Development of an integrated Cast3M mesh generator in TACTICS

DE LA RECHERCHE À L'INDUSTRIE

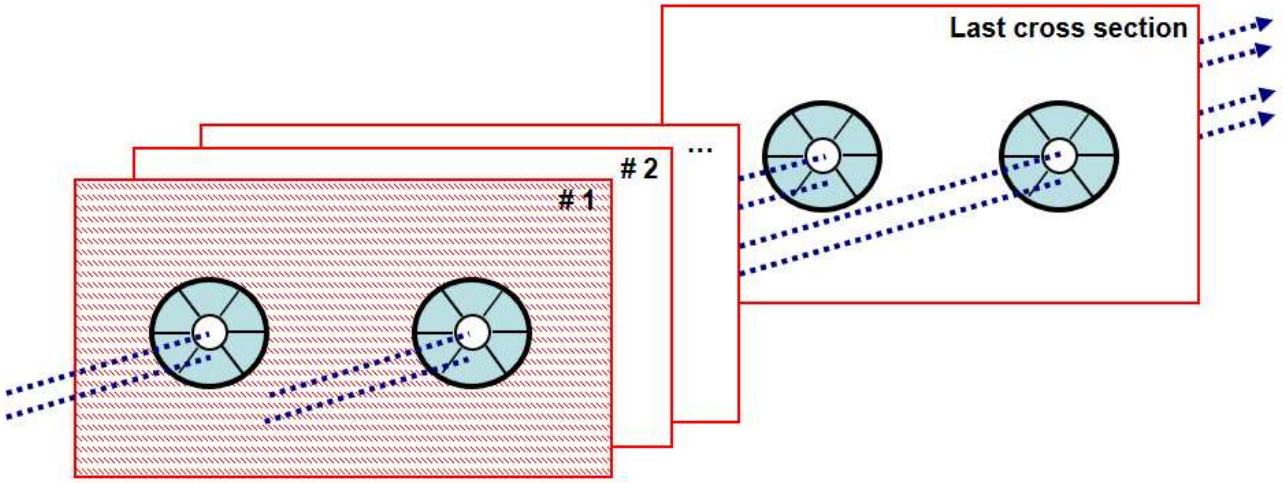
Benoît LACROIX[1], Guillaume GRANDMOUGIN[1], Quentin LE COZ[2], Alexandre LOUZGUITI[1], Sylvie NICOLLET[1], François NUNIO[1], Alexandre TORRE[1], Louis ZANI[1]

[1] CEA, [2] ASSYSTEM

Content

1 – Context and rationale for the project

- **Thermohydraulic analyses** for CICC fusion magnets, mainly in burn (ΔT_{ma}) and possibly in quench ($T_{hot\ spot}$) conditions, usually rely on a **quasi-3D approach**
  - ✓ Conductors: **He flow** thermohydraulics (1 or 2 channels) + **strands** (and possibly jacket) **thermal** treated in **1D longitudinal**
  - ✓ Structures: **transverse thermal** exchange (conductors/conductors and conductors/case) treated in **2D**



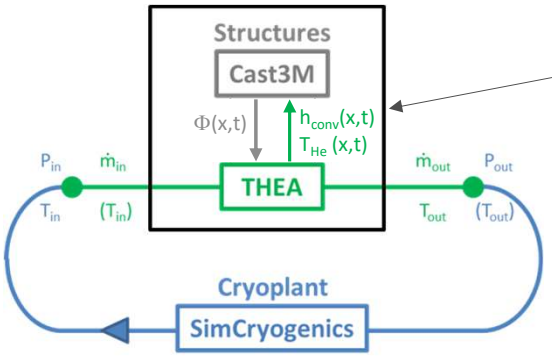
- This **approach** is the one **retained in existing tools**: SuperMagnet (CryoSoft), VENECIA (Alphysica), 4C (PoliTo), TACTICS (CEA)

TACTICS overview

TACTICS = THEA - Cast3M - SimCryogenics

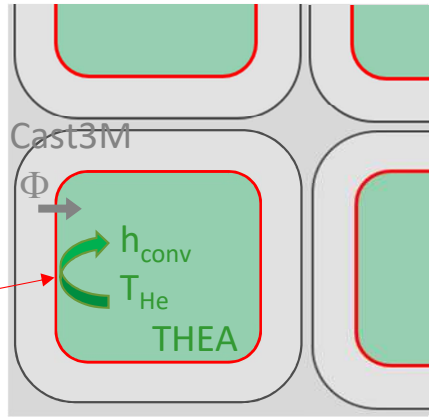
Quasi-3D transient calculation achieved by **co-simulation** of:

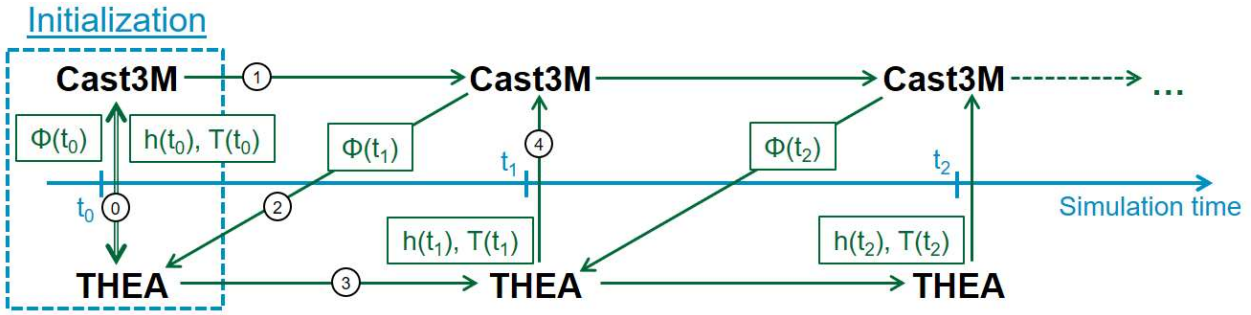
- **THEA** for 1D thermohydraulics in cables
- **Cast3M** for 2D finite elements transverse thermal diffusion
- **SimCryogenics** (CEA/DSBT software) for cryodistribution modelling



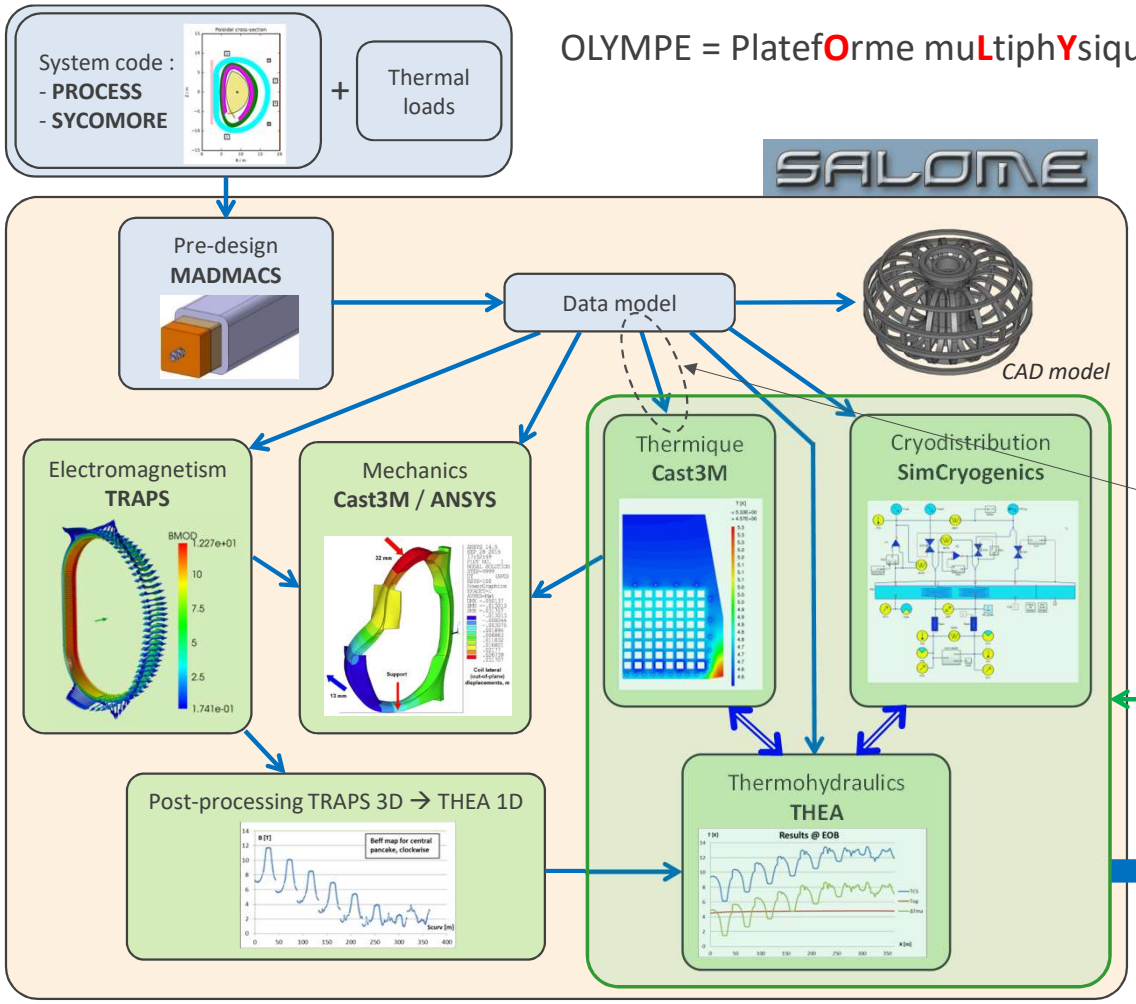
« TACTICS magnet » = THEA + Cast3M

Coupling interface between THEA and Cast3M = jacket inner surface





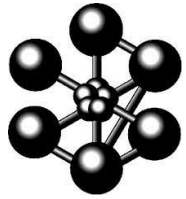
OLYMPE overview



OLYMPE = PlatefOrme muLtiphYsique pour aiMants suPraconductEurs

- **OLYMPE** main objective is to integrate and interface codes for:
  - ✓ facilitating the **conduct of multiphysic analyses**
  - ✓ allowing faster and easier **evaluation of several designs**
- The integration of TACTICS in OLYMPE, notably for multiple designs analyses, requires that **data for Cast3M** 2D finite elements model shall be **included in OLYMPE data model**

Cast3M overview

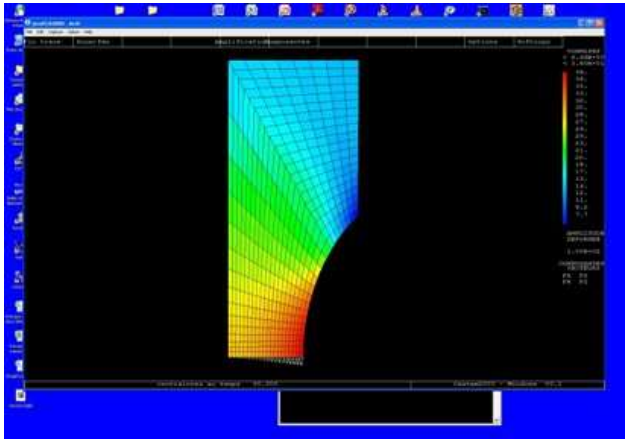


Cast3M Finite Elements code:

- ✓ Developed at CEA since the 1980s, first for mechanical and thermal analyses, then extended to other areas
- ✓ Organization of **one calculation** = building the **input file**
  - Construction of geometry, of meshing and of physical problem (model, materials, boundary conditions, etc.)
  - Resolution (in most cases, PASAPAS transient procedure)
  - Post-processing for results visualization and analysis
- ✓ Input file is written in **GIBIANE**, which is an interpreted, interactive and object oriented language
- ✓ A **GUI allows visualizing mesh and results maps**, but it does not allow direct **mesh generation** which must be **coded in GIBIANE** in the input file

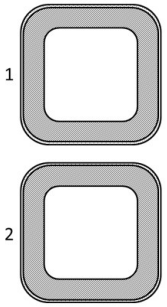
Example of GIBIANE instructions

```
* distance
*************
PGI5X = e_int_gi*(COS(BETA_GLR)) ;
PGI5Y = e_int_gi*(SIN(BETA_GLR)) ;
* points
XPGI5 = (D1_GLR_D coor 1) + PGI5X ;
YPGI5 = (D1_GLR_D coor 2) + PGI5Y ;
PGI5 = XPGI5 YPGI5 ;
si PR ;
C1_GLR_G = C1_GLR_G plus (PW ((EPCOND/2) + RPY.1 + PW + RP_R)) ;
C1_GLR_D = C1_GLR_G syme 'DROI' P_SYM1 P_SYM2 ;
sinon ;
C1_GLR_G = C1_GLR_G plus ((3*EPX_1/2) (EPY_1/2)) ;
C1_GLR_D = C1_GLR_G syme 'DROI' P_SYM1 P_SYM2 ;
finsi ;
```

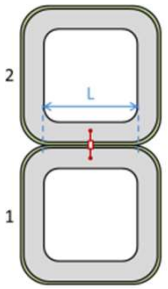


Status of Cast3M module in TACTICS

- 1 cast3M model per cross section for better use of multiple processors
- **Casings** (TF coil) - always **modelled for burn** analyses (1 model for each leg)
  - usually **not modelled for quench** analyses
- **Modelling of thermal coupling inside WP**

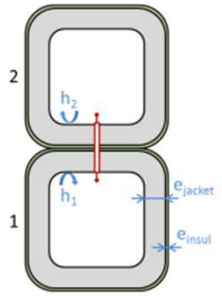


THEA
No transverse coupling



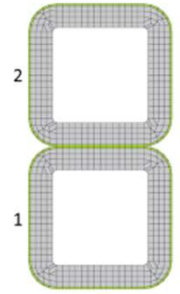
THEA with R_{th1}
Jacket/jacket coupling

$$\Phi_{tr}\left[{}^{W}/_{m}\right] = \Delta T_{jacket\ 1\leftrightarrow 2} \cdot \frac{\lambda_{insul} * L}{2 * e_{insul}}$$



THEA with R_{th2}
Bundle/bundle coupling

$$\Phi_{tr}\left[{}^{W}/_{m}\right] = \Delta T_{jacket\ 1\leftrightarrow 2} \cdot \frac{\lambda_{insul} * L}{2 * e_{insul}}$$



TACTICS
FE model

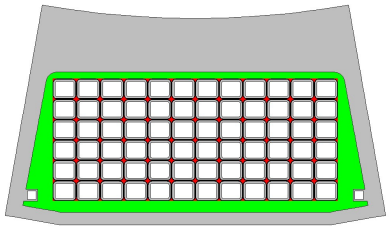
Results presented at MT-25: *"Q. Le Coz et al., Quench simulation of a DEMO TF coil using a quasi-3D coupling tool", 2017*

- THEA with bundle/bundle thermal coupling (R_{th2}) is suitable for burn simulations (almost stationary conditions)
- For **quench analyses, thermal diffusion must be considered** for providing accurate results (quench propagation, $T_{hot\ spot}$)

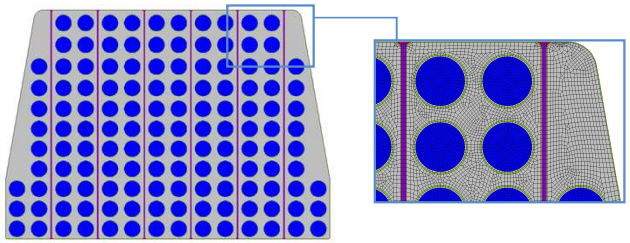
→ **WP internal structures** (jackets, insulating materials) are **always modelled in Cast3M** module of TACTICS

Project rationale

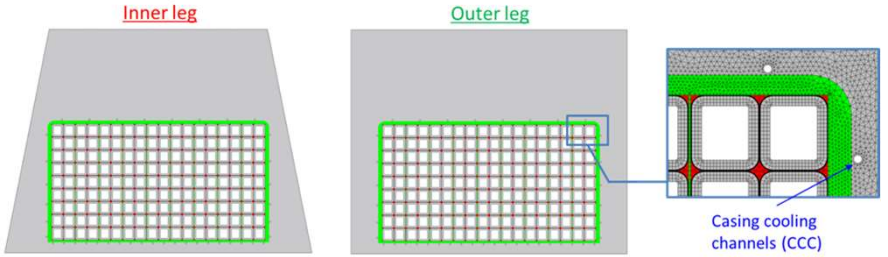
➢ Modelling of **different magnets**: each project treated separately → **lack of standardization** in GIBIANE input files



JT-60SA TF
Inner leg



ITER TF WP
Inner leg



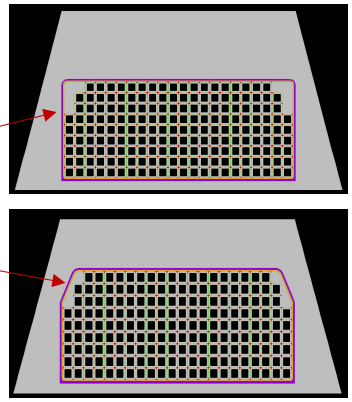
DEMO TF
CEA design proposal without radial plate

➡ Main motivations for rationalizing Cast3M mesh generation:

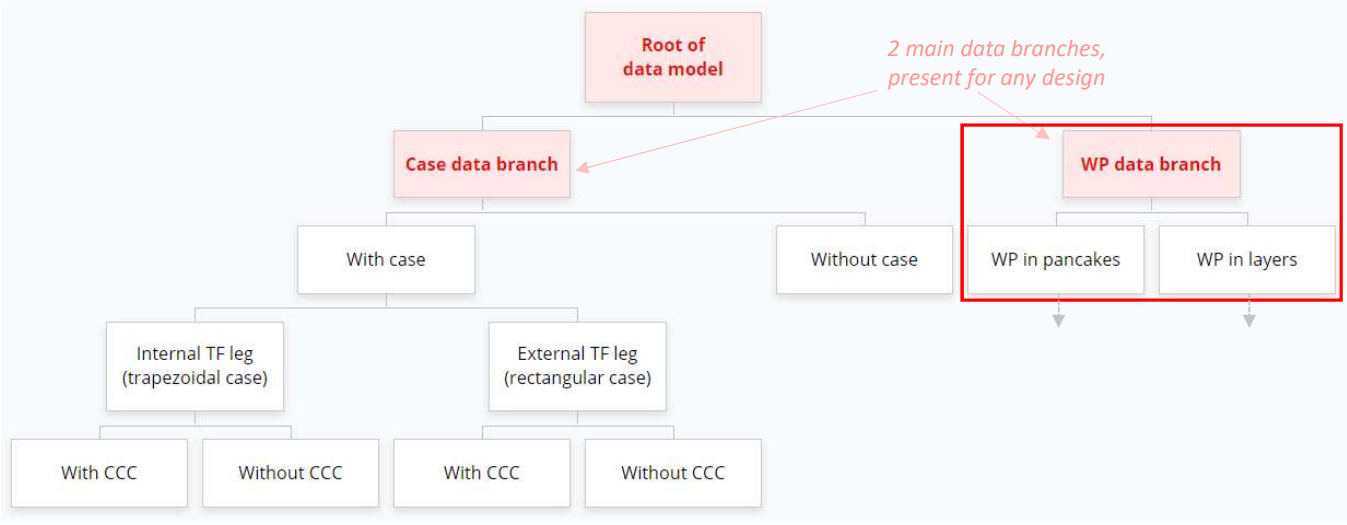
- ✓ Gather and externalize geometrical data, notably for preparing **connection with OLYMPE data model**
- ✓ **Mutualisation of developments and improvements** in geometry/mesh building (e.g. introduction of shorter lateral pancakes, management of network of case cooling channels)
- ✓ Make Cast3M module of TACTICS **more accessible** to non-specialits of Cast3M

2 – Database and associated tools

- **Many design features** shall be considered
  - **winding pack shape**:
    - presence of shorter lateral pancakes or not
    - if shorter lateral pancakes, rectangular or hexagonal WP
  - design with or without **radial plates**
  - presence of **casings** or not
  - **casing shape** (trapezoidal for inner TF leg or rectangular for TF outer leg)
  - **CCC network** (modelled or not)
  - ...

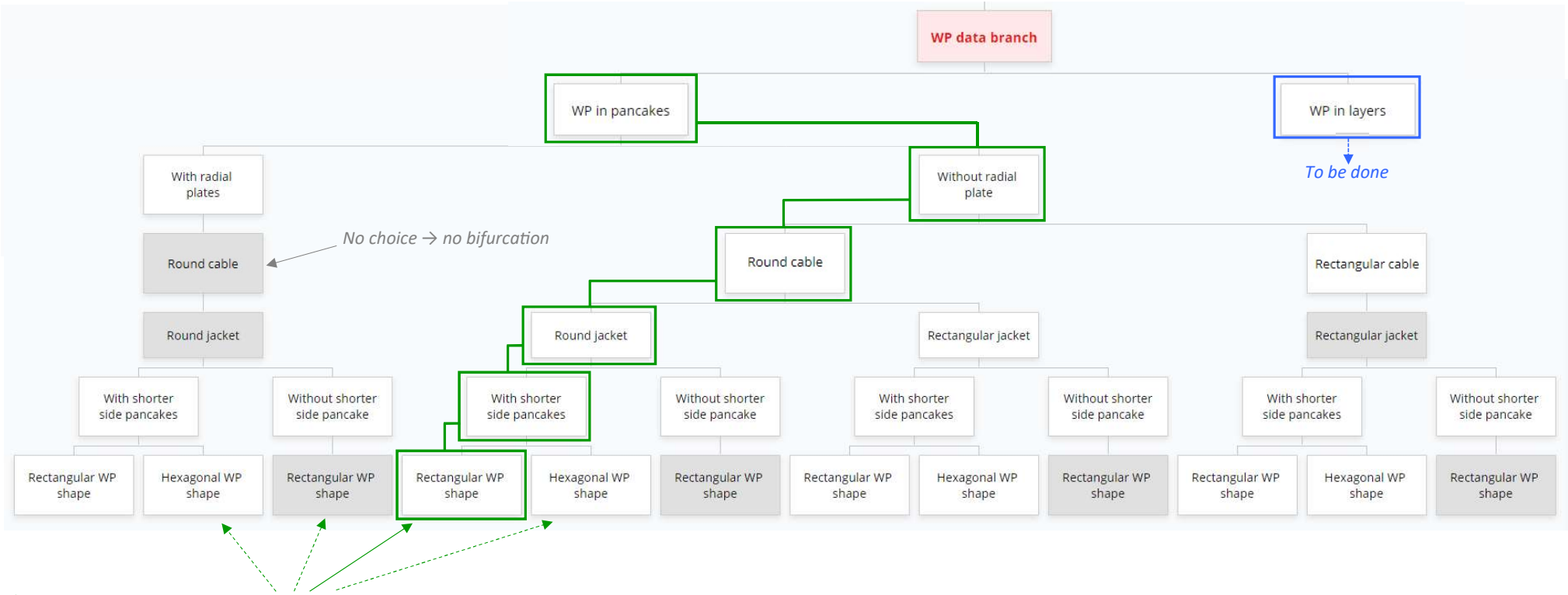


- The large number of design feature combinations calls for a **tree structure database**



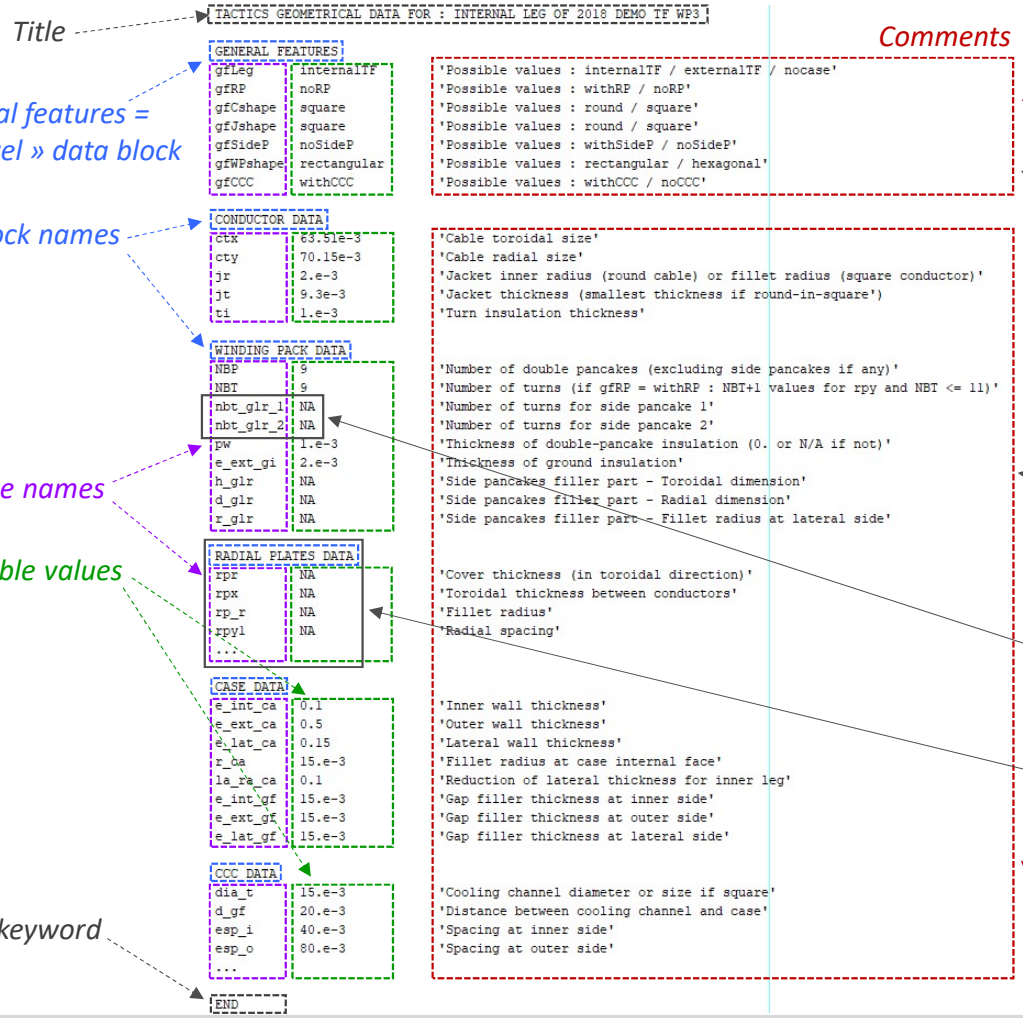
Database organization

- Status of WP data branch



- Each lowest level cell (“leaf”) corresponds to one combination of **general features** and to one **set of data** to complete
- Due to the **tree structure**, data management requires **adequate tools** (e.g. Packagespy Python module) or a **specific methodology**

Data file: geoTACTICS.dat

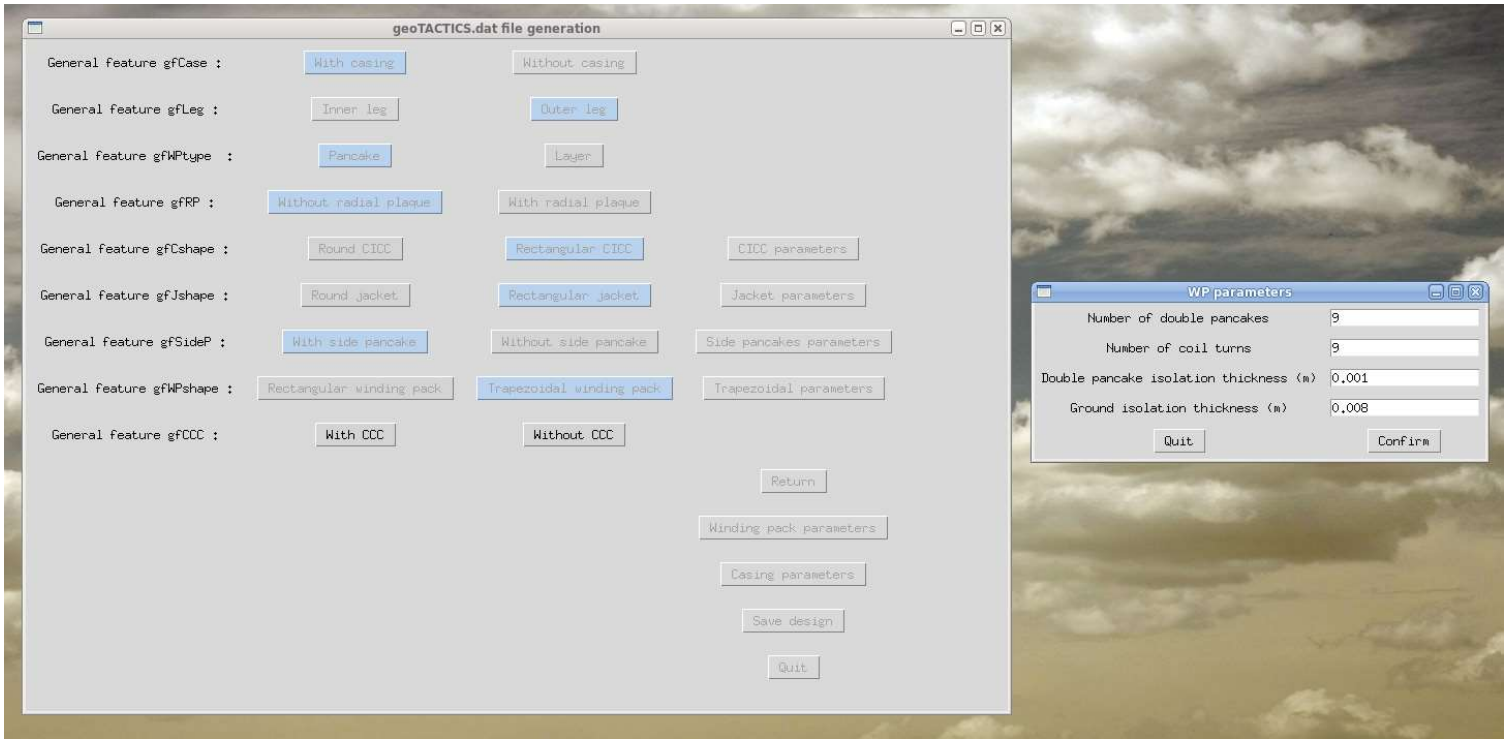


- **Only some combinations of general features are valid**, corresponding to one branch of the data tree, for example:
  - design with radial plates implies round CICC
    gfRP = withRP ⇒ gfCshape = round
  - design without shorter side pancakes implies rectangular WP
    gfSideP = noSideP ⇒ gfWPshape = rectangular
- **Data blocks are exhaustive:** they contain all variables corresponding to all valid sets of general features
- **Depending on the set of general features**, only some data shall be **completed**, others are **not applicable (NA)**, for example:
  - For a design without shorter side pancakes, no side pancake number is expected
    gfSideP = noSideP ⇒ nbt_glr_1 = NA and nbt_glr_2 = NA
  - For a design without radial plates, no data is expected in the radial plates data block

Tools developed for generating and/or checking data file

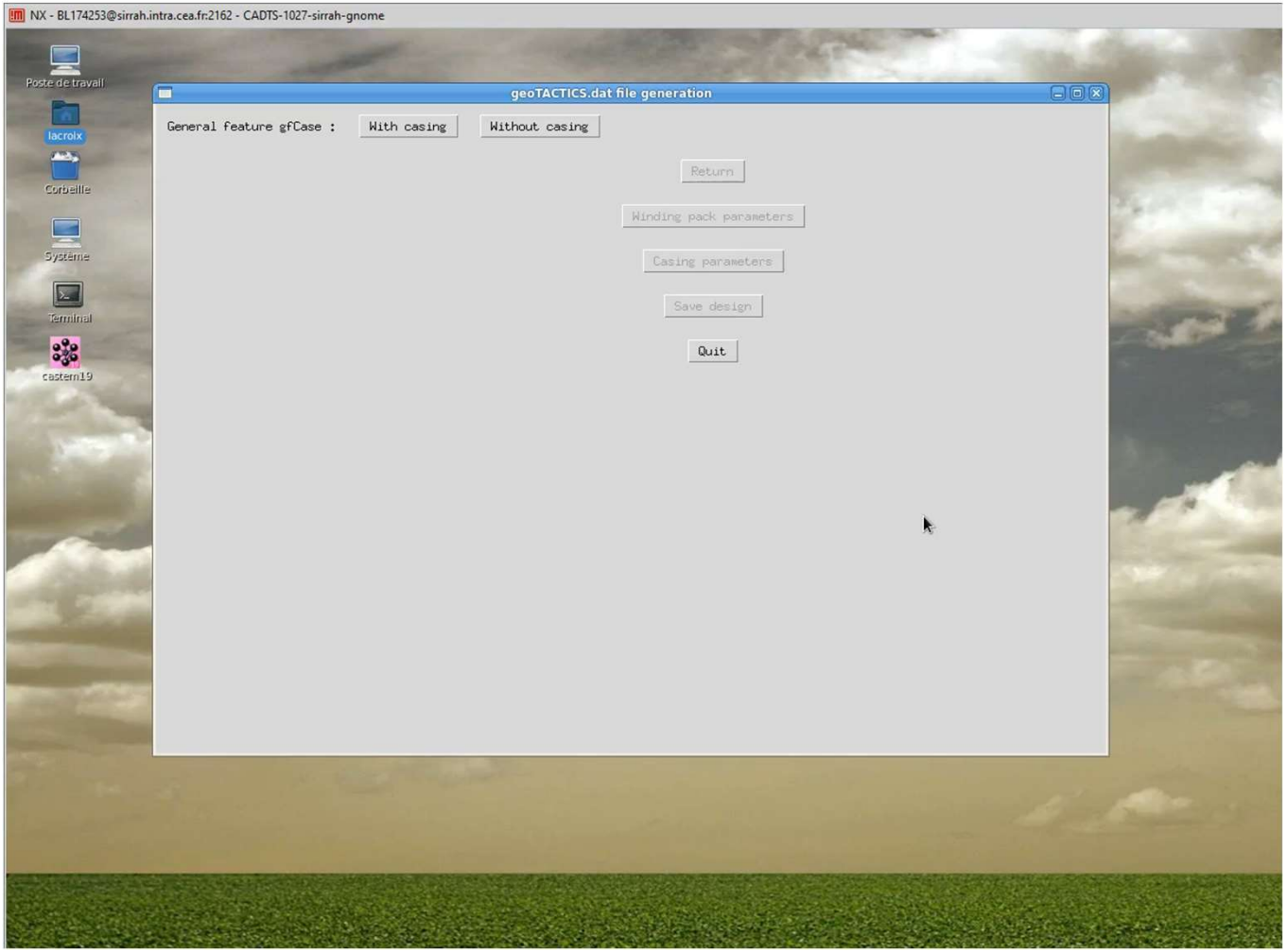
2 tools have been developed for validation of geoTACTICS.dat file

- **First tool**: a **G**raphical **U**ser **I**nterface has been developed in **Python**, using Tkinter module
  The **GUI** allows **complete generation** of the file and also **checks consistency**



Tools developed for generating and/or checking data file

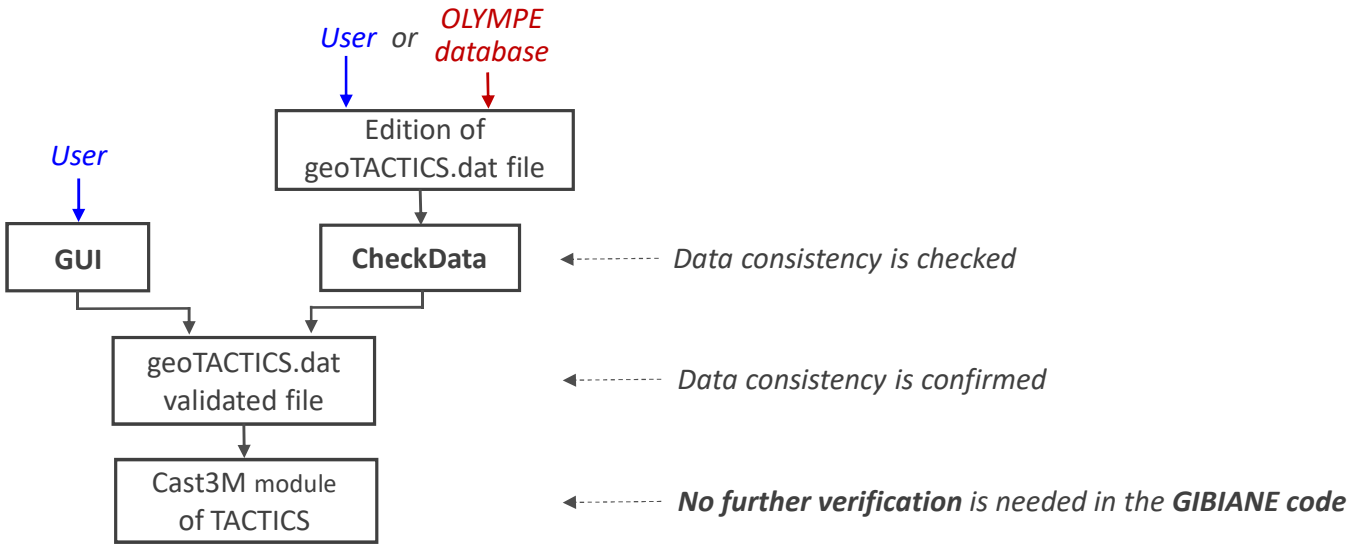
Running the GUI:



Tools developed for generating and/or checking data file

➢ **2nd tool**: A fortran **program *checkData*** has been coded **for validating a geoTACTICS.dat** file previously completed

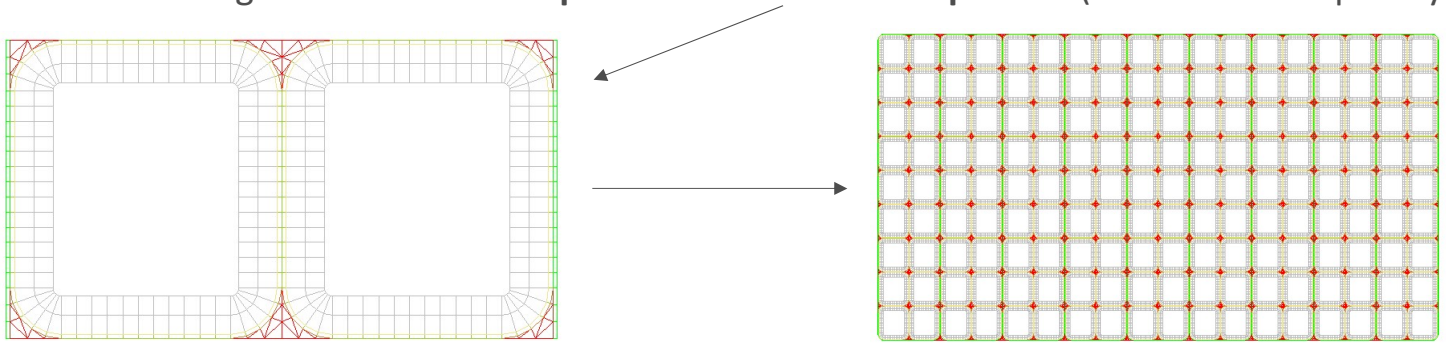
- Option 1:
  - the general features (GF) block of geoTACTICS.dat is filled
  - chekData program checks consistency of GF block, and stops if the combination of general features is invalid
  - checkData creates a template with “NA” or “?” at the right variable places
  - the user fills the template, replacing the “?” characters by values
- Option 2:
  - the whole geoTACTICS.dat file is filled
  - checkData **checks consistency of the whole file**, i.e. consistency of GF block and of “NA” variables



3 – Implementation of the mesh generator in Cast3M

A new version of the Cast3M Gibiane input file has been developed

- The part of the code dedicated to **geometry and mesh building** has been **unified and rewritten**
- Geometrical **features and data** are read from **geoTACTICS.dat** file
- Some parameters are still **entered directly in the Gibiane** code:
  - characteristic dimensions for defining **finite elements sizes**, for example 1 mm for jacket typical dimension will lead to 2 elements through a jacket of 2 mm thickness
  - coefficients allowing **local mesh refinement**
  - choice for **detailed or merged insulating materials** (pure resin, G10, conductors insulation, ground insulation) inside the winding pack
- The WP mesh building is based on the **duplication of a structural pattern** (method developed by F. Nunio IRFU)



Example of different levels of mesh detail

« Finest » mesh
- 3 elements in jacket thickness
- Insulation materials detailed

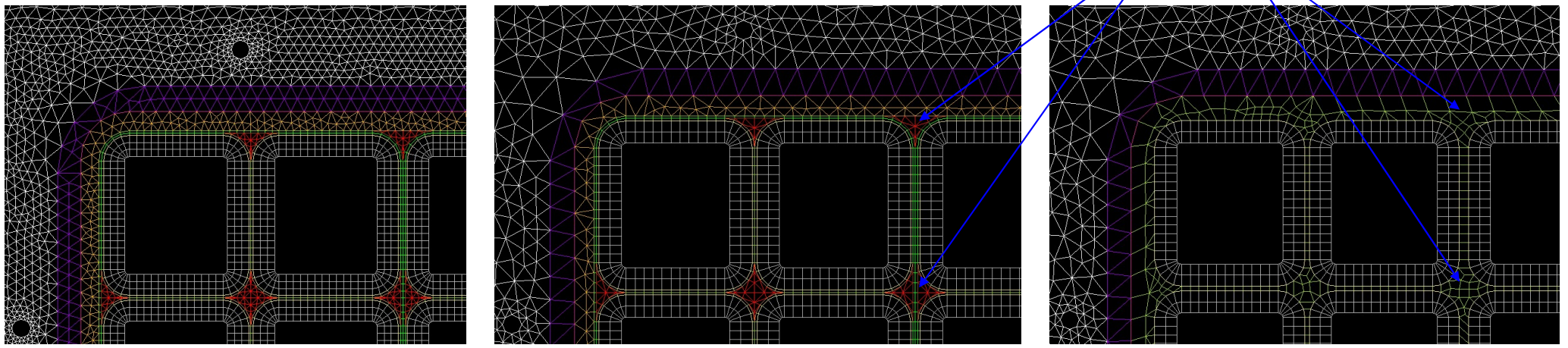
→ 89777 elements

« Normal » mesh
- 2 elements in jacket thickness
- Insulation materials detailed

→ 65062 elements

Mesh with merged insulation
- 2 elements in jacket thickness
- Insulation materials merged

→ 51329 elements

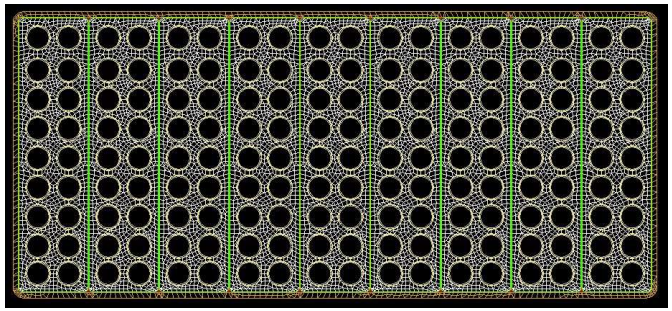


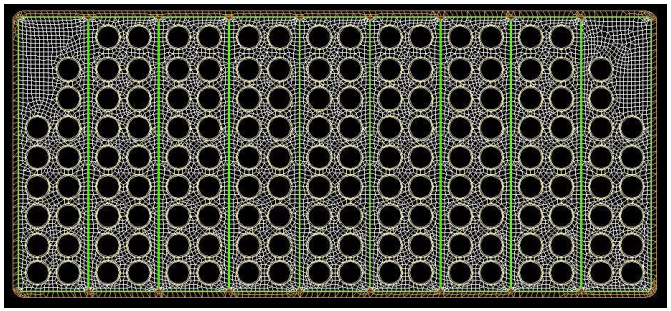
Simulation time reduced by ~ 30 % - 40 %

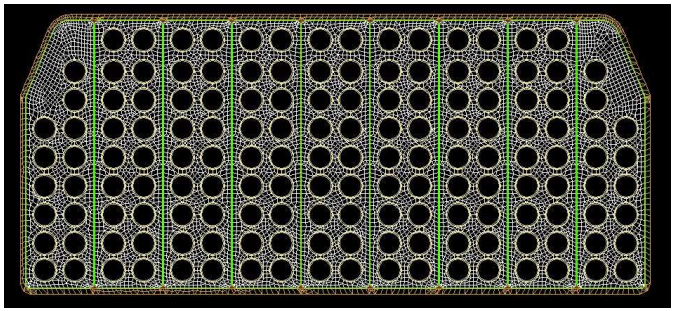
Simulation time reduced by ~ 50 %

➢ **Merging WP insulation materials** implies **same physical properties** and **loss of orthotropy**
- ✓ The impact was found little and totally **acceptable for thermal analyses**, both in burn and quench conditions
- ✓ But it is **not suitable for mechanical analyses**

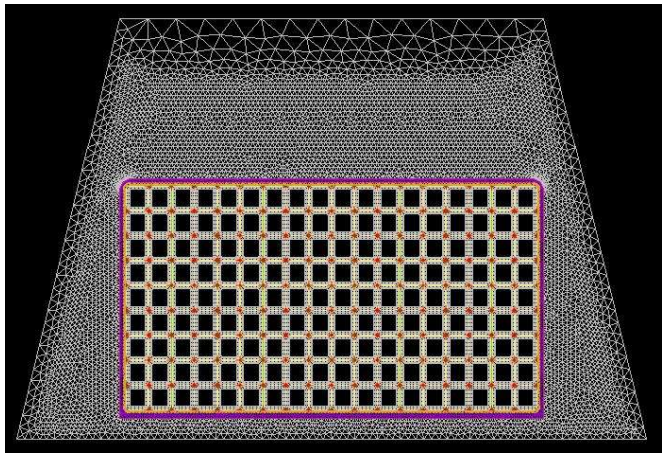
Example of meshes for different designs

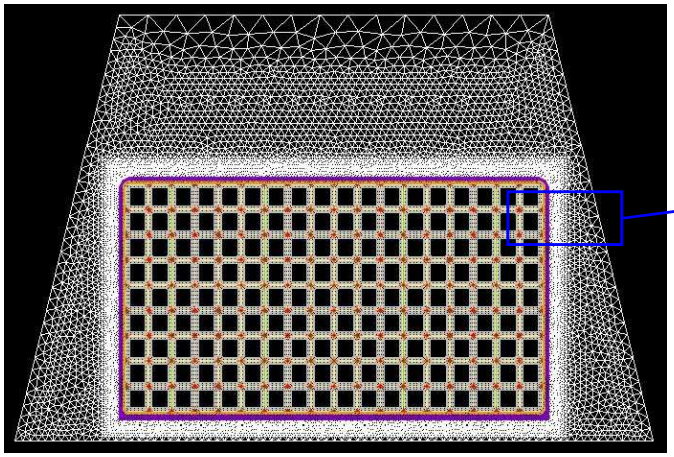


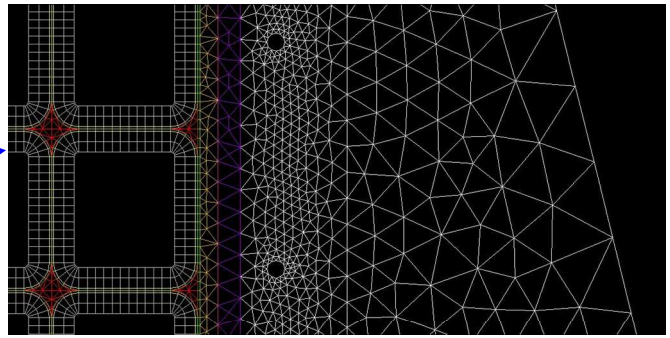




WP with radial plates → *with shorter side pancakes* → *with hexagonal shape*







Inner leg TFC without RP nor CCC → *With CCC*

Different types of design can be generated, even by a user with moderate practice of Gibiane and Cast3M

Content

4 – Conclusion

Overview

- An **integrated mesh generator** has been developed for Cast3M module of TACTICS
- It is **usable by a non-specialist** of Cast3M
- It provides mesh building for **pancake designs**:
  - with or without radial plates
  - with detailed or merged modelling of insulation materials of WP
- Data are **gathered in a dedicated file** (geoTACTICS.dat), allowing easier **interfacing with OLYMPE** data model
- The tree data structure has motivated the development of a GUI and of a verification tool (chekData)

Perspectives

- Implement **WP in layers** (larger amount of data)
- Improve the Gibiane code for **limiting user's adjustments**
- **Set the limits** of automation and **finalize the coding** for features which can take various forms, such as **casing shape** or **case cooling channels network**

Thank you for your attention