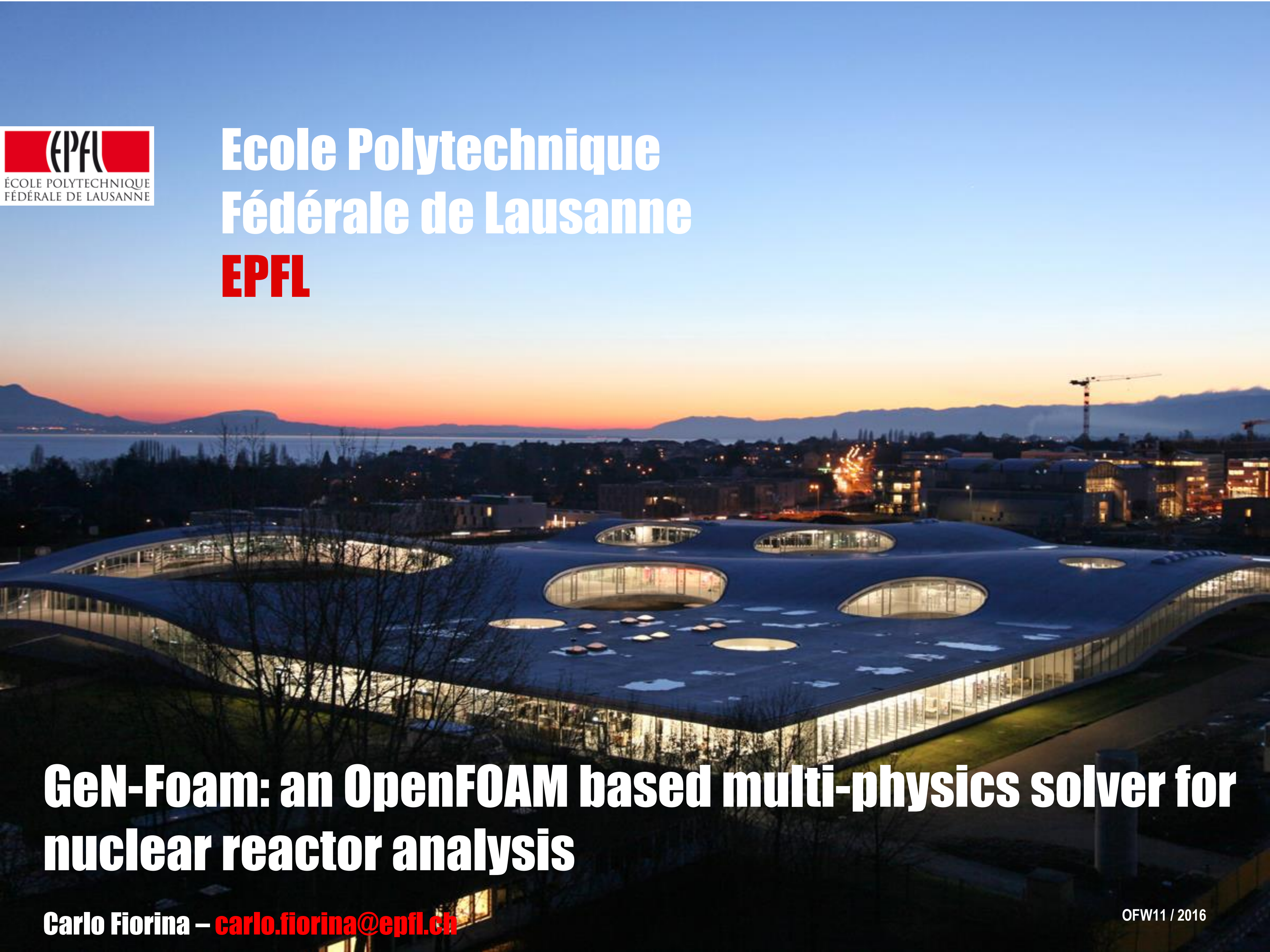


ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Ecole Polytechnique
Fédérale de Lausanne
EPFL

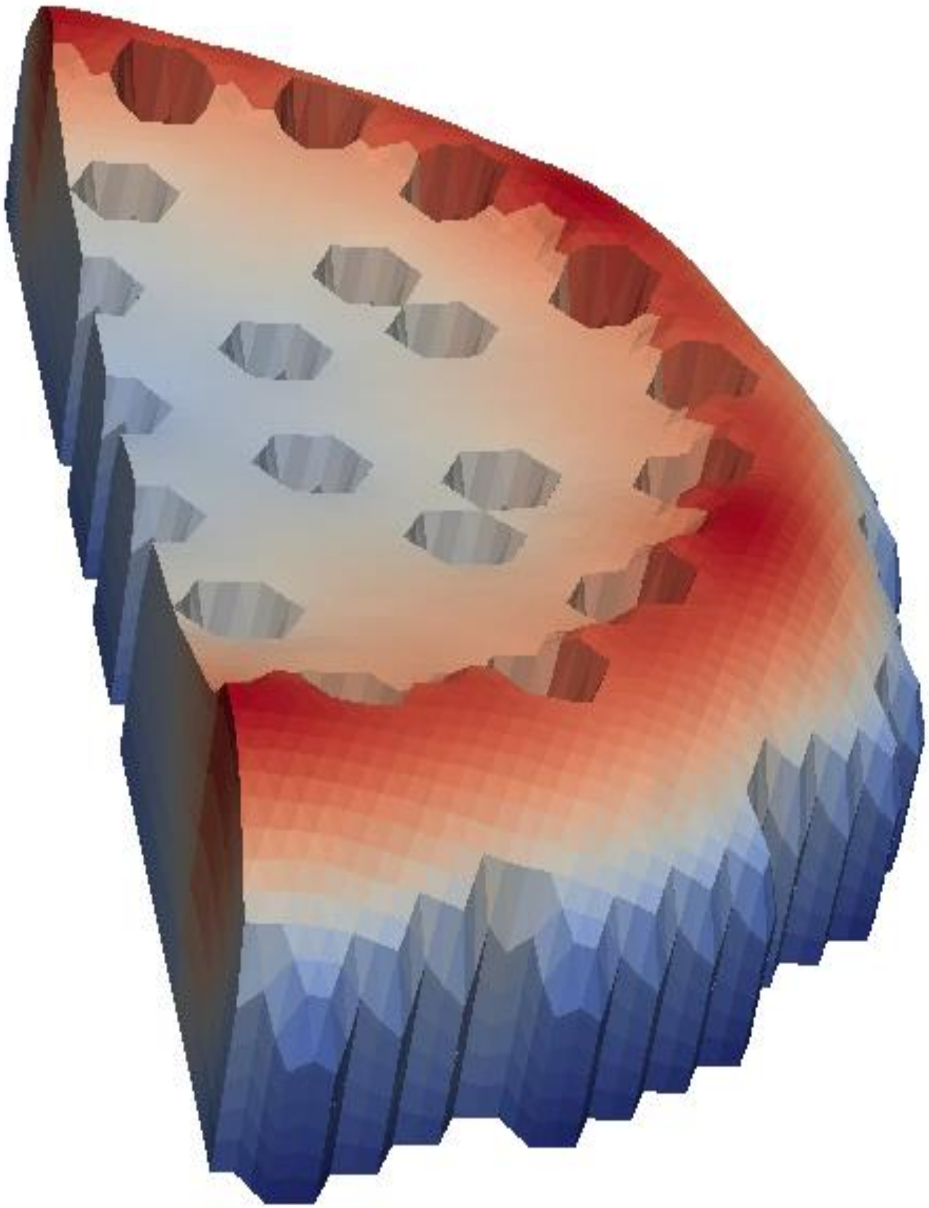
GeN-Foam: an OpenFOAM based multi-physics solver for nuclear reactor analysis

Carlo Fiorina – carlo.fiorina@epfl.ch

OFW11 / 2016

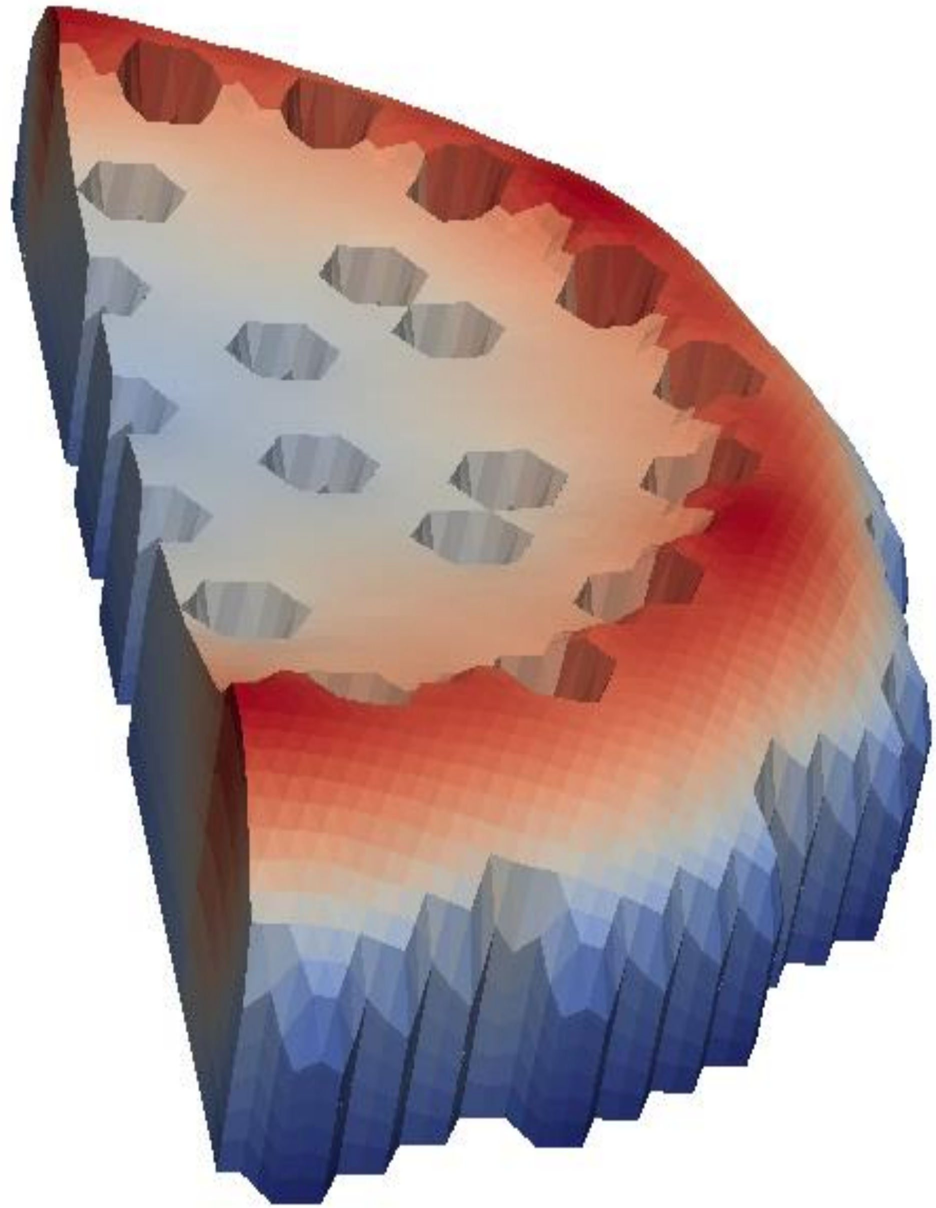
Outline

- Background
- The GeN-Foam solver
- Ongoing developments



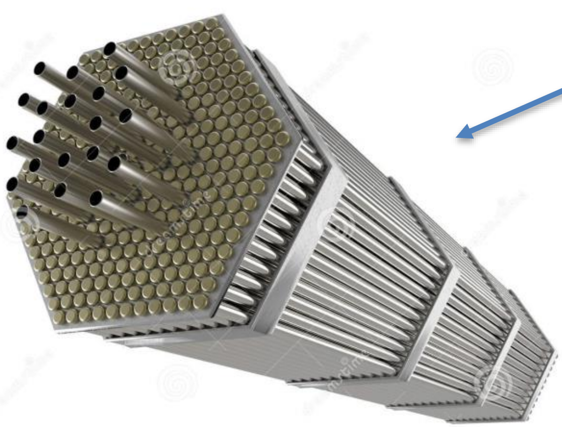
Outline

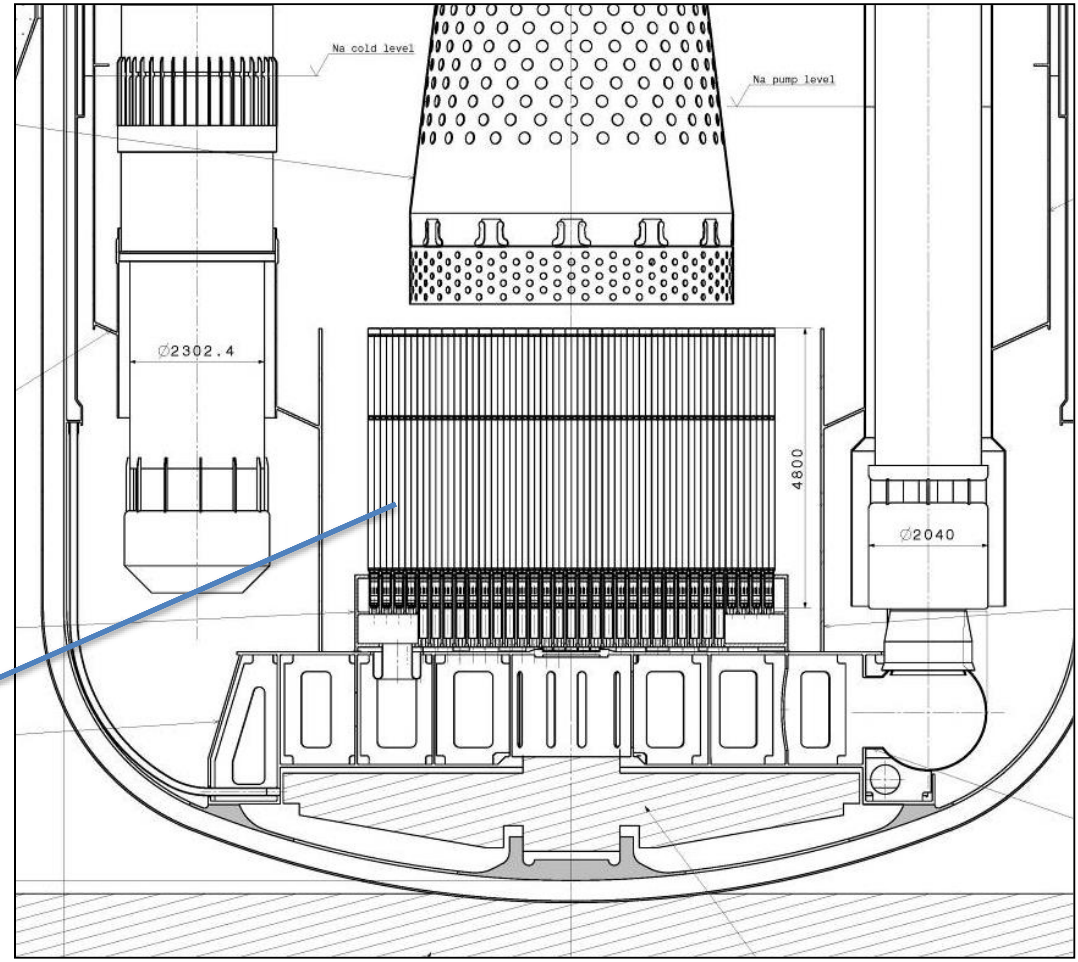
- Background
- The GeN-Foam solver
- Ongoing developments



Background

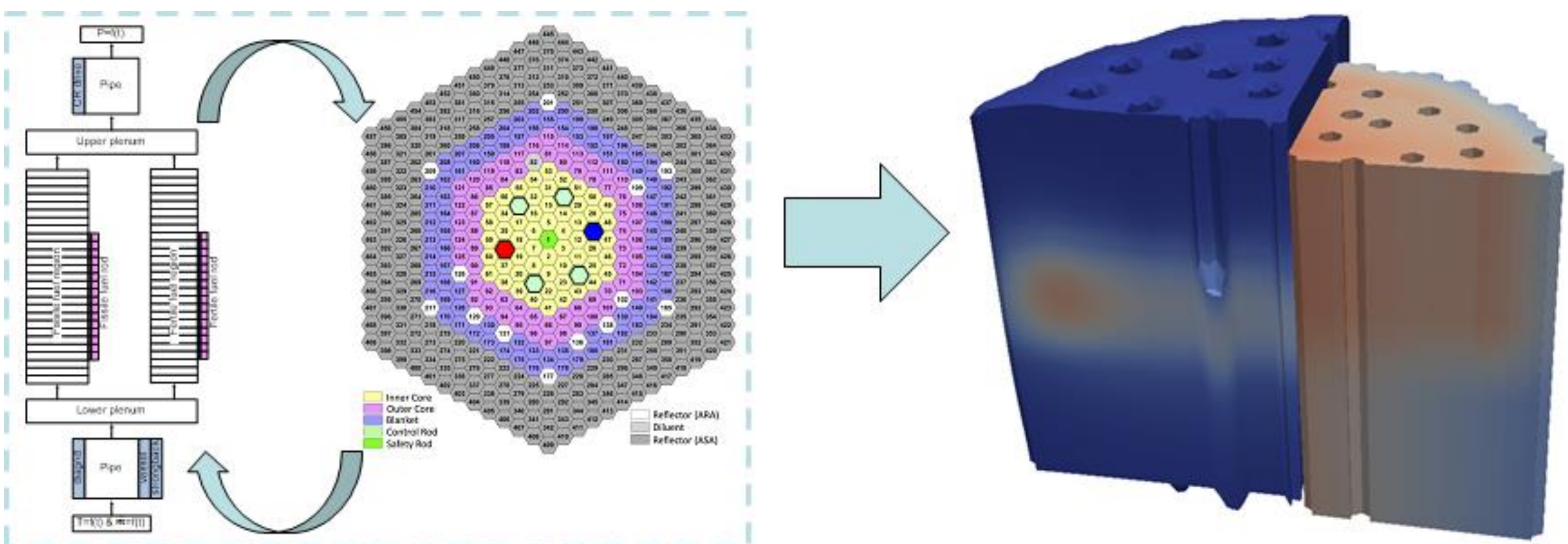
- **Nuclear reactor simulation:**
  - ✓ Fluid dynamics (complex core plus pool or loop with HXs)
  - ✓ Heat transfer
  - ✓ Neutron transport (fission of heavy nuclides, transport of neutrons, and interaction with matter)
  - ✓ Thermal expansions (influence chain reaction)
  - ✓ Mass transfer (neutron poisons, neutron emitters)





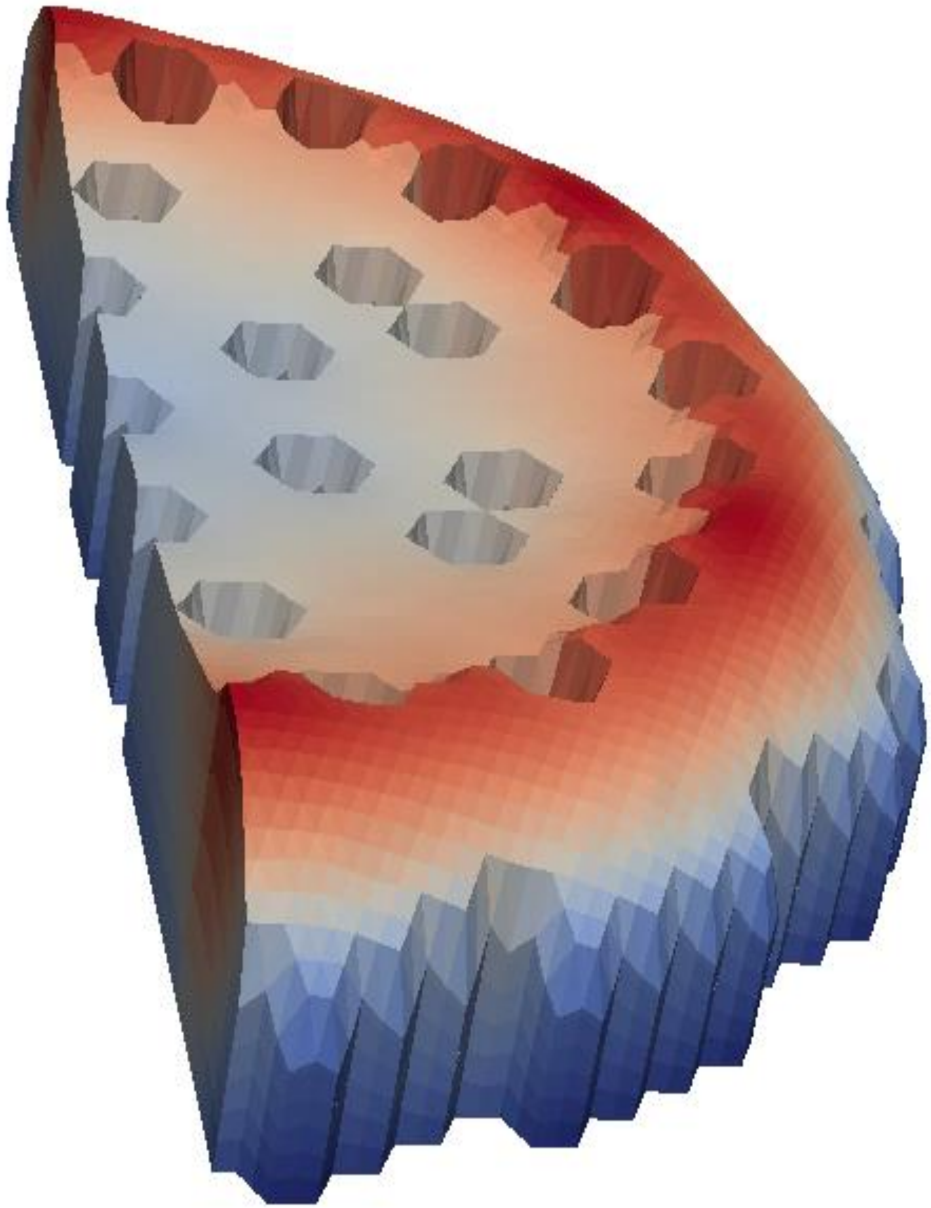
Background

- **Objective: steady state temperature and flow distribution + response to accidents**
- **Legacy 1D thermal-hydraulic codes + neutronic codes with structured mesh** coupled externally
- **Objective at EPFL**: development and validation of a **modern multi-physics tool** featuring:
  - ✓ **tight, modern coupling** (no data-transfer interface between codes)
  - ✓ **high performance** (parallel computing)
  - ✓ **flexibility** (easy modification, implementation of new features, unstructured meshes)
- What for: **state-of-the-art <u>multi-physics</u> tool** to supplement “legacy” codes



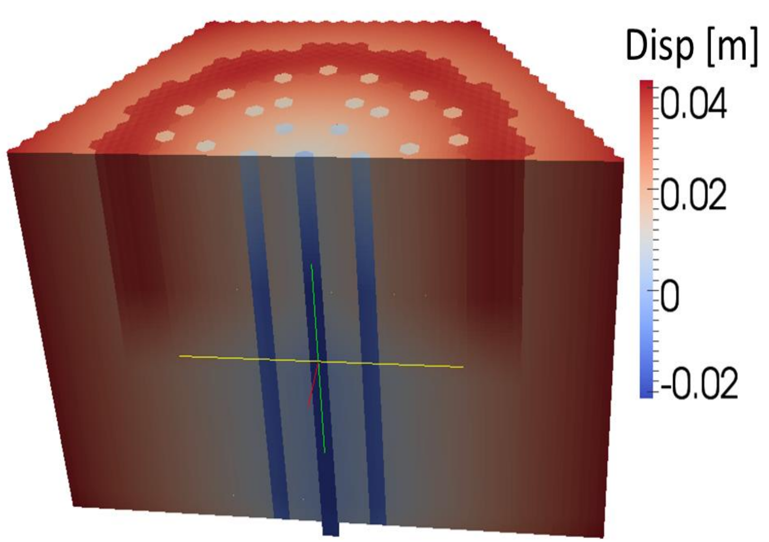
Outline

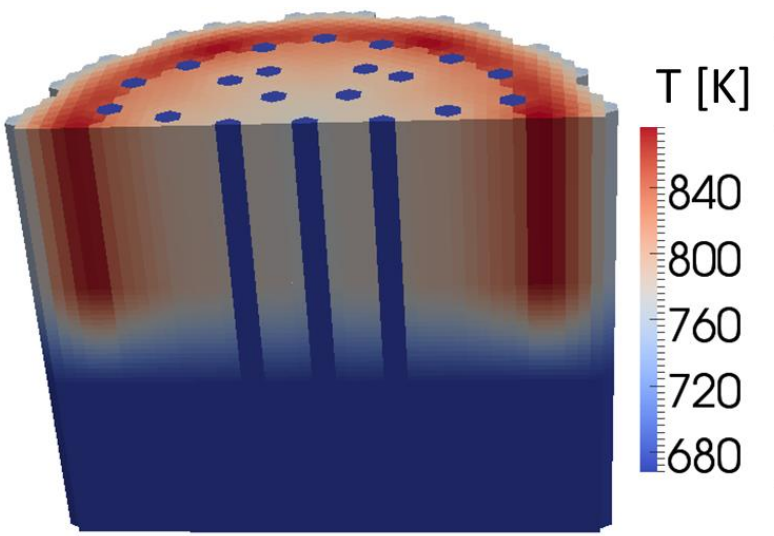
- Background
- The GeN-Foam solver
- Ongoing developments

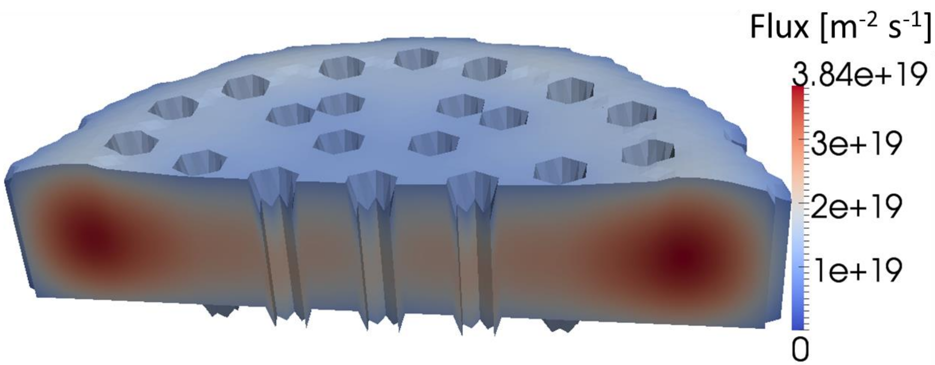


GeN-Foam – Generalized Nuclear Foam

- 2D or 3D **transient analysis of full core and/or primary loop**
- Solves for:
  - ✓ **Coarse/fine mesh thermal hydraulics**
  - ✓ Subscale **fuel temperature** field (coarse mesh)
  - ✓ **Neutronics (neutron transport)**
  - ✓ **Thermal mechanics**
- **Implicitly coupled (semi)**
- Three independent **unstructured meshes**
- **Adaptive** time step (cfl + max power variation)







Coarse/fine mesh thermo-hydraulics

- **Coarse mesh with porous medium equations** for selected mesh zones (no Darcy, but full NS + source terms):

$$\frac{\partial \gamma\rho}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}) = 0$$

$$\frac{\partial \gamma\rho\boldsymbol{u}}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}\otimes\boldsymbol{u}) = \nabla\cdot(\mu_T\nabla\boldsymbol{u}) - \nabla\gamma p + p_i\nabla\gamma + \gamma\boldsymbol{F_g} + \gamma\boldsymbol{F_{ss}}$$

$$\boldsymbol{F_{ss}} = \boldsymbol{\kappa}(\boldsymbol{u_D})\cdot\boldsymbol{u_D}$$

$$\kappa(u_D)_{ii} = \frac{f_{D,i}\rho u_{D,i}}{2D_h\gamma^2}$$

$$f_{D,i} = A_{f_{D,i}} Re^{B_{f_{D,i}}}$$

$$\frac{\partial \gamma\rho e}{\partial t} + \nabla\cdot\left(\boldsymbol{u}\gamma(\rho e + p)\right) = \nabla\cdot(\gamma k_T\nabla \mathrm{T}) + \gamma\boldsymbol{F_{ss}}\cdot\boldsymbol{u} + \gamma\dot{Q}_{ss}$$

$$\dot{Q}_{ss} = A_V h(T_{SS} - T)$$

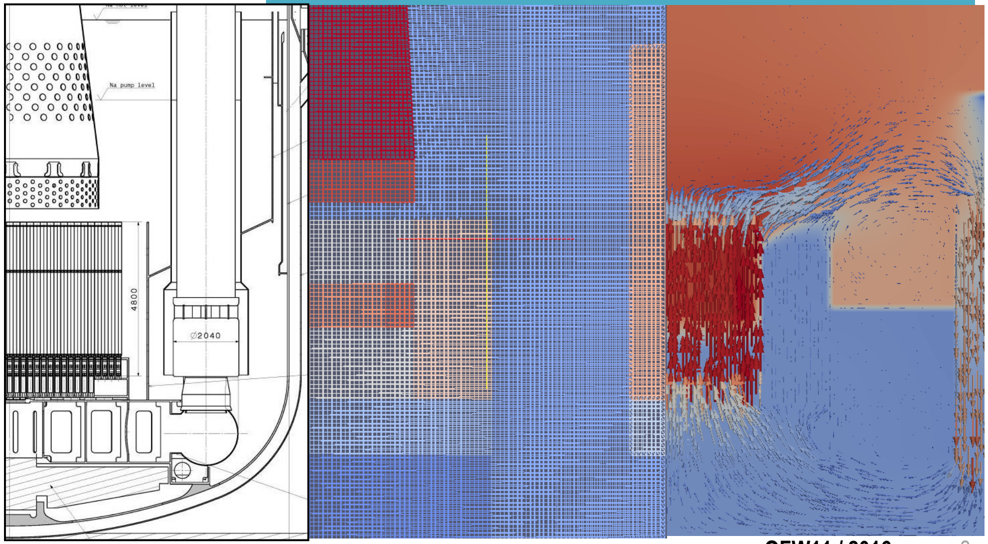
$$Nu_i = A_{Nu,i} Re^{B_{Nu,i}} Pr^{C_{Nu,i}} + D_{Nu,i}$$

$$\rho_{SS} c_{p,ss}\frac{\partial T_{SS}}{\partial t} = \nabla\cdot(\gamma\boldsymbol{k_{SS}}\nabla T) + A_V h(T - T_{SS})$$

- Source term to replicate standard correlations for pressure drops and heat transfer
- Equivalent to 1D in main flow direction, but possibility to investigate some 3D effects
- One set of equations

```
fvm::ddt(rho, U)
+ (1/porosity)*fvm::div(phi, U)
+ turb.divDevRhoReff(U)
- porousMedium.momentumSource(U)

fvm::ddt(rho*porosity, he) +
fvm::div(phi, he) +
fvc::ddt(rho*porosity, K) +
fvc::div(phi, K) +
fvc::div(fvc::absolute(phi/fvc::interpolate(rho), U),p,"div(phiv,p)")-
fvm::laplacian(turb.alphaEff(), he)
==
porousMedium.externalVolHeatSource()-
porousMedium.heatTransferCoefficient()*
porousMedium.volumetricArea()
*(thermo.T()-Tstructures) +
subscaleFuel.heatSources(…)
```



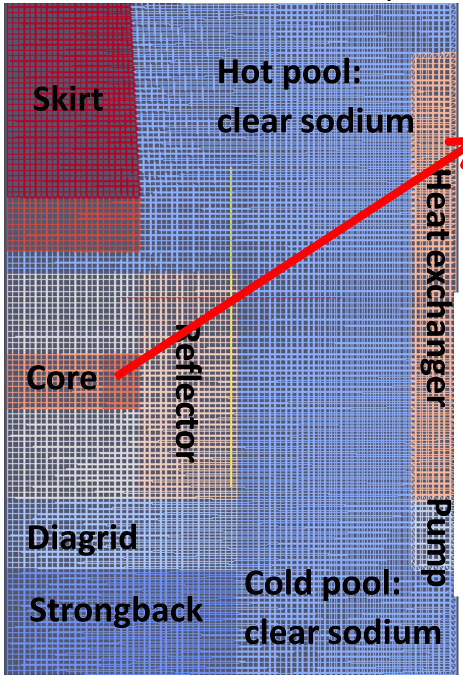
Fuel model

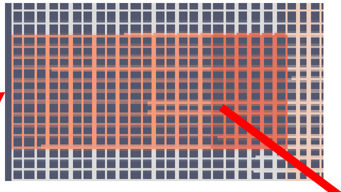
- Used for **coarse-mesh treatment of core regions**
- It evaluates **temperature profiles in cladding and fuel**, for each cell in the fuel region
- It returns the **heat source for the thermal-hydraulic solver**
- **Finite-difference second-order implicit** solution for temperatures in subscale fuel and cladding. **1-D problem** (no axial conduction)

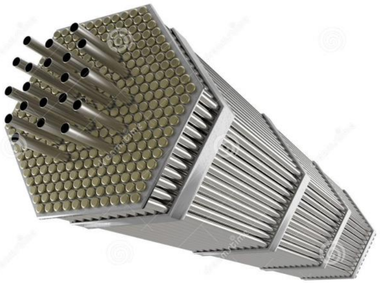
```
fvm::ddt(rho*porosity, he) +
…
==
… +
subscaleFuel.heatSources(…)
```

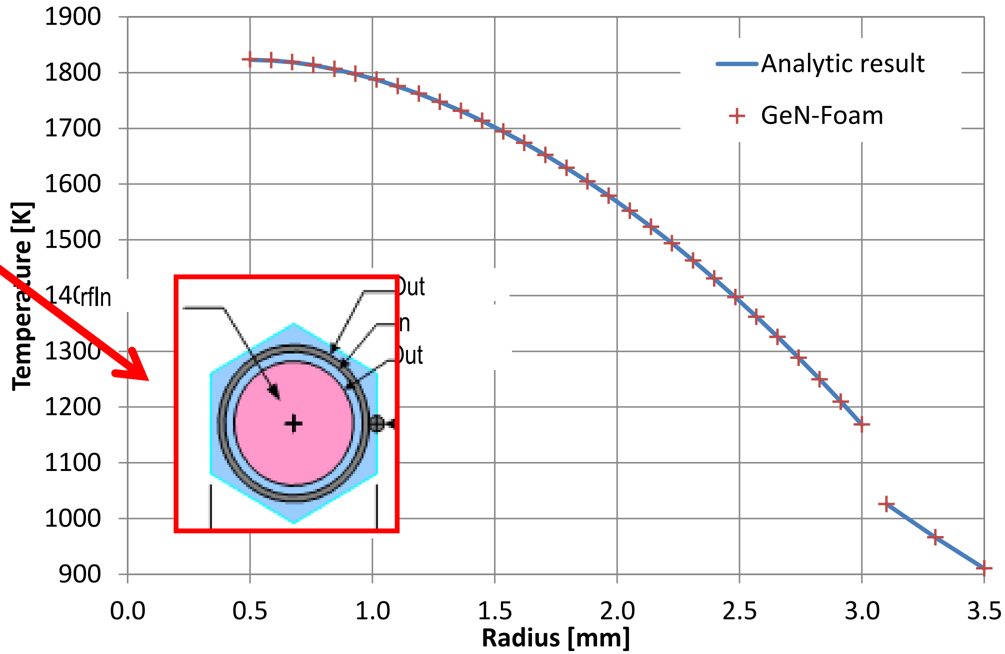
$$\rho_f c_{p,f} \frac{\partial T_f}{\partial t} = k_f \frac{\partial^2 T_f}{\partial r^2} + k_f \frac{1}{r}\frac{\partial T_f}{\partial r} + \dot{Q}_f$$

$$\rho_c c_{p,c} \frac{\partial T_c}{\partial t} = k_c \frac{\partial^2 T_c}{\partial r^2} + k_c \frac{1}{r}\frac{\partial T_c}{\partial r}$$







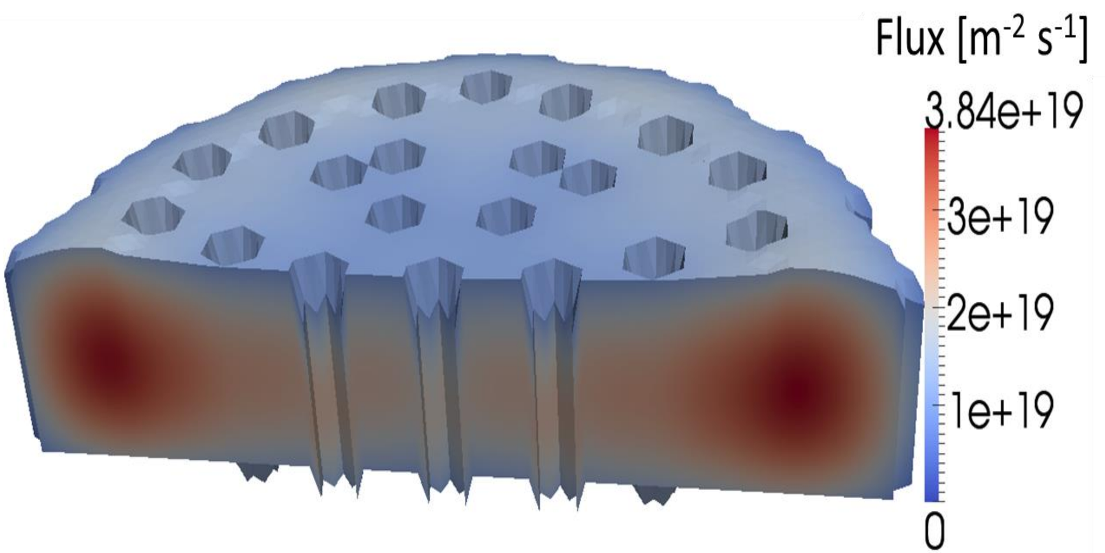


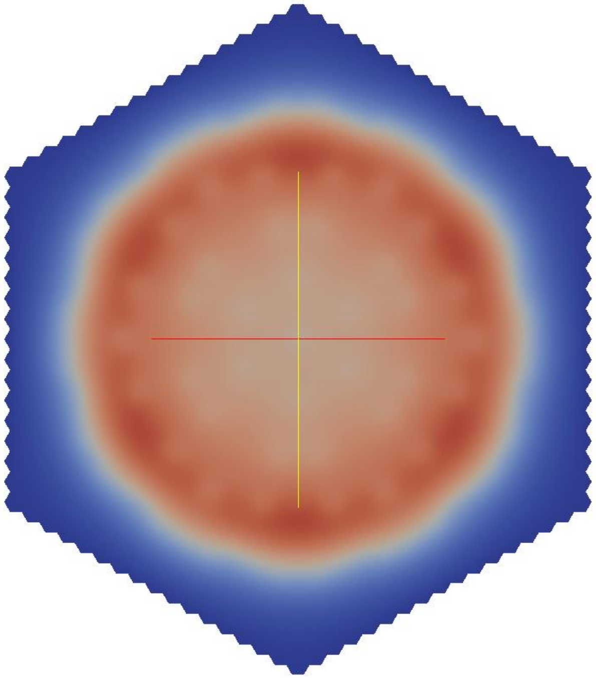
Neutronics – diffusion / SP3

- **Multi-group diffusion or SP3** with user-selected energy groups and discontinuity factors
- **Eigenvalue** (power iteration) or **time dependent**
- **Mesh deformed** according to displacement field (direct prediction of expansion-related feedbacks)

$$\frac{1}{v_i}\frac{\partial \hat{\varphi}_{0,i}}{\partial t} = \nabla \cdot D_i \nabla \hat{\varphi}_{0,i} + \frac{\nu\Sigma_{f,i}(1-\beta_t)\chi_{p,i}}{k_{eff}}\hat{\varphi}_{0,i} - \Sigma_{r,i}\hat{\varphi}_{0,i} + \frac{S_{n,i}(1-\beta_t)\chi_{p,i}}{k_{eff}} + S_d\chi_{d,i} + S_{s,i} + 2\,\Sigma_{r,i}\varphi_{2,i} + 2\frac{1}{v_i}\frac{\partial \varphi_{2,i}}{\partial t}$$

$$\frac{3}{v_i}\frac{\partial \varphi_{2,i}}{\partial t} = \frac{3}{7}\nabla \cdot \frac{1}{\Sigma_{t,i}}\nabla \varphi_{2,i} - \left(\frac{5}{3}\Sigma_t + \frac{4}{3}\Sigma_{r,i}\right)\varphi_{2,i} + \frac{2}{3}\Sigma_{r,i}\hat{\varphi}_{0,i} - \frac{2}{3}\frac{\nu\Sigma_{f,i}(1-\beta_t)\chi_{p,i}}{k_{eff}}\hat{\varphi}_{0,i} - \frac{2}{3}\frac{S_{n,i}(1-\beta_t)\chi_{p,i}}{k_{eff}} - \frac{2}{3}S_d\chi_{d,i} - \frac{2}{3}S_{s,i} + \frac{2}{3}\frac{1}{v_i}\frac{\partial \hat{\varphi}_{0,i}}{\partial t}$$



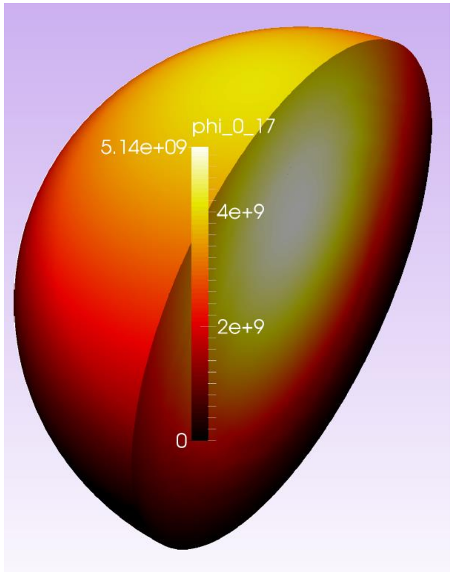


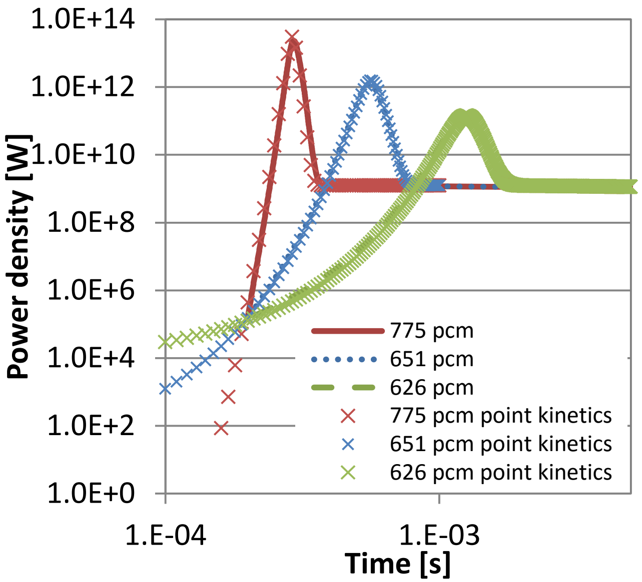
Neutronics – discrete ordinates

- Developed **for accurate neutronic calculations**
- Discrete ordinates:
  - ✓ Neutrons subdivided based on energy and direction:

$$\frac{\partial \varphi_{ei,di}}{V_{ei}\partial t} + \nabla \cdot (\Omega \cdot \varphi_{ei,di}) + \Sigma_{t,ei}\varphi_{ei,di} = S_{ei,di}$$

    with *S* left explicit and Picard iterations
  - ✓ User-selected scattering approx (P1,…,P7), number of energy groups, number of directions
  - ✓ **Eigenvalue / time dependent calculations**
- Good candidate for implementation in GeN-Foam
  - ✓ Requires optimization



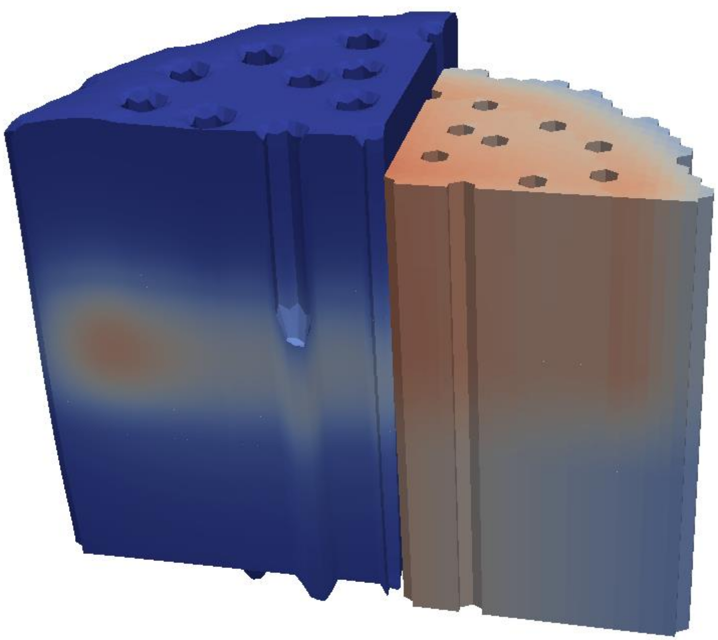


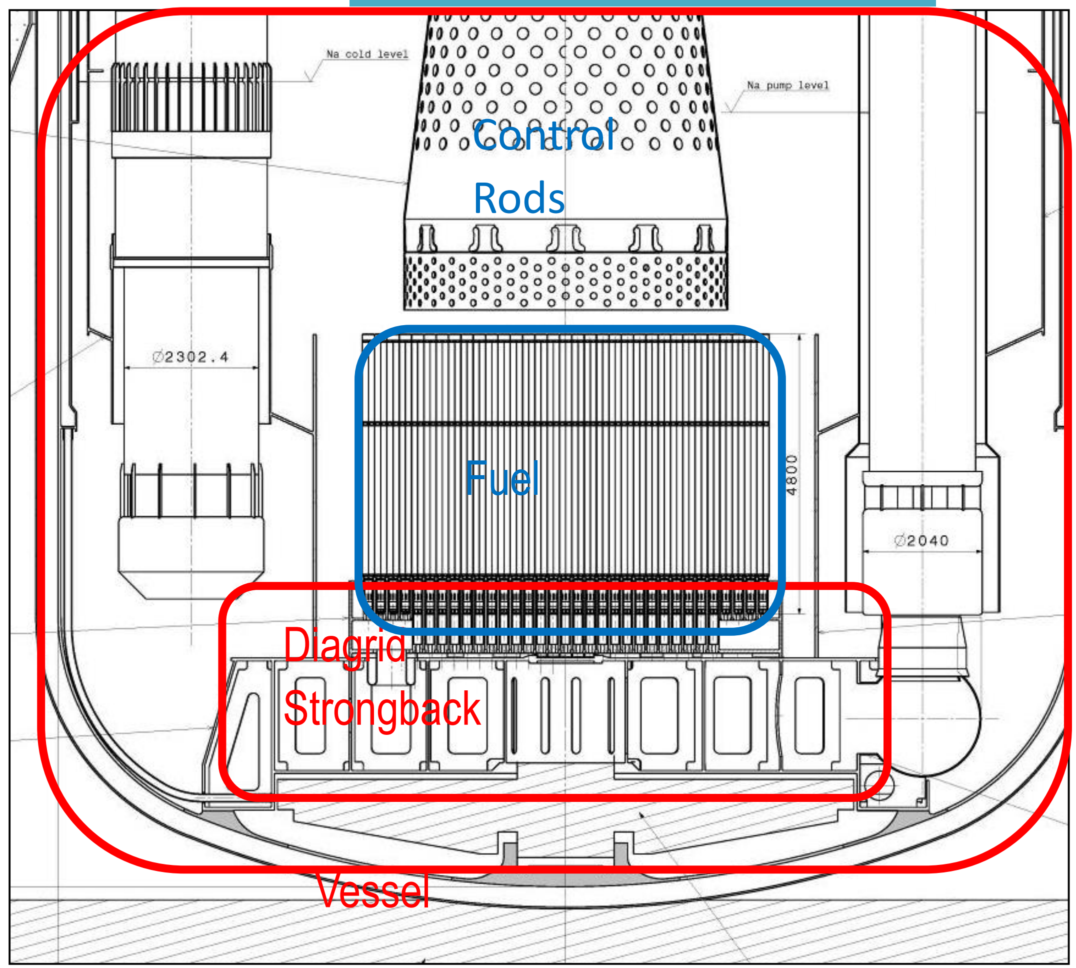
Thermal mechanics

- **To displace neutronic mesh** in fast reactor calculations
- Available displacement-based solver can be used for general structures
- Fuel and CR driveline expansion is based on expansion coefficients

$$\boldsymbol{v_f} \cdot \nabla D_f = \alpha_{f/c}\left(T_{f/c} - T_{f/c,ref}\right)$$

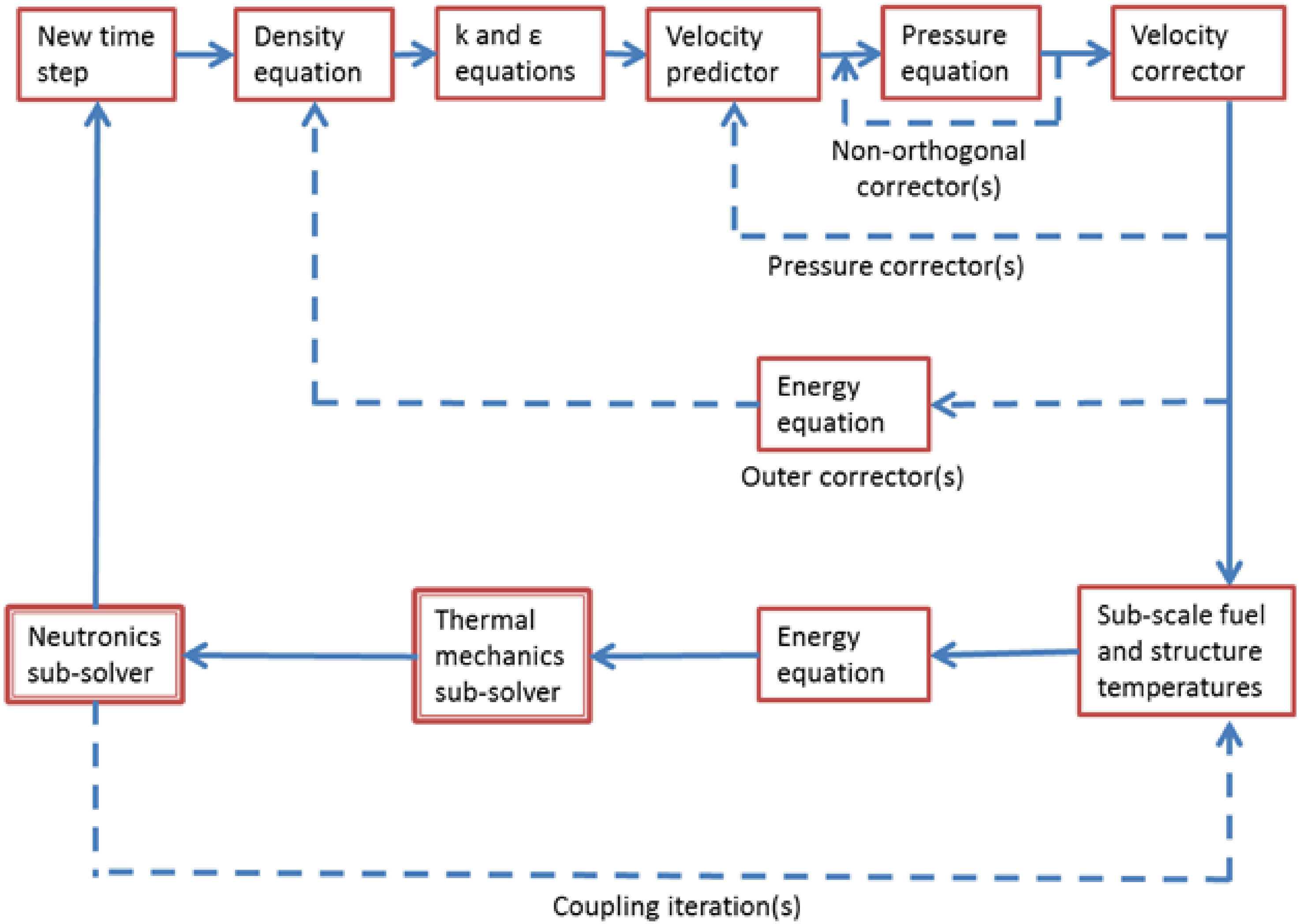
```
fvm::d2dt2(Disp) ==
fvm::laplacian(2*mu + lambda,
Disp, "laplacian(DD,D)")+
divSigmaExp
```





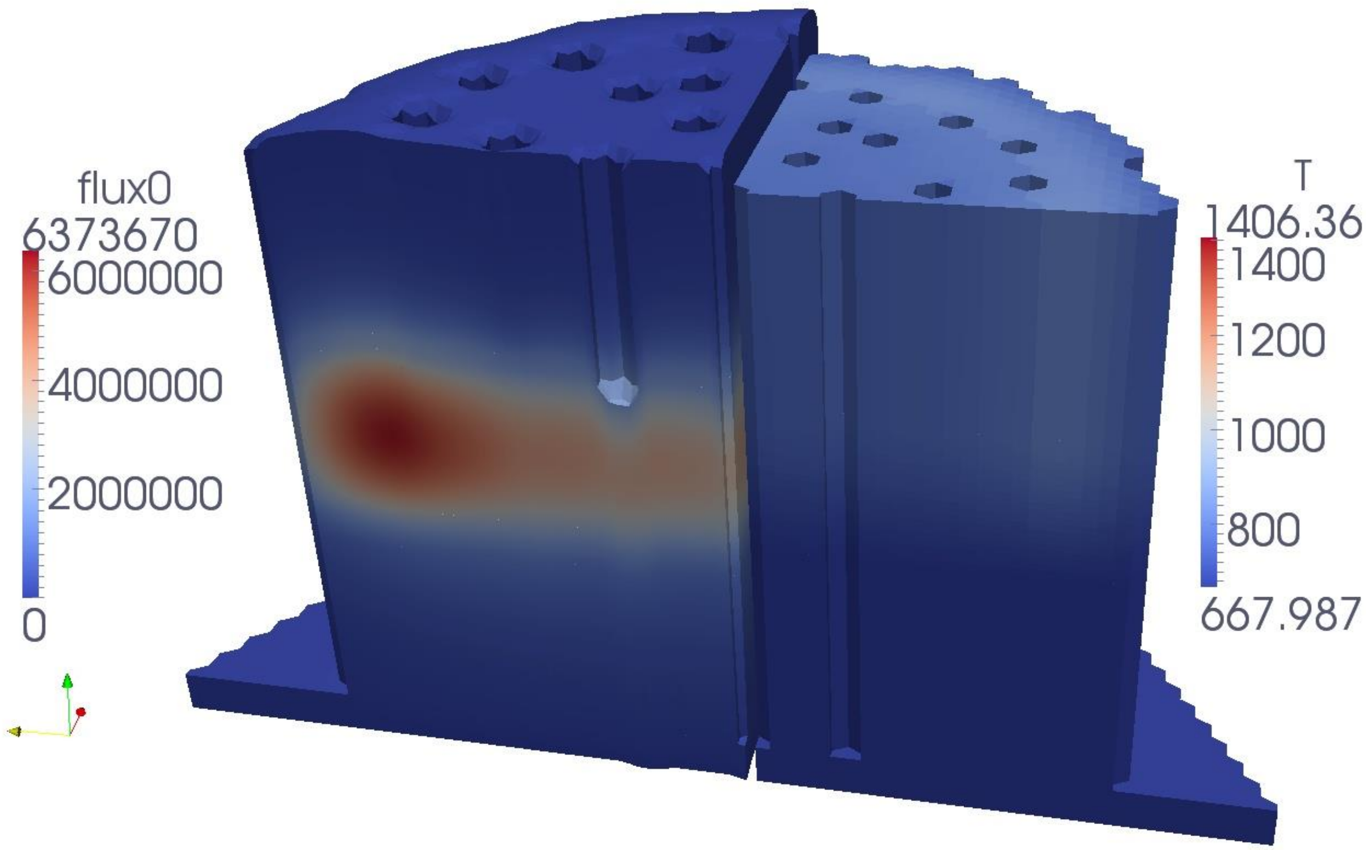
Coupling strategy

- Coupling based on PIMPLE loop + iteration on energy/thermal-mechanics/neutronics



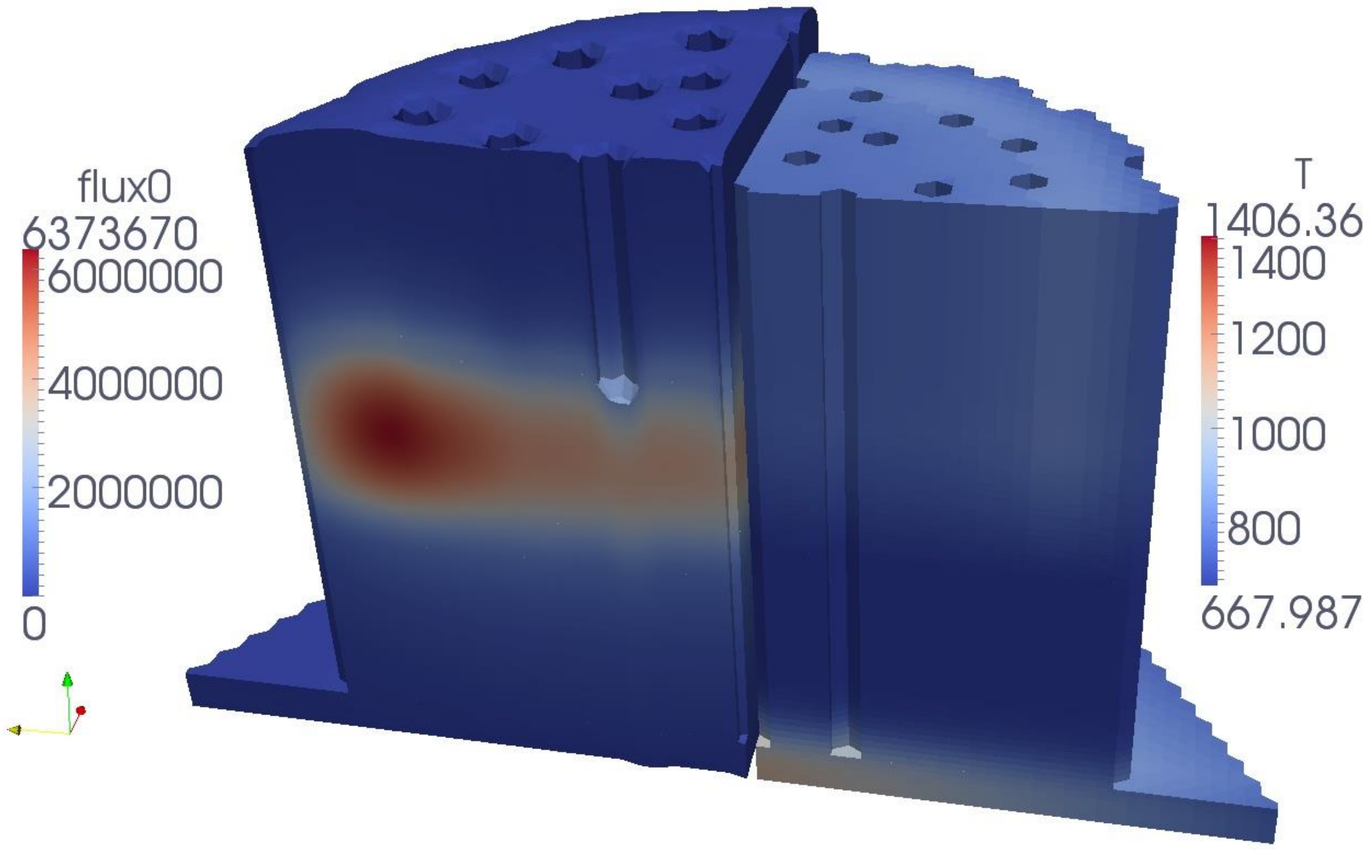
Examples – the ESFR

- Example of **a ULOF/ULOHS**



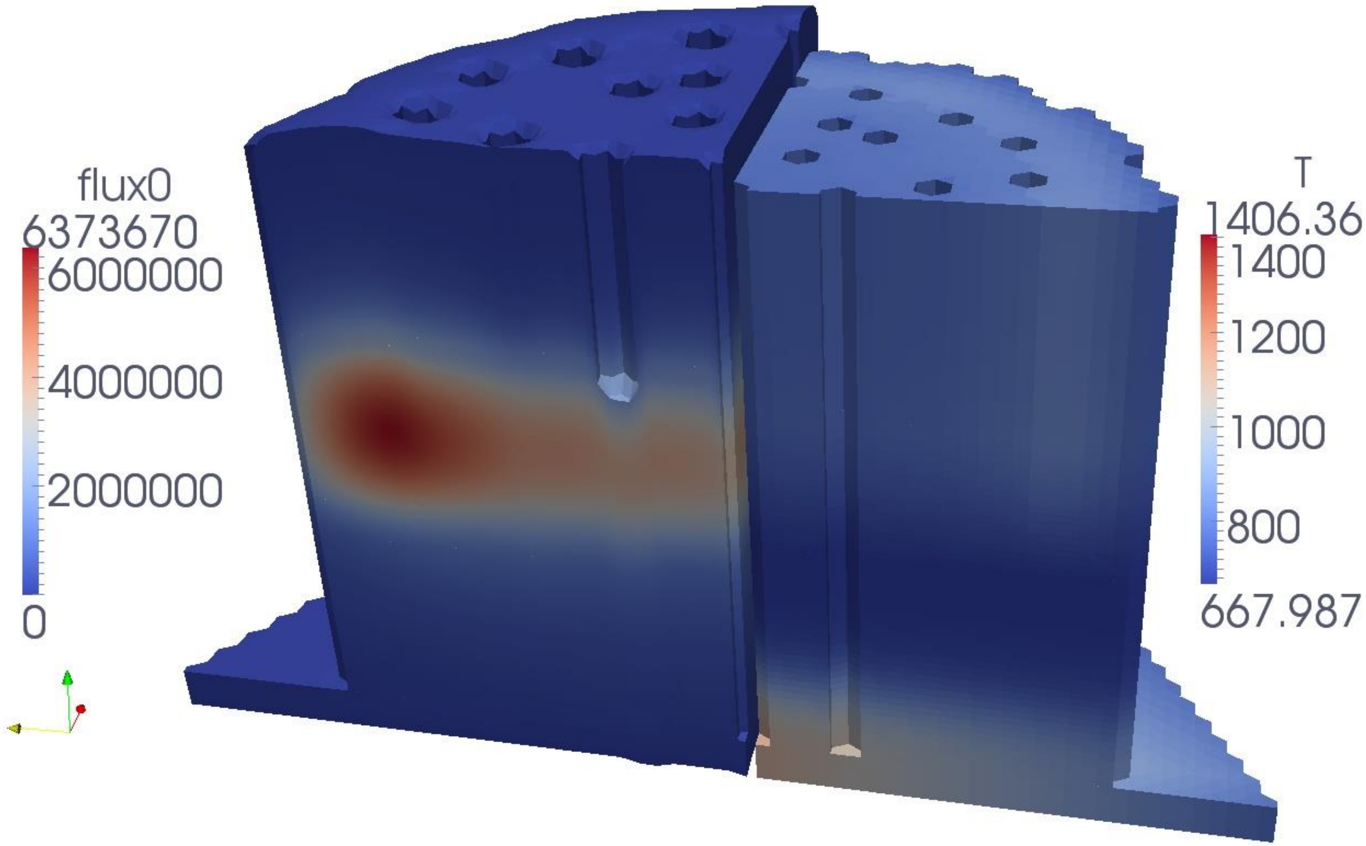
Examples – the ESFR

- Example of **a ULOF/ULOHS**



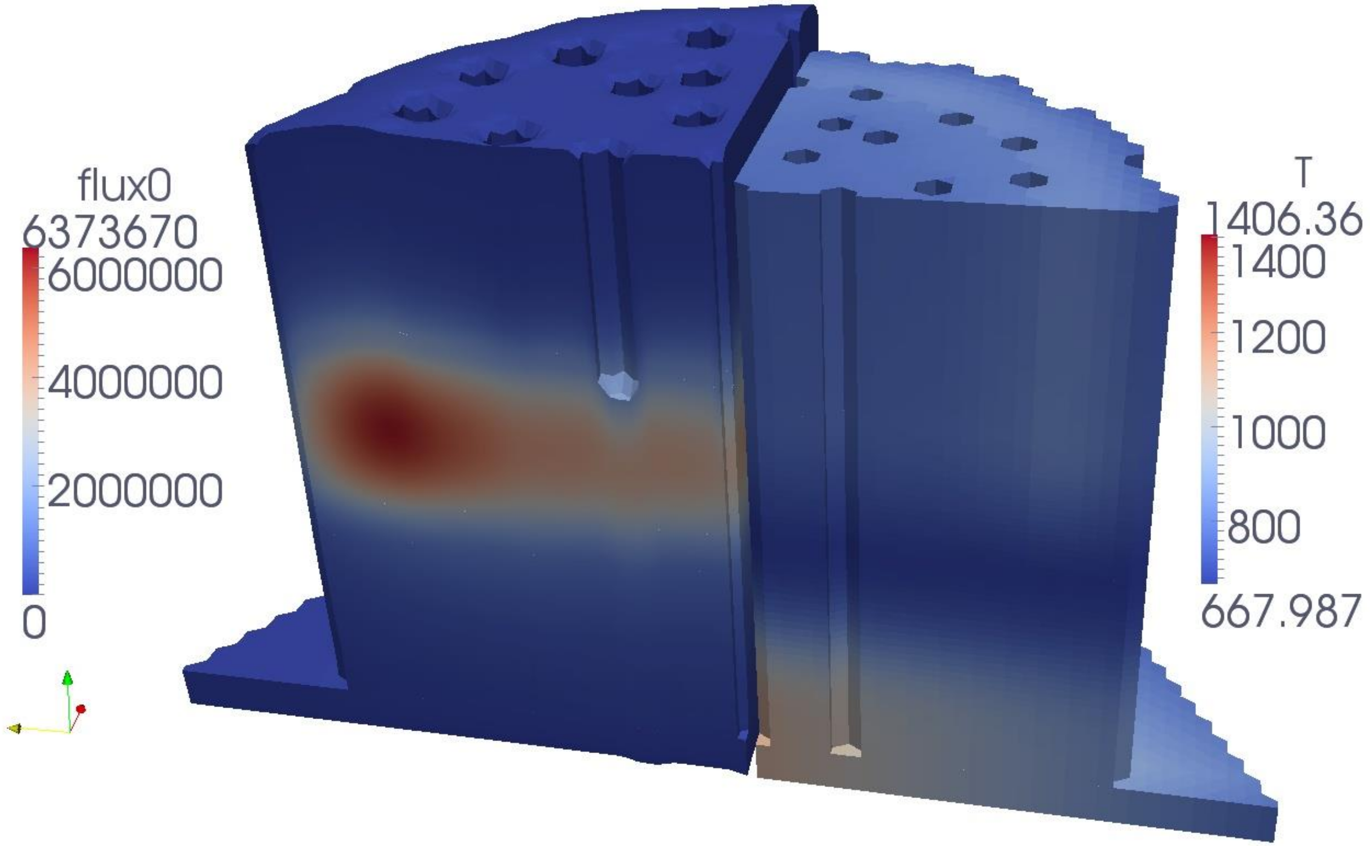
Examples – the ESFR

- Example of **a ULOF/ULOHS**



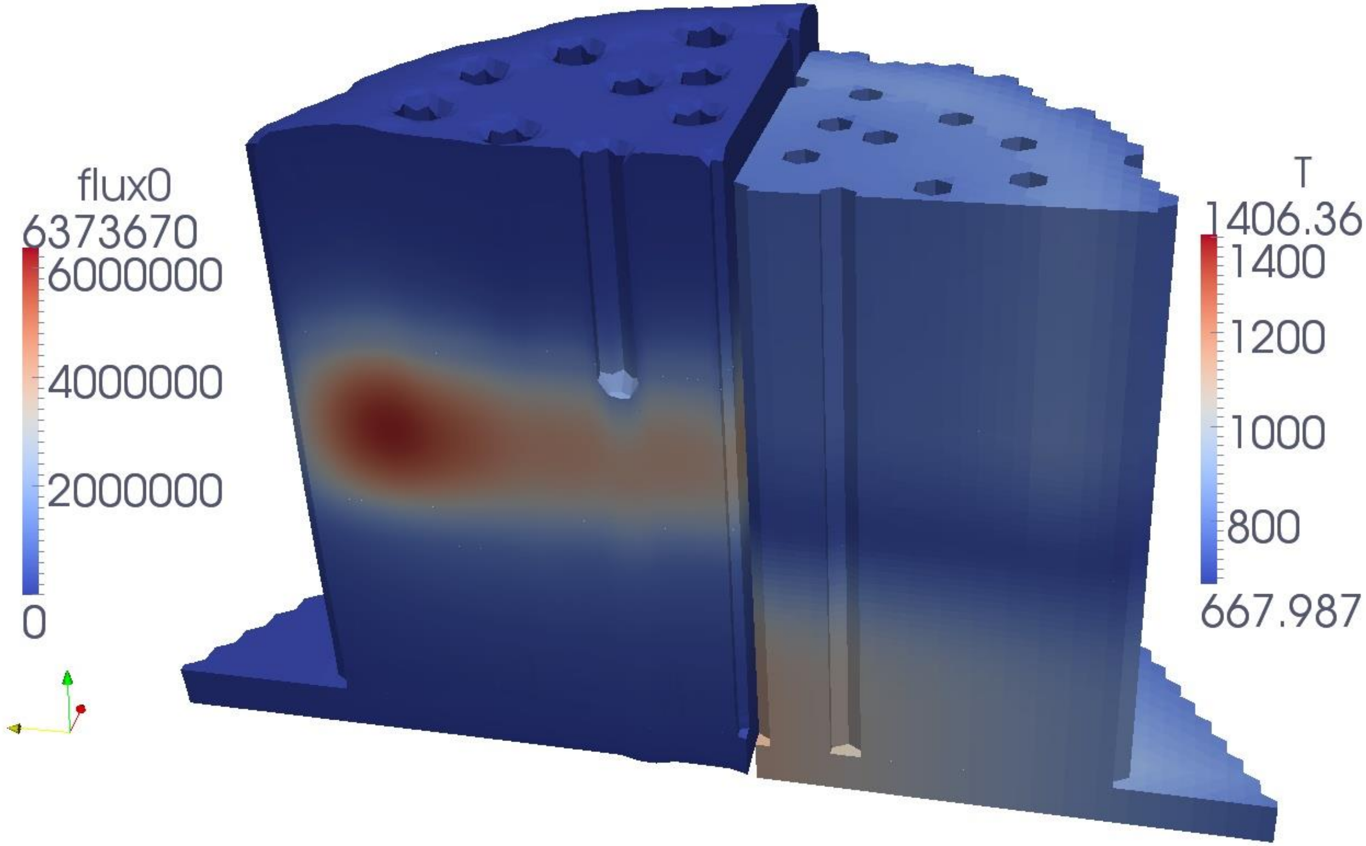
Examples – the ESFR

- Example of **a ULOF/ULOHS**



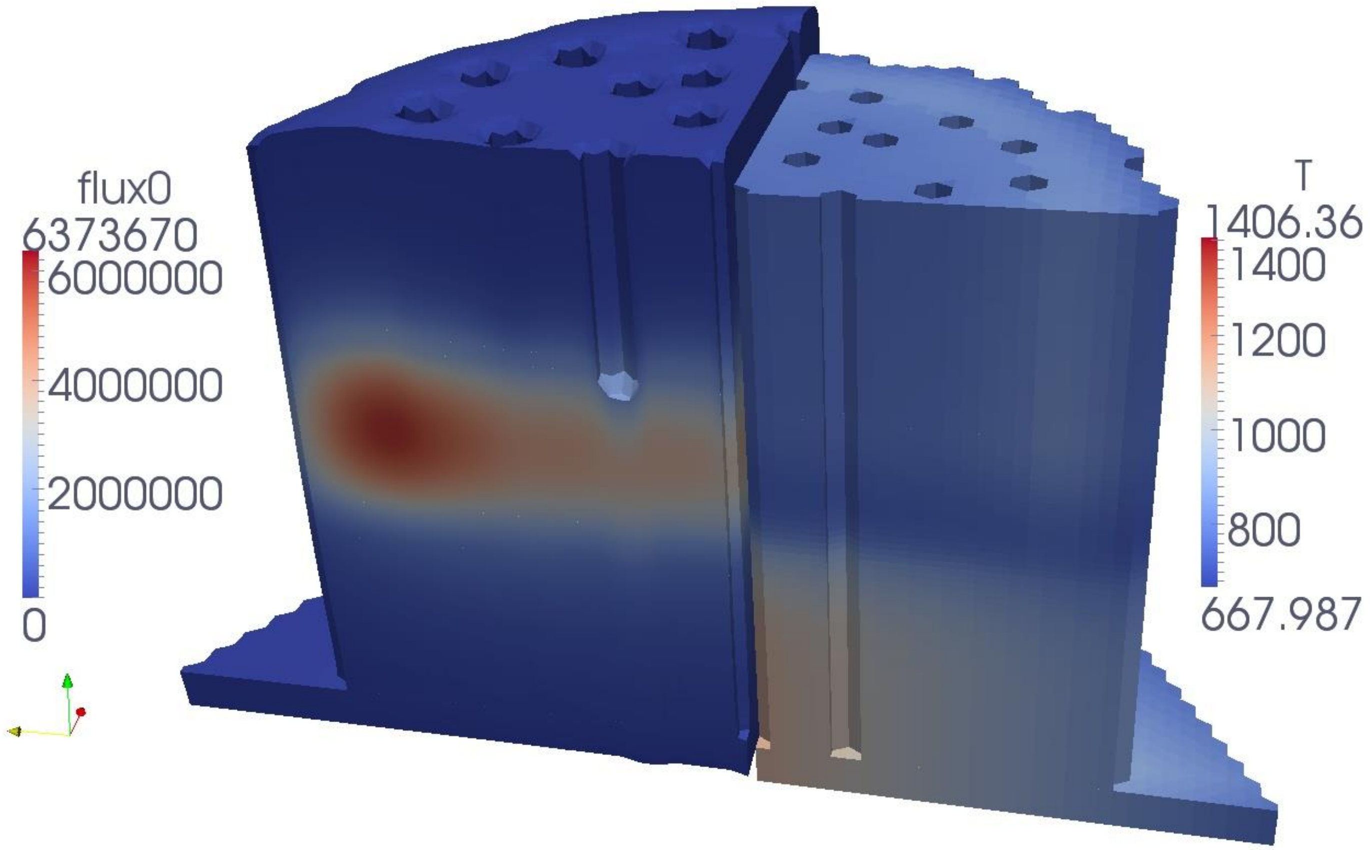
Examples – the ESFR

- Example of **a ULOF/ULOHS**



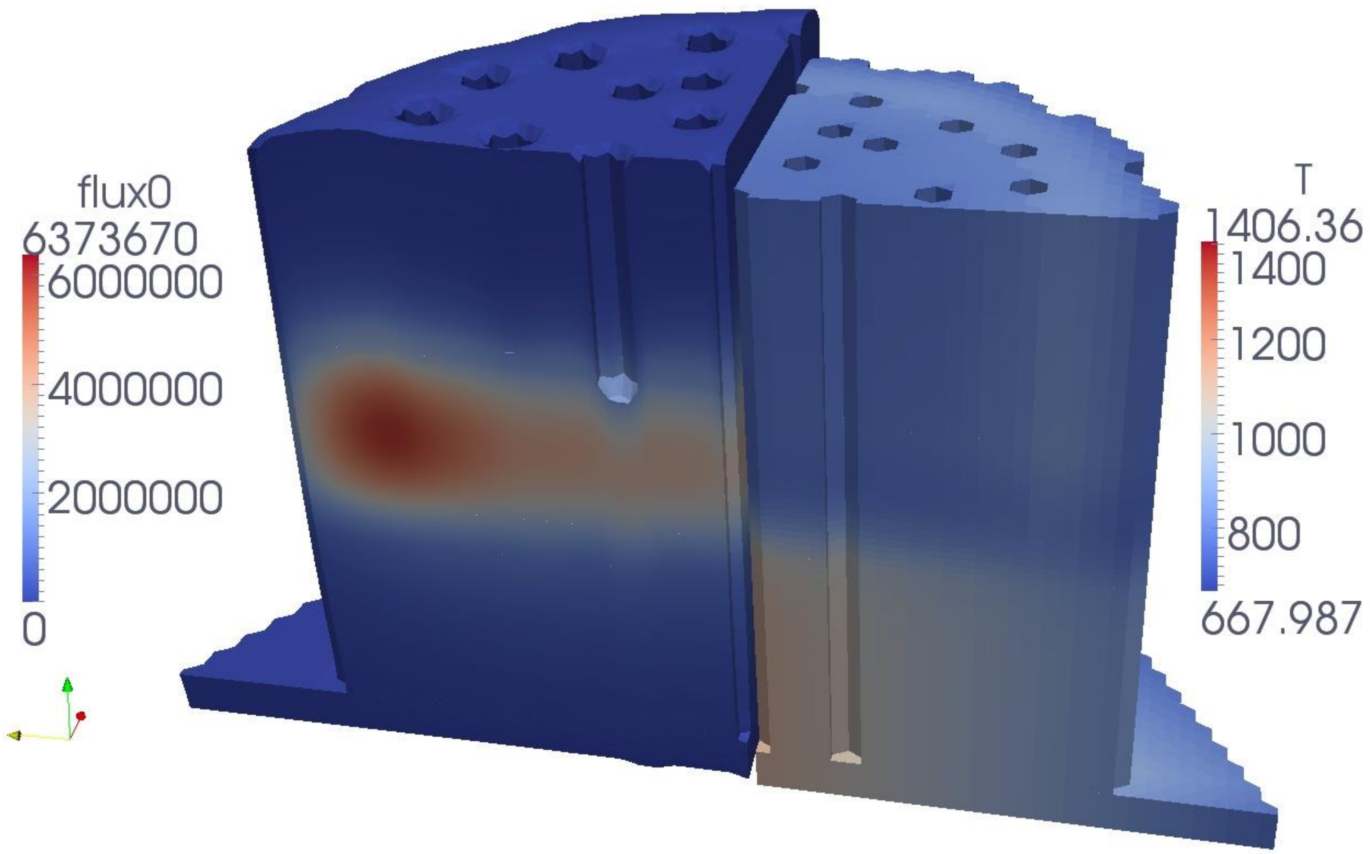
Examples – the ESFR

- Example of **a ULOF/ULOHS**



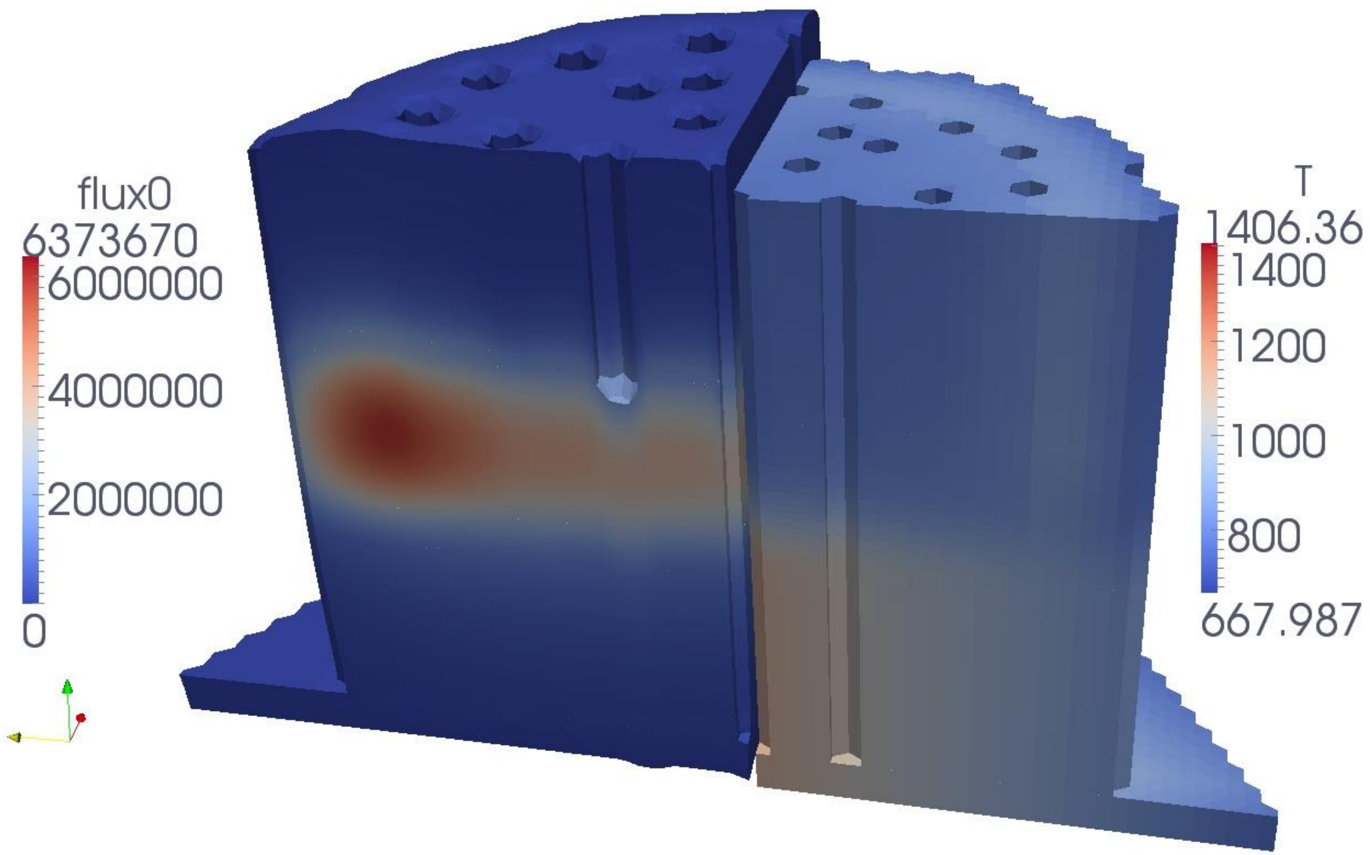
Examples – the ESFR

- Example of **a ULOF/ULOHS**



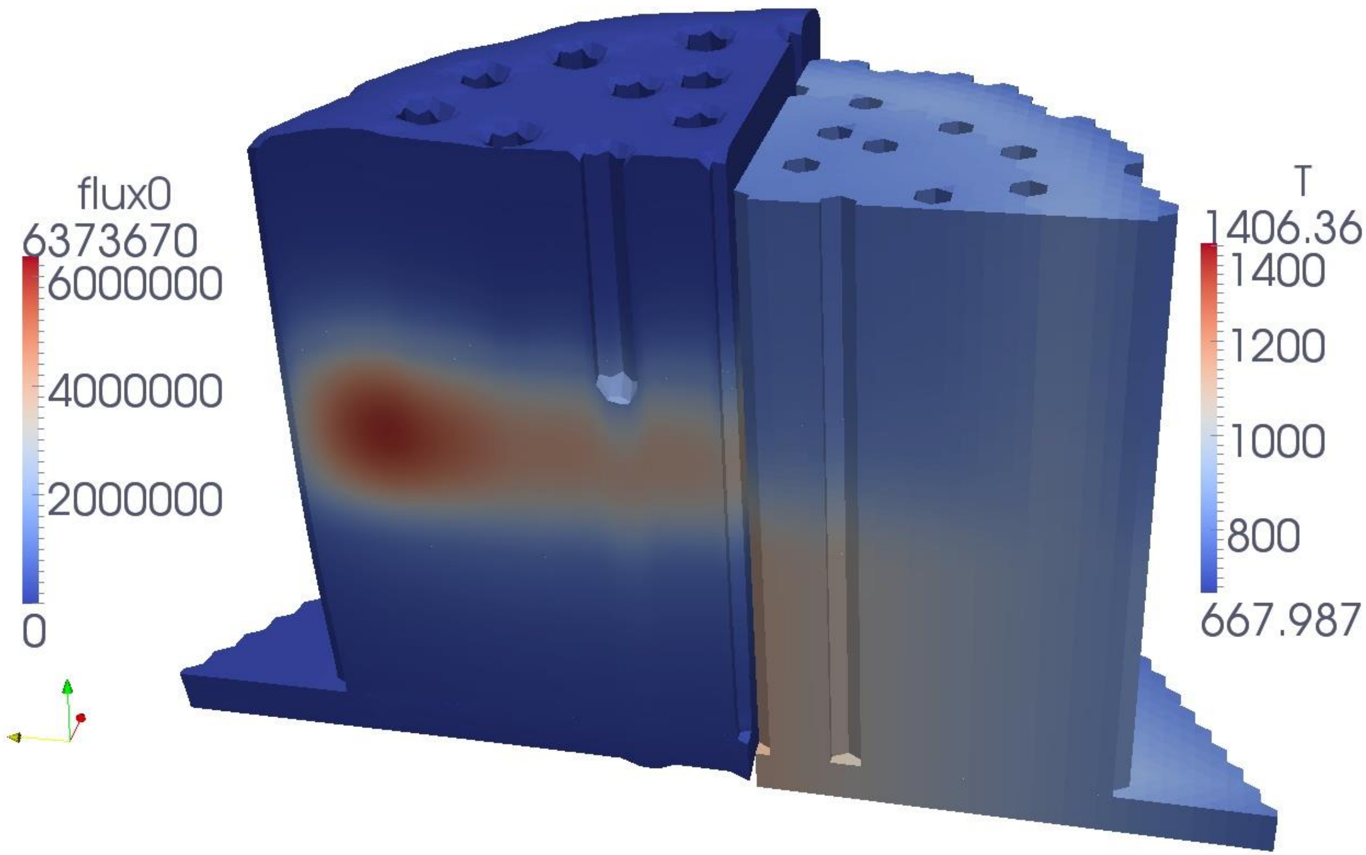
Examples – the ESFR

- Example of **a ULOF/ULOHS**



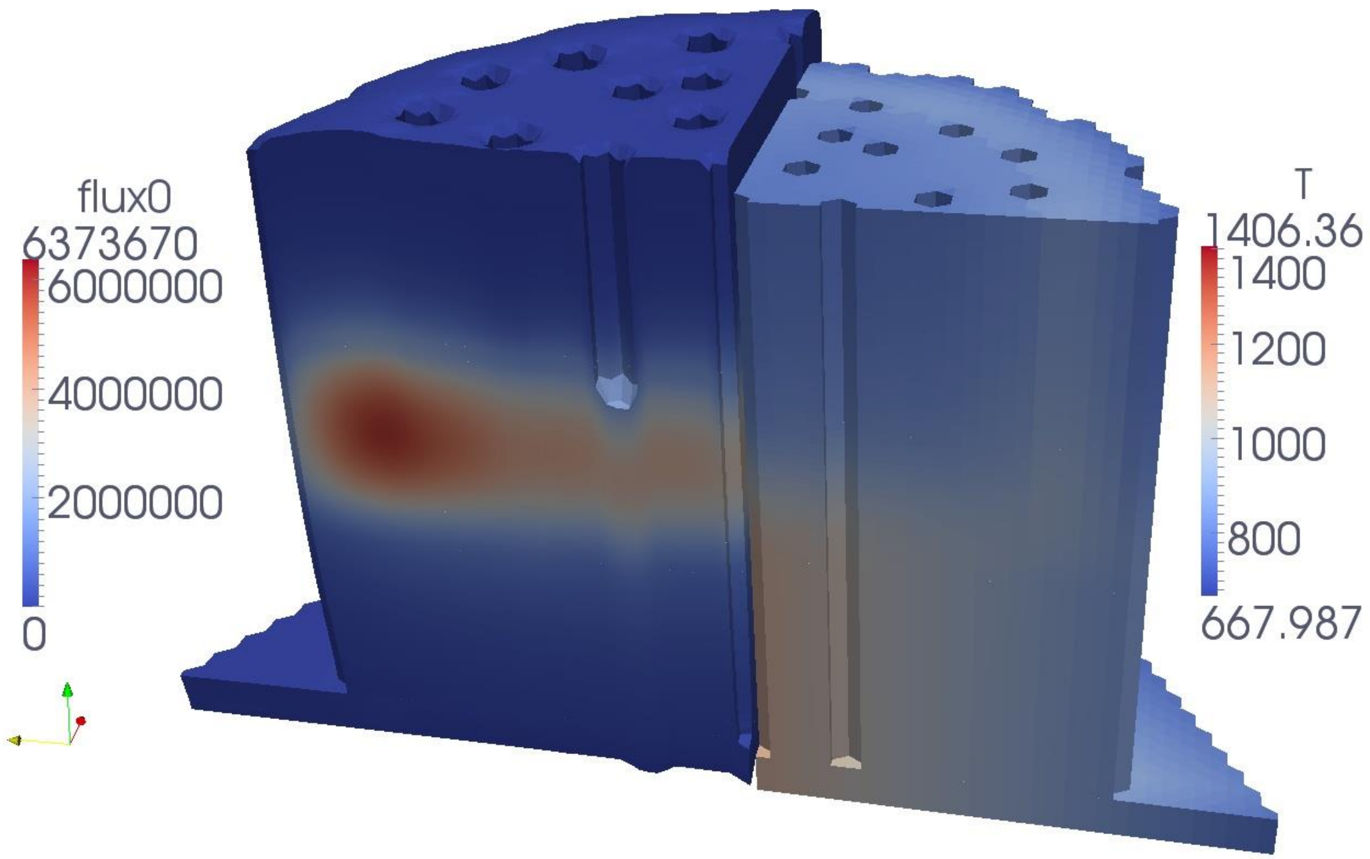
Examples – the ESFR

- Example of **a ULOF/ULOHS**



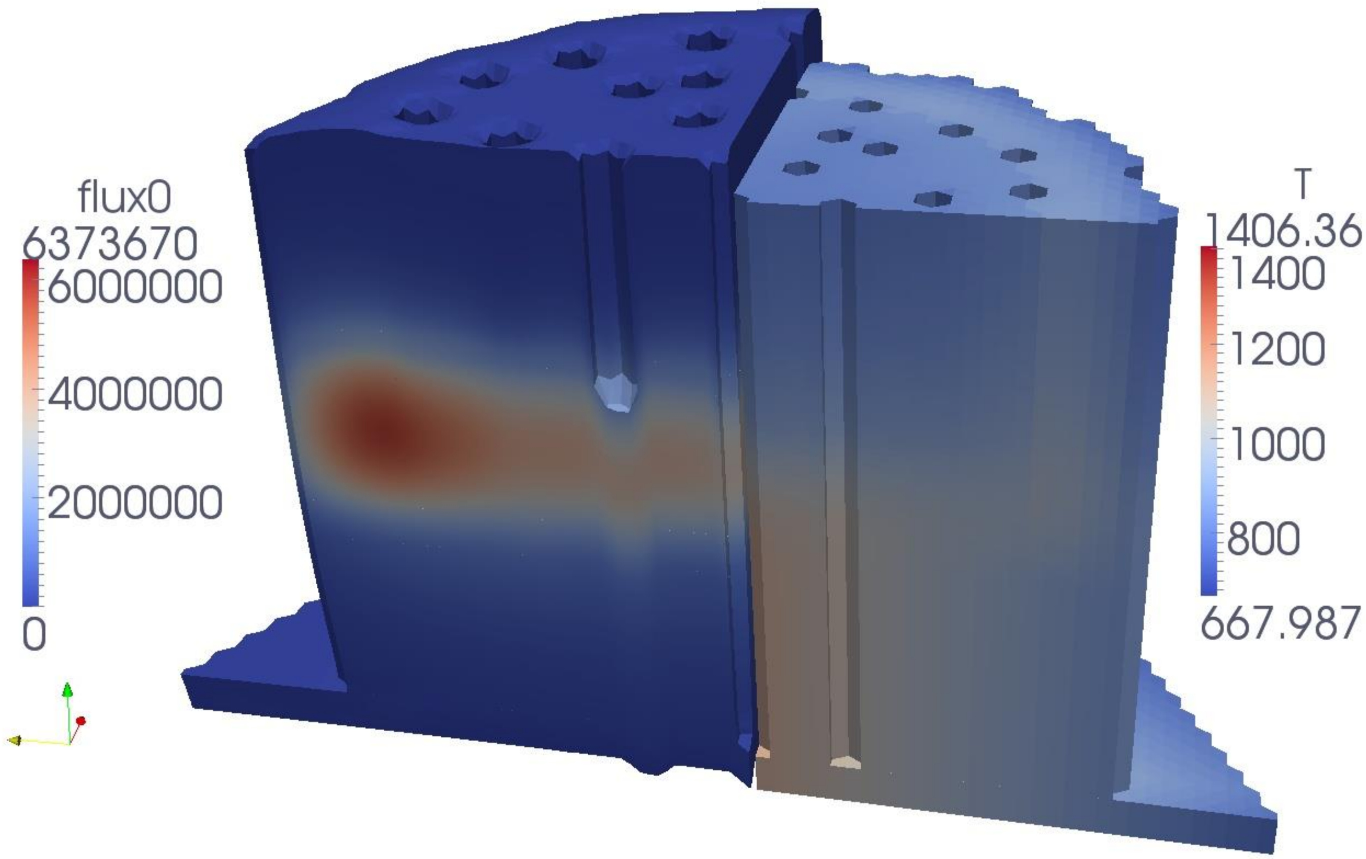
Examples – the ESFR

- Example of **a ULOF/ULOHS**



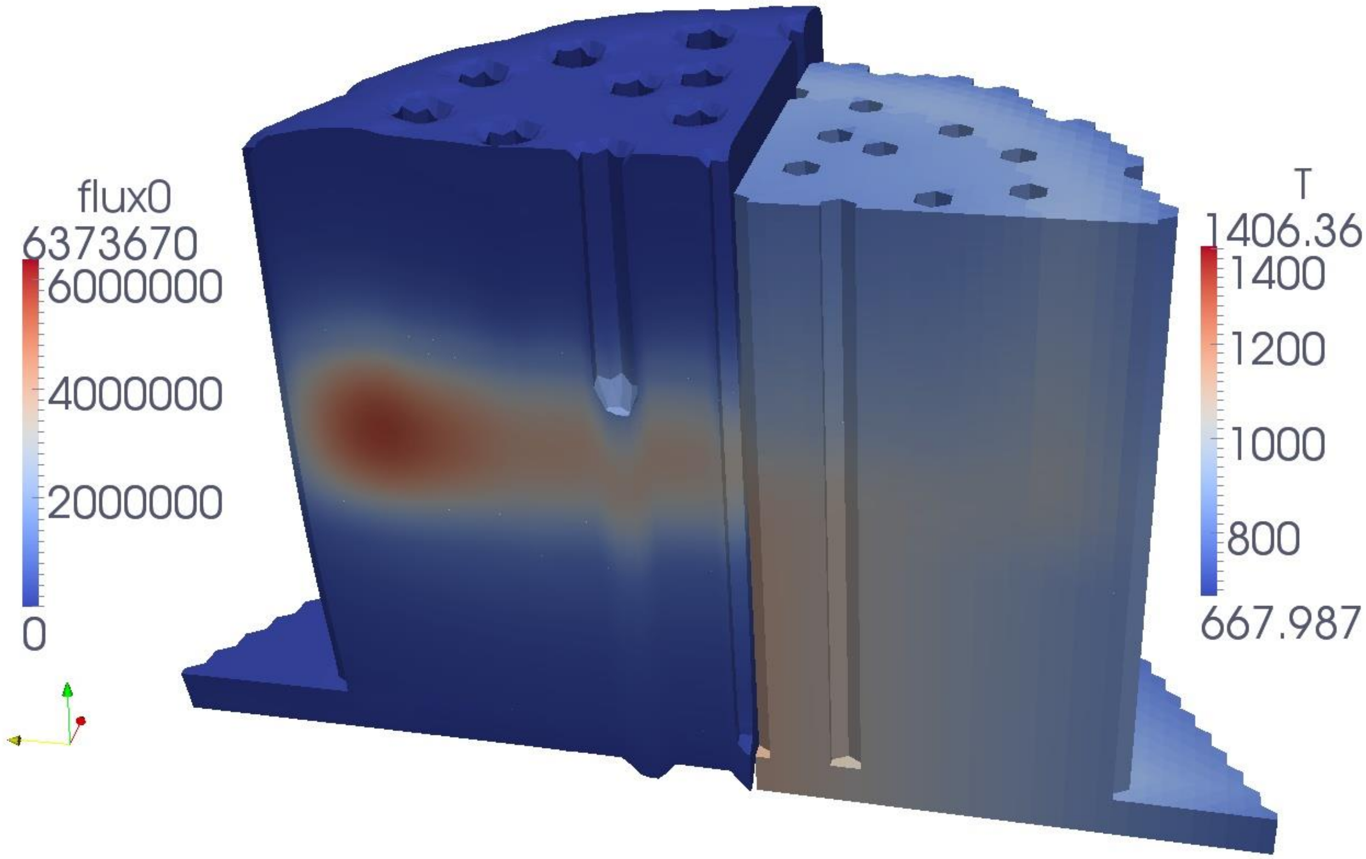
Examples – the ESFR

- Example of **a ULOF/ULOHS**



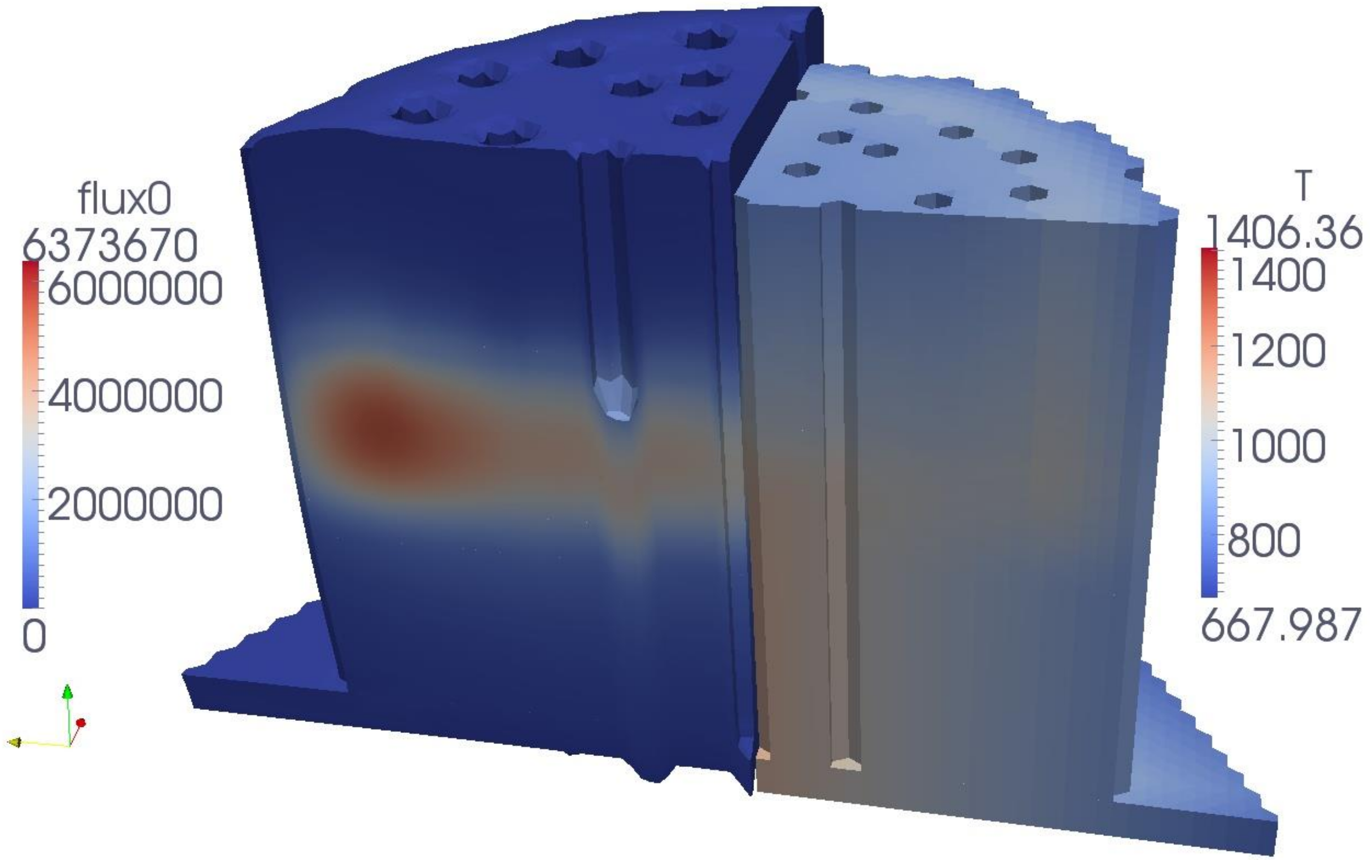
Examples – the ESFR

- Example of **a ULOF/ULOHS**



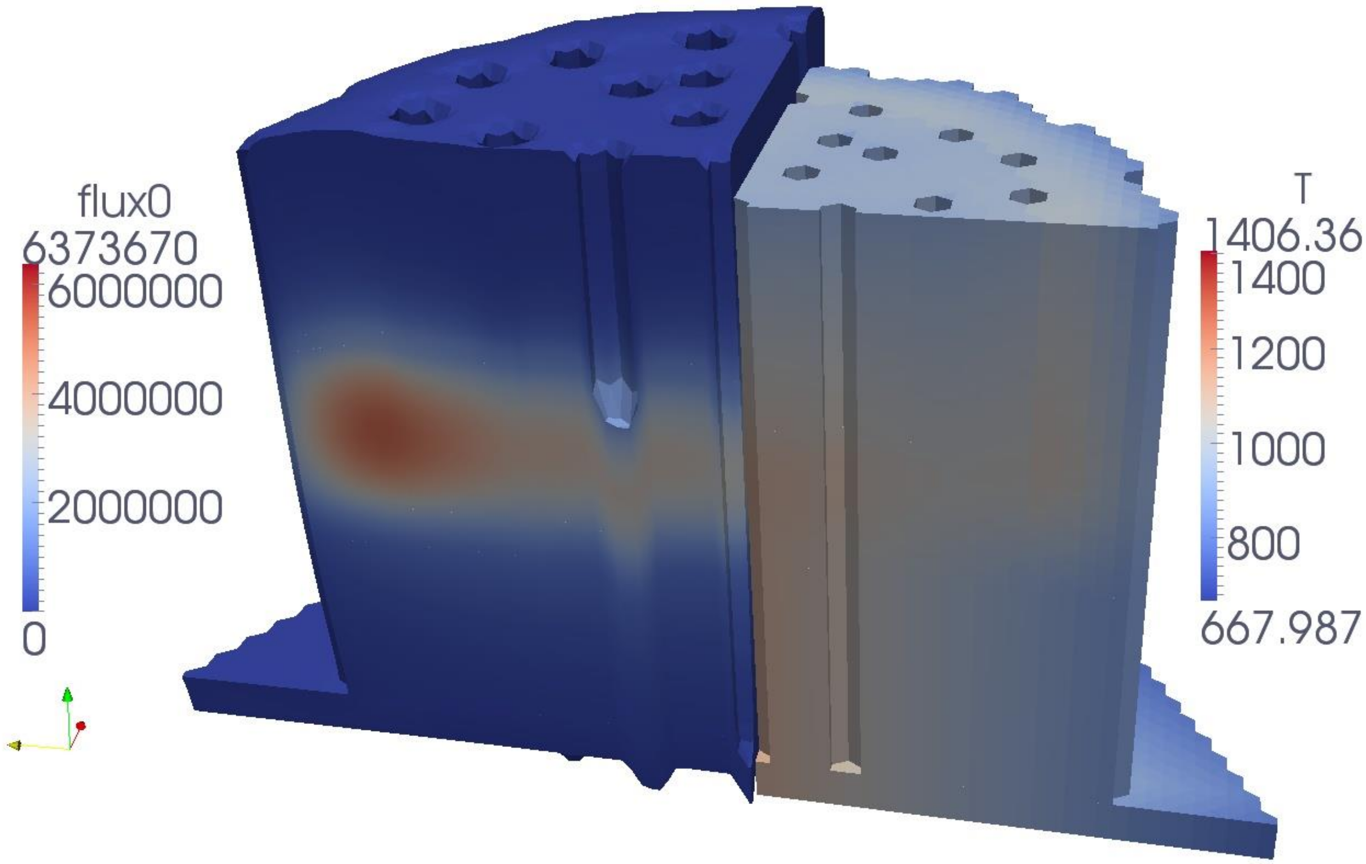
Examples – the ESFR

- Example of **a ULOF/ULOHS**



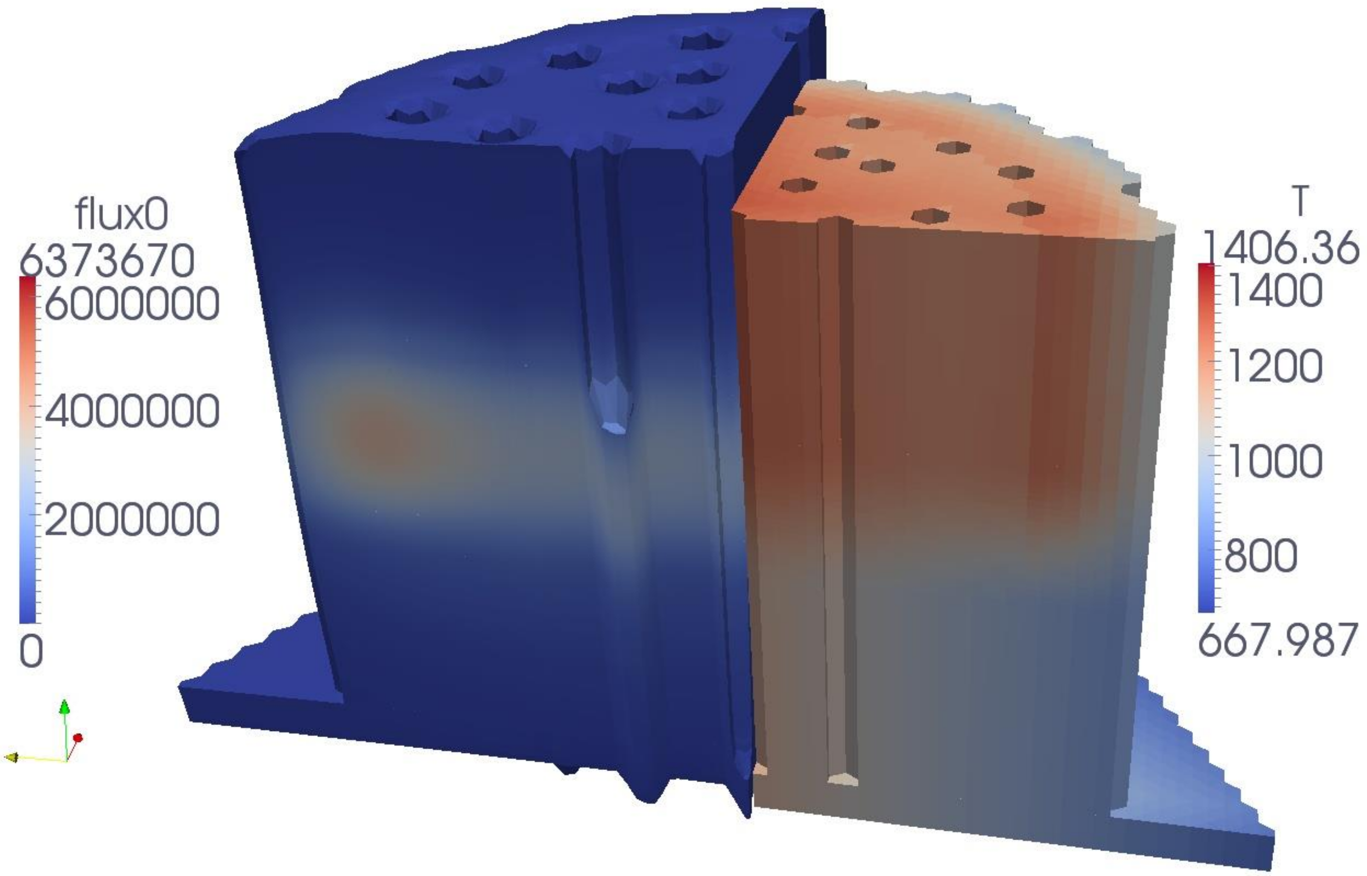
Examples – the ESFR

- Example of **a ULOF/ULOHS**



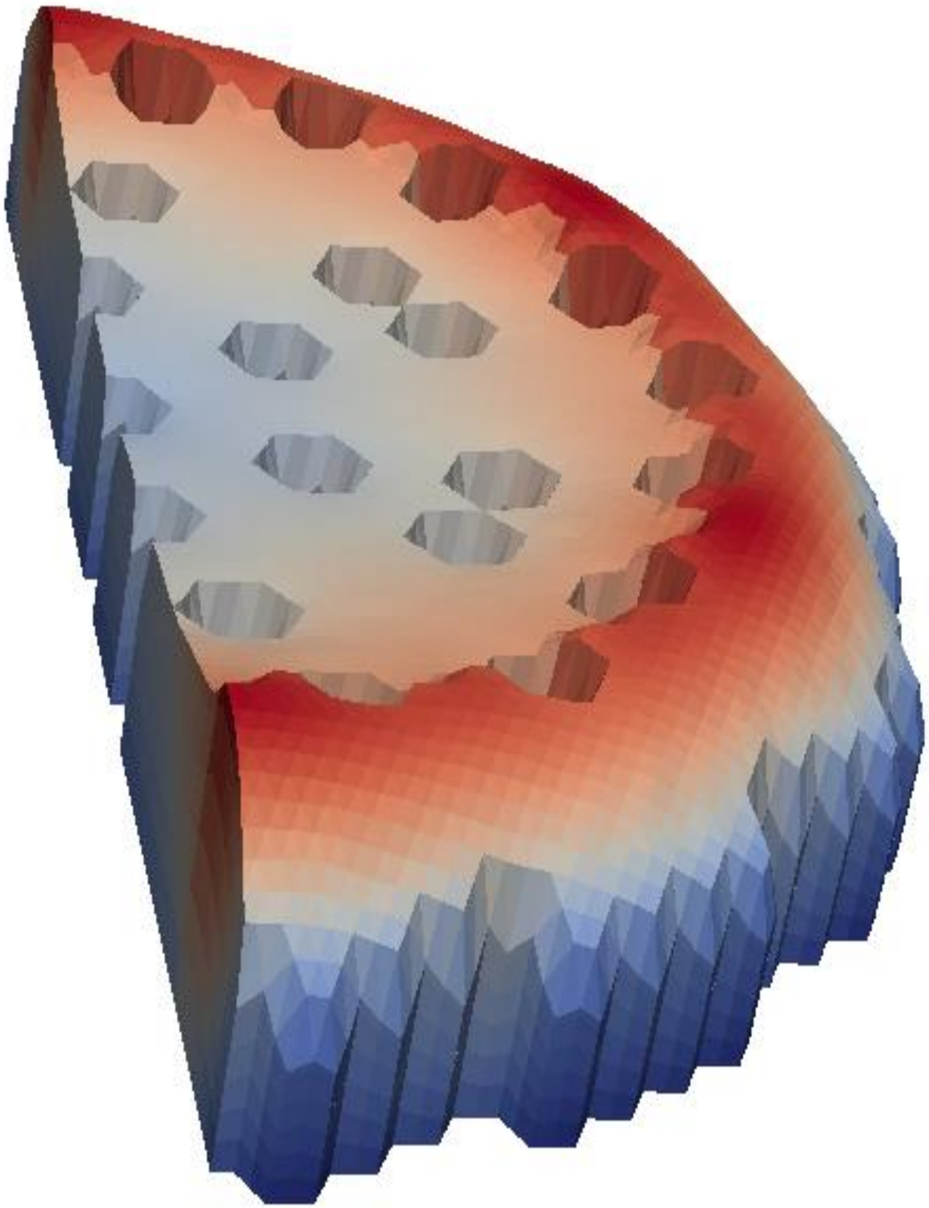
Examples – the ESFR

- Example of **a ULOF/ULOHS**



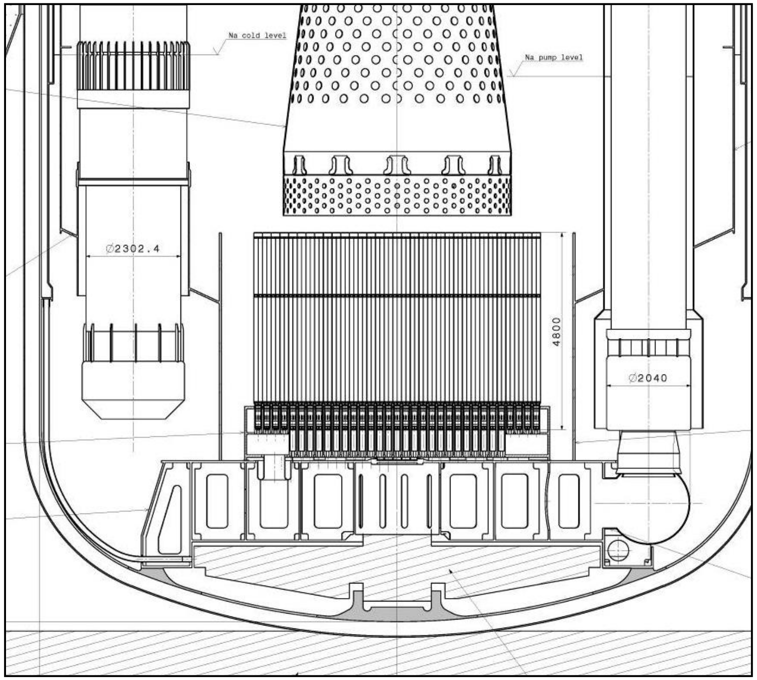
Outline

- Background
- The GeN-Foam solver
- Ongoing developments



Ongoing developments

- Coupling with the TRANSURANUS **fuel behavior solver**: collaboration with JRC/ITU and the University of Cambridge (external coupling)
- Inclusion of a **discrete ordinate** solver (?)
- Coupling with Serpent **Monte Carlo code for neutron transport**: collaboration with UC Berkeley
- **ROM** for control-oriented studies
- **Two-phase flow**
  - ✓ PROBLEM: fine-mesh Euler-Euler CMFD too computationally expensive (and not really appreciated by regulators)
  - ✓ -> need for coarse-mesh treatments



Coarse/fine mesh thermo-hydraulics

- **Coarse mesh with porous medium equations** for selected mesh zones (full NS + source terms):

$$\frac{\partial \gamma\rho}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}) = 0$$

$$\frac{\partial \gamma\rho\boldsymbol{u}}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}\otimes\boldsymbol{u}) = \nabla\cdot(\mu_T\nabla\boldsymbol{u}) - \nabla\gamma p + p_i\nabla\gamma + \gamma\boldsymbol{F_g} + \gamma\boldsymbol{F_{ss}}$$

$$\boldsymbol{F_{ss}} = \boldsymbol{\kappa}(\boldsymbol{u_D})\cdot\boldsymbol{u_D}$$

$$\kappa(u_D)_{ii} = \frac{f_{D,i}\rho u_{D,i}}{2D_h\gamma^2}$$

$$f_{D,i} = A_{f_{D,i}} Re^{B_{f_{D,i}}}$$

$$\frac{\partial \gamma\rho e}{\partial t} + \nabla\cdot\big(\boldsymbol{u}\gamma(\rho e + p)\big) = \nabla\cdot(\gamma k_T\nabla \mathrm{T}) + \gamma\boldsymbol{F_{ss}}\cdot\boldsymbol{u} + \gamma\dot{Q}_{ss}$$

$$\dot{Q}_{ss} = A_V h(T_{SS} - T)$$

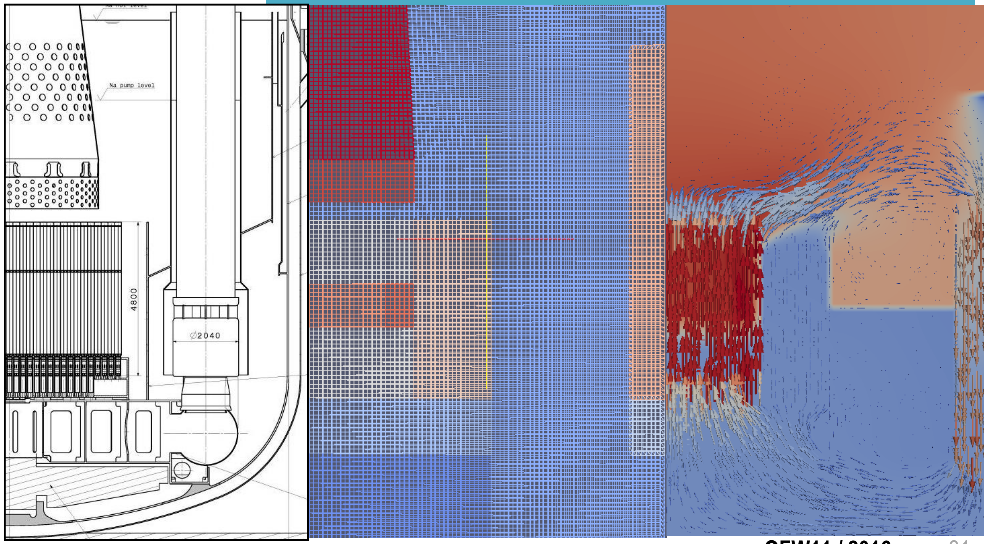
$$Nu_i = A_{Nu,i} Re^{B_{Nu,i}} Pr^{C_{Nu,i}} + D_{Nu,i}$$

$$\rho_{SS} c_{p,ss}\frac{\partial T_{SS}}{\partial t} = \nabla\cdot(\gamma\boldsymbol{k_{SS}}\nabla T) + A_V h(T - T_{SS})$$

- Equivalent to 1D in main flow direction, but possibility to investigate some 3D effects
- One set of equations (“porous terms” disappear in clear fluid): implicit “natural” coupling between porous and clear zones

```
fvm::ddt(rho, U)
+ (1/porosity)*fvm::div(phi, U)
+ turb.divDevRhoReff(U)
- porousMedium.momentumSource(U)

fvm::ddt(rho*porosity, he) +
fvm::div(phi, he) +
fvc::ddt(rho*porosity, K) +
fvc::div(phi, K) +
fvc::div(fvc::absolute(phi/fvc::interpo
late(rho), U),p,"div(phiv,p)")-
fvm::laplacian(turb.alphaEff(), he)
==
porousMedium.externalVolHeatSource()-
porousMedium.heatTransferCoefficient()*
porousMedium.volumetricArea()
*(thermo.T()-Tstructures) +
subscaleFuel.heatSources(…)
```



Coarse/fine mesh thermo-hydraulics

- **Coarse mesh with porous medium equations** for selected mesh zones (full NS + source terms):

$$\frac{\partial \gamma\rho}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}) = 0$$

$$\frac{\partial \gamma\rho\boldsymbol{u}}{\partial t} + \nabla\cdot(\gamma\rho\boldsymbol{u}\otimes\boldsymbol{u}) = \nabla\cdot(\mu_T\nabla\boldsymbol{u}) - \nabla\gamma p + p_i\nabla\gamma + \gamma\boldsymbol{F_g} + \gamma\boldsymbol{F_{ss}}$$

$$\boldsymbol{F_{ss}} = \boldsymbol{\kappa}(\boldsymbol{u_D})\cdot\boldsymbol{u_D}$$

$$\kappa(u_D)_{ii} = \frac{f_{D,i}\rho u_{D,i}}{2D_h\gamma^2}$$

$$f_{D,i} = A_{f_{D,i}} Re^{B_{f_{D,i}}}$$

$$\frac{\partial \gamma\rho e}{\partial t} + \nabla\cdot\big(\boldsymbol{u}\gamma(\rho e + p)\big) = \nabla\cdot(\gamma k_T\nabla T) + \gamma\boldsymbol{F_{ss}}\cdot\boldsymbol{u} + \gamma\dot{Q}_{ss}$$

$$\dot{Q}_{ss} = A_V h(T_{SS} - T)$$

$$Nu_i = A_{Nu,i} Re^{B_{Nu,i}} Pr^{C_{Nu,i}} + D_{Nu,i}$$

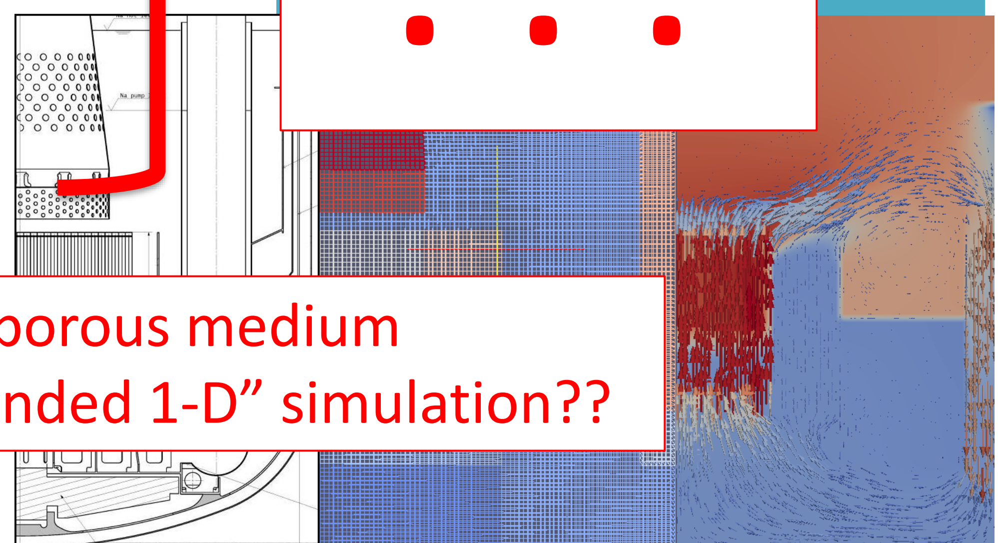
$$\rho_{SS} c_{p,ss}\frac{\partial T_{SS}}{\partial t} = \nabla\cdot(\gamma\boldsymbol{k_{SS}}\nabla T) + A_V h(T - T_{SS})$$

```
fvm::ddt(rho, U)
+ (1/porosity)*fvm::div(phi, U)
+ turb.divDevRhoReff(U)
(U)
c::interpo
-
, he)
ource()-
ficient()*
```

X 2

???

- Equivalent to 1D in main flow direction, but possibility to investigate some 3D effects
- One set o… in clear flu… porous and clear zones



Euler – Euler 2-fluids porous medium approach for an “extended 1-D” simulation??

Thank you for your attention

Carlo Fiorina – carlo.fiorina@epfl.ch