HIDALGO2
CENTRE OF EXCELLENCE

CoE Hidalgo2 Urban Building Pilot

University of Strasbourg - ICUBE

Christophe Prud’homme, Vincent Chabannes, Abdoulaye Diallo, Philippe Pincon, Maryam Maslek, Patrick Lemoine, Javier Cladellas
March 28, 2024

EuroHPC Joint Undertaking

Grant number: 101093457

HiDALGO2
CENTRE OF EXCELLENCE

CoE HiDALGO2

Partners

8 partners from 7 countries:

- **Instytut Chemii Bioorganicznej Polskiej Akademii Nauk (PSNC)** – Poland – Coordination
- **University Of Stuttgart (USTUTT)** – Germany – Technical Coordination
- **ATOS Spain SA (ATOS)** – Spain – Quality Manager
- **Szechenyi Istvan Egyetem (SZE)** – Hungary – Use case owner
- **Meteogrid SL (MTG)** – Spain – Use case owner
- **Universite De Strasbourg (UNISTRA)** – France – Use case owner
- **Erevnitiko Panepistimiako Institouto Systimaton Epikoinonion Kai Ypolgiston-EMP (ICCS)** – Greece – Data analytics and AI support
- **Future Needs Management Consulting LTD (Future Needs)** – Cyprus - Dissemination

Use cases

URBAN AIR PROJECT

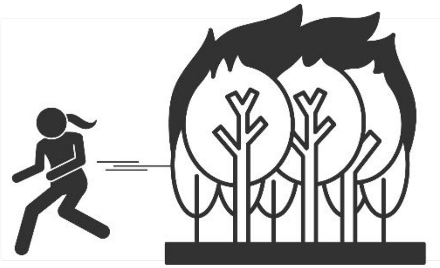
Evolution of the air in the urban areas considering pollution, wind, comfort and planning

URBAN BUILDINGS

Advanced building models for better integration with architecture. Providing a source term for heat and air pollutants (CO_2 and NO_x) to the urban air pollution model.

RENEWABLE ENERGY SOURCES

Energy production from renewable sources like wind and solar panels. Solution accustomed to urban and rural areas.



WILDFIRES

Simulation of wildfire atmosphere interactions and smoke dispersion in forest and urban areas.

Outline

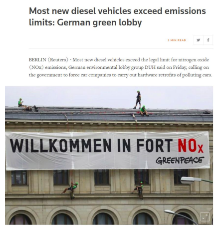
- Urban Air Pollution Pilot
- Urban Building Pilot
    - Overview and Objectives of the Urban Building Pilot
    - Workflow presentation
    - Mesh and Partitioning
    - Modeling and discretization
    - Post-processing and data exchange (coupling) I/O
    - Demo

HiDALGO2
CENTRE OF EXCELLENCE

Urban Air Project

Global challenges with the urban air flow: urban air pollution

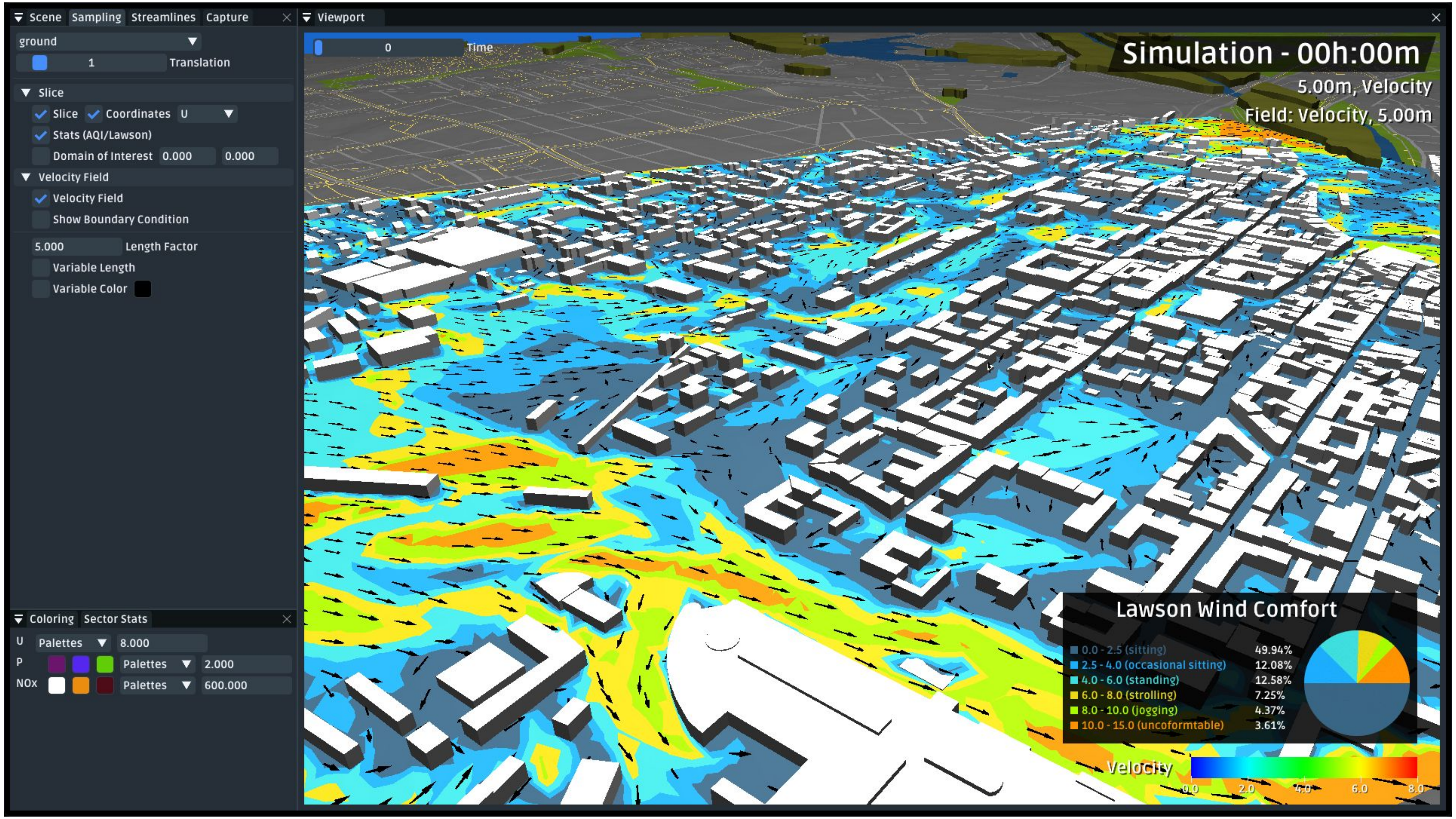
1. Importance of urban challenges: 55% of the world's population lives in urban areas now, and 68% by 2050, says UN, see UN DESA at https://population.un.org/wup/.
2. Challenge 1: Air quality
   - WHO: 6.7 million deaths are attributable each year to exposure to ambient and household air pollution, see https://www.who.int/data/gho/data/themes/air-pollution.
   - Many cities are polluted (air quality values are above health-critical values) □ Protests by citizens and new, stricter regulations by policymakers
3. Challenge 2: Wind comfort and safety
   - Urban wind can cause discomfort for pedestrians and even critical safety situations in particular near high buildings
4. Challenge 3: Urban planning
   - How can urban policymakers mitigate or cease the negative effects of urban challenges?


Most new diesel vehicles exceed emissions limits: German green lobby

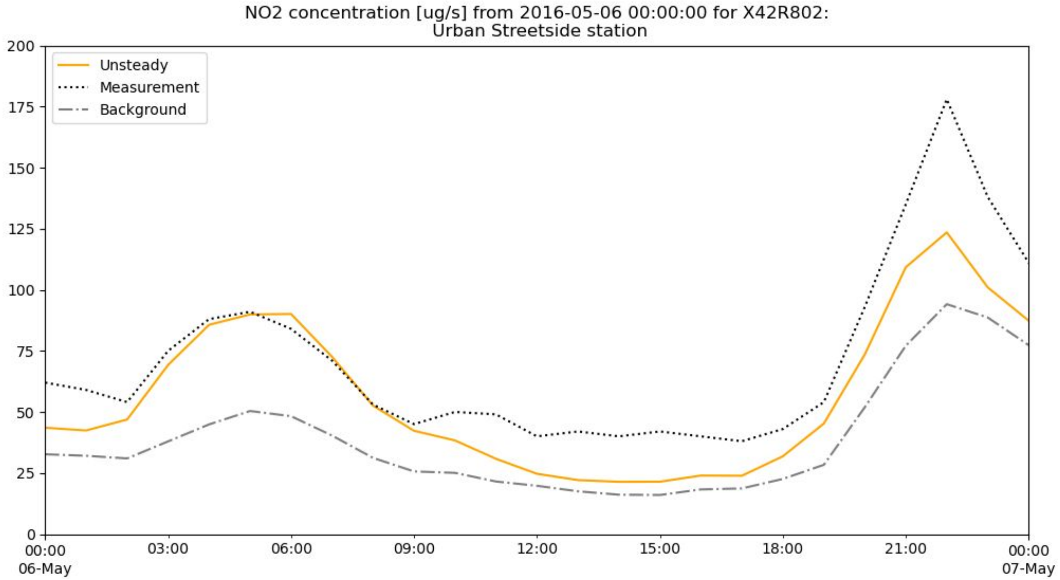
Reuters

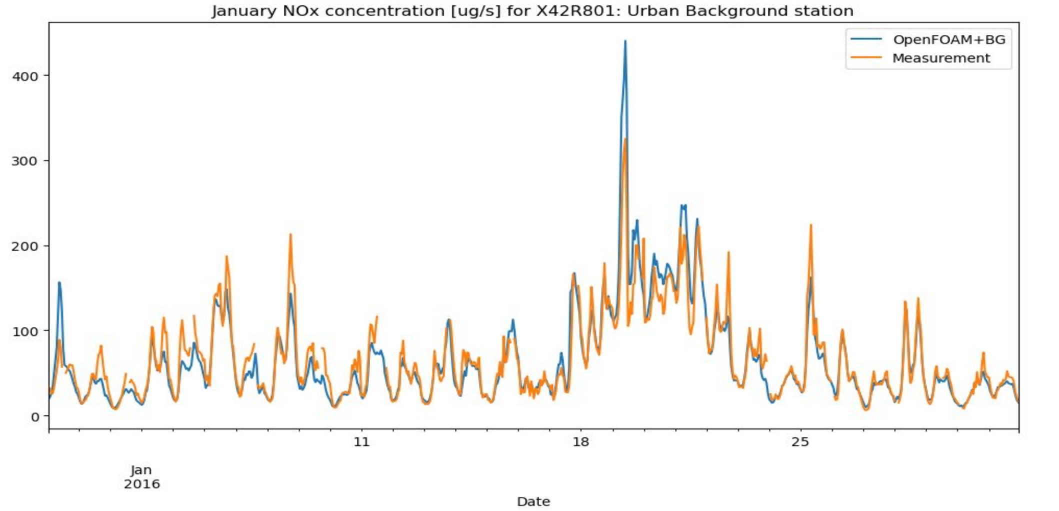
Freepik

Visualization



UAP, CFD Module with OpenFOAM
Validation on FAIRMODE Intercomparison exercise on Antwerp

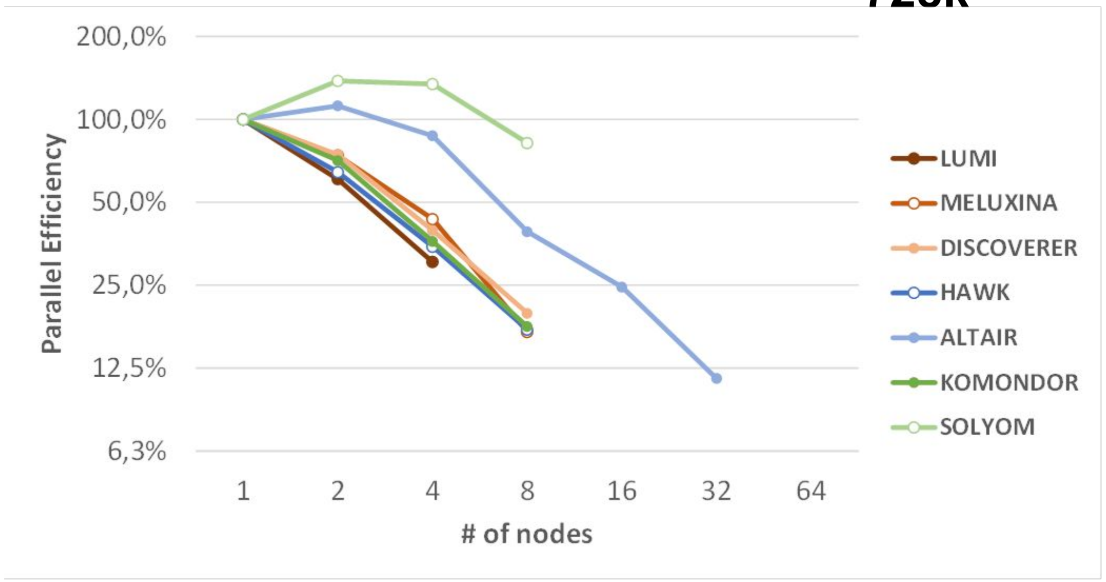




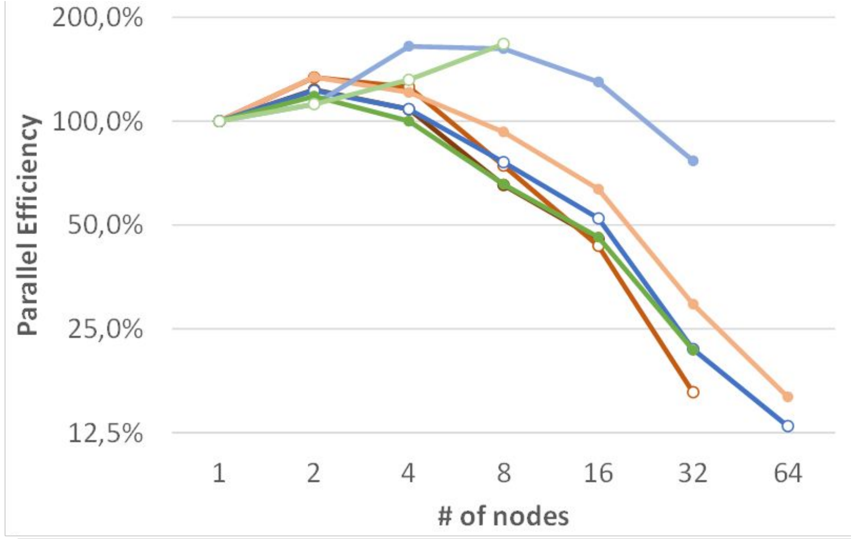
regarding

Benchmarking: Parallel Efficiency with OpenFOAM

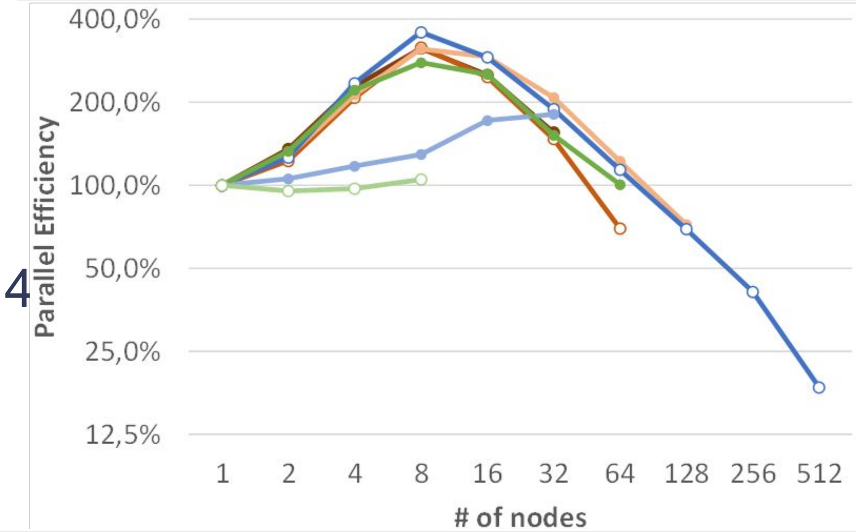
728k



3.4M



14M



- Győr geometry with mesh sizes 728k, 3.4M, 14
- Testruns on 7 architectures
- Parallel efficiency with regard to 1 node

RedSIM: solver for the compressible Euler and Navier-Stokes equation 2nd order FVM on polyhedral meshes

Code features

- Written in C99-style, compiled as C++, avoids C++ features (STL, templates, exceptions, ...)
- Parallel implementations: MPI+Pthread, multi-GPU (genuinely in CUDA)
- Optimized code, with no dependency, runs easily on each platform we tried (HPC, ...)
- One single algorithm and code for 1D, 2D, 3D
- POD-DEIM model order reduction mode
- Integrated in-house real-time 3D visualizer, interactive (digital twin) feature

Efficiency: Urban Airflow, Gyor. **N = 30,291,099**		
efficiency	**KOMONDOR**, 4x NVIDIA A100-SXM4-40GB	**VEGA**, 4x NVIDIA A100-SXM4-40GB
P = 1	1	-
P = 2	0.9449242544	-
P = 4	0.8752212861	-

Efficiency: Urban Airflow, Gyor. **N = 18,342,623**		
efficiency	KOMONDOR, 4x NVIDIA A100-SXM4-40GB	VEGA, 4x NVIDIA A100-SXM4-40GB
P = 1	1	1
P = 2	0.9225121646	0.9742910708
P = 4	0.8208693259	0.4716167595

Efficiency: Urban Airflow, Gyor. **N = 3,154,126**		
efficiency	KOMONDOR, 4x NVIDIA A100-SXM4-40GB	VEGA, 4x NVIDIA A100-SXM4-40GB
P = 1	1	1
P = 2	0.7968193341	0.9222091521
P = 4	0.5919466096	0.4022917729

Wall-Clock Time: Urban Airflow, Gyor. **N = 18342623**			
wall-clock (s)	ONDOR, 4x NVIDIA A100-SXM4-4	, 4x NVIDIA A100-SXM4-	SOLYOM, 1x NVIDIA Tesla V100S-PCIE-32GB
P = 1	2781.590725	2783.904732	2127.01869
P = 2	1507.617369	1428.682257	NO DATA
P = 4	847.147846	1475.724026	NO DATA
Wall-Clock Time: Urban Airflow, Gyor. **N = 3154126**			
wall-clock (s)	**KOMONDOR**	**VEGA**	**SOLYOM**, 1x NVIDIA Tesla V100S-PCIE-32GB
P = 1	432.583269	433.15866	276.966294
P = 2	271.443758	234.848385	NO DATA
P = 4	182.695222	269.181903	NO DATA

HIDALGO2
CENTRE OF EXCELLENCE

Urban Building Overview & Objectives

Context

Building sector in the EU [1]:

- **36%** of GHG emission
- **40%** of final energy consumption

Horizon 2050 objectives:

- Double annual energy renovation rates in the next 10 years [2]
- E.g. 700 000 renovation/year in France

➔ Building Energy simulation:

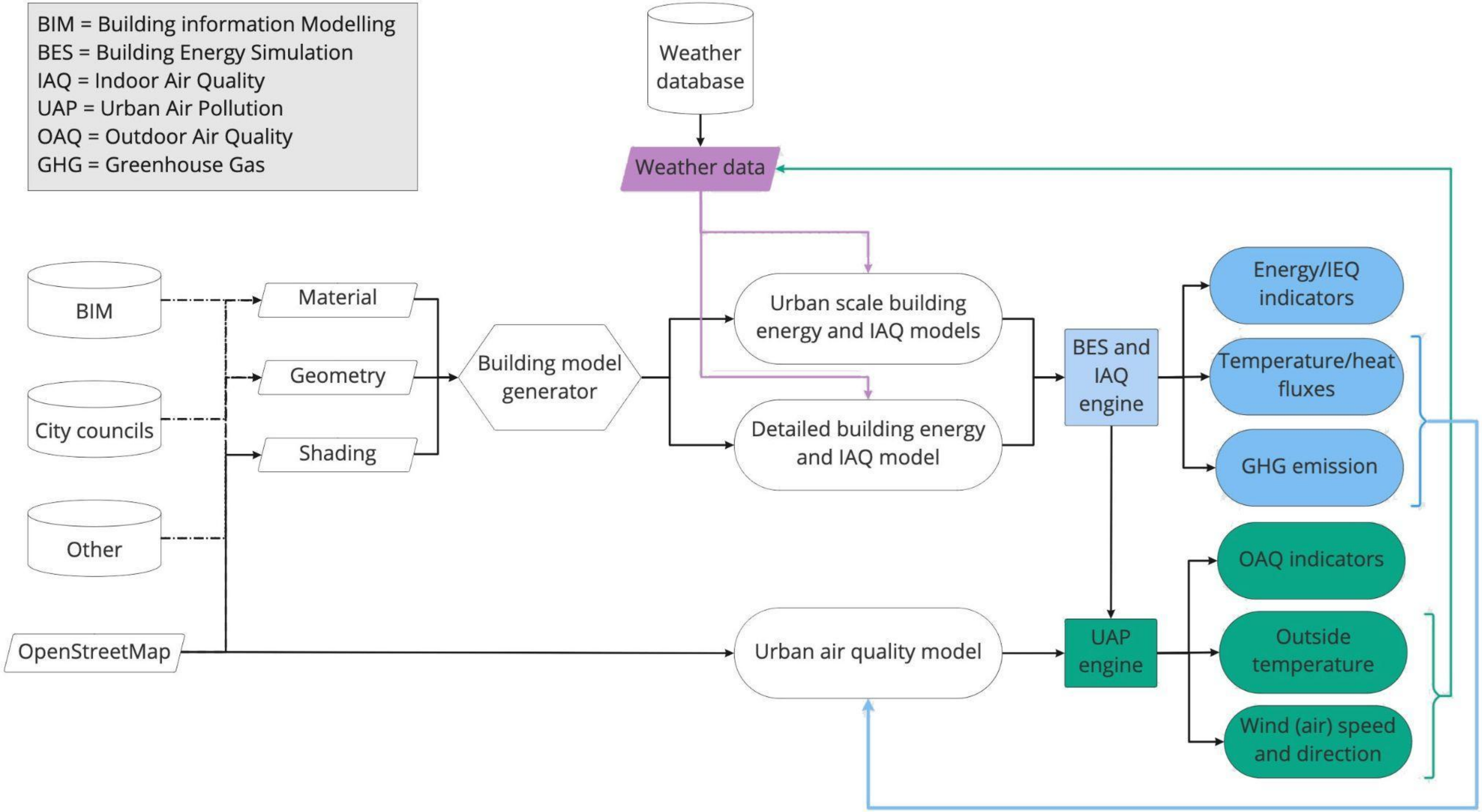
- Accurately assess energy performance of existing buildings
- Identify sources of energy savings (anomalies and areas for improvement)
- Compare and evaluate renovation and/or energy management strategies
- Ensure the optimal management of buildings

[1]: https://ec.europa.eu/info/news/focus-energy-efficiency-buildings-2020-lut-17_en
[2]: https://energy.ec.europa.eu/topics/energy-efficiency/energy-efficient-buildings/renovation-wave_en

Objectives

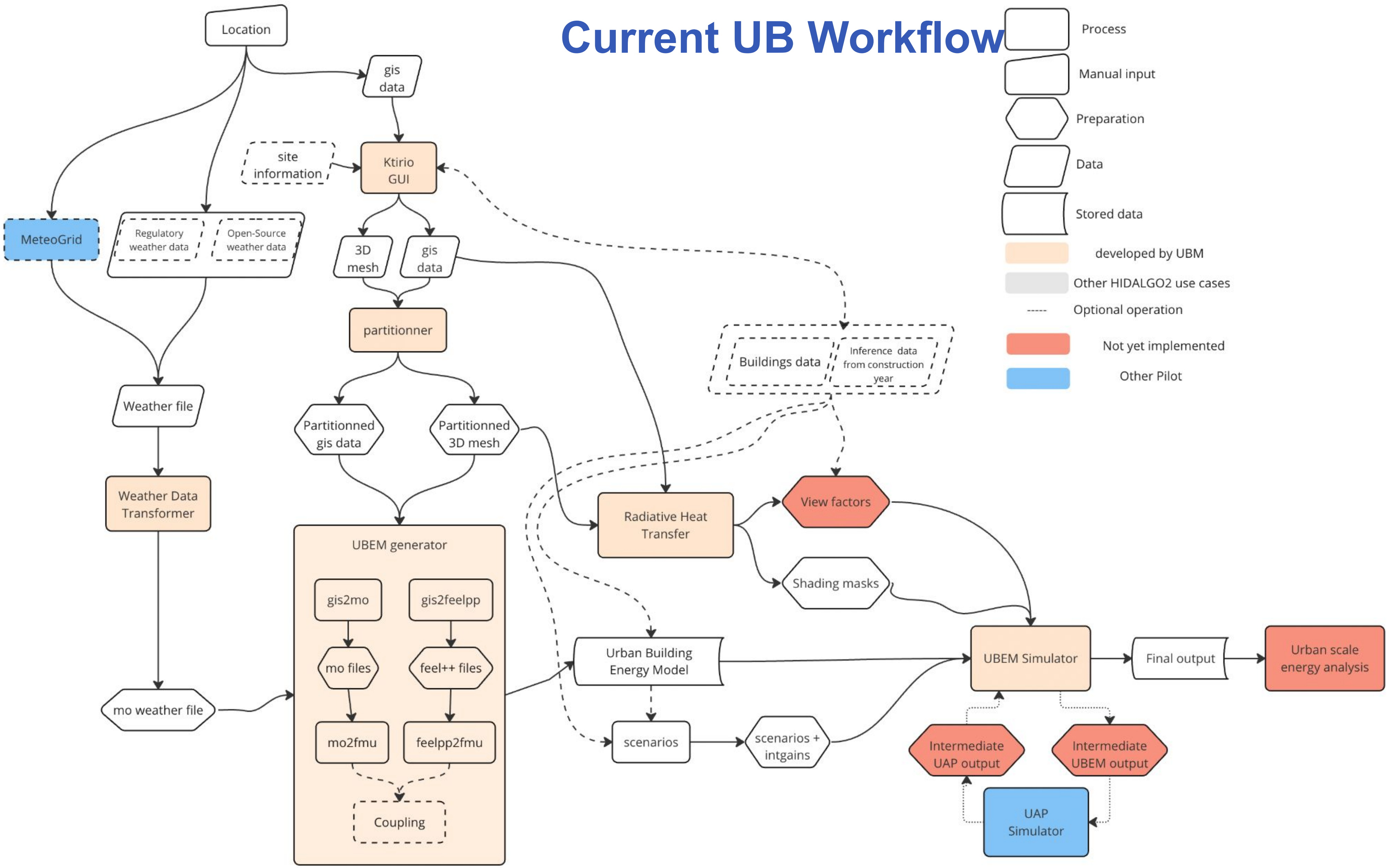
- Predict energy consumption, thermal comfort and indoor air quality at
  - Building scale
  - Urban scale and beyond
- Integrate the building stock in its environment:
  - Couple with Urban Air Pollution (UAP) model
    - ➜ contribution of the building stock (heat and GHG, NOx) to the outdoor air quality model (UAP)
    - ➜ improved boundary conditions of the building model (wind speed, outdoor temperature)
  - Improved radiative heat transfer on buildings' envelope, through a better estimation of solar shading

Objectives



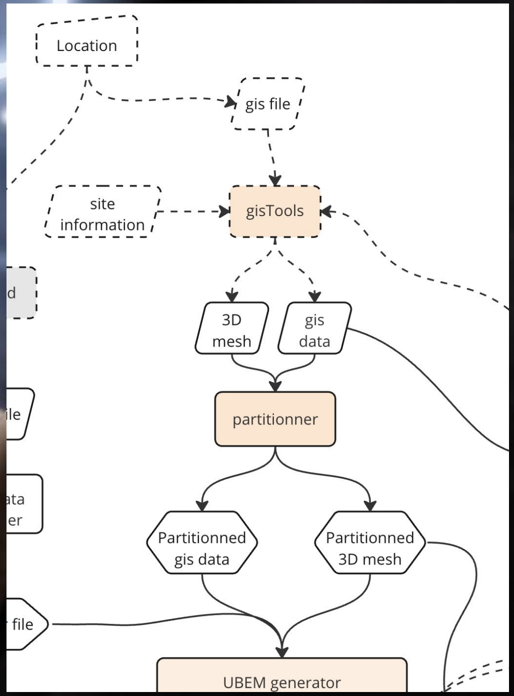
BIM = Building information Modelling
BES = Building Energy Simulation
IAQ = Indoor Air Quality
UAP = Urban Air Pollution
OAQ = Outdoor Air Quality
GHG = Greenhouse Gas
Weather database
Weather data
BIM
City councils
Other
OpenStreetMap
Material
Geometry
Shading
Building model generator
Urban scale building energy and IAQ models
Detailed building energy and IAQ model
Urban air quality model
BES and IAQ engine
UAP engine
Energy/IEQ indicators
Temperature/heat fluxes
GHG emission
OAQ indicators
Outside temperature
Wind (air) speed and direction

Current UB Workflow

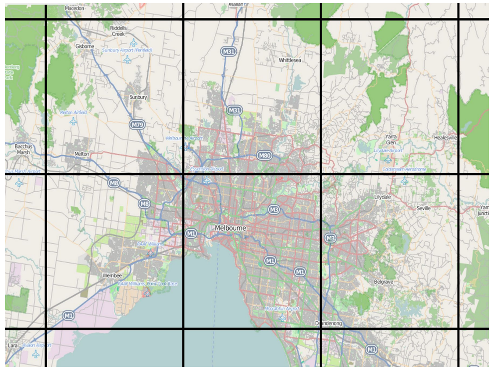


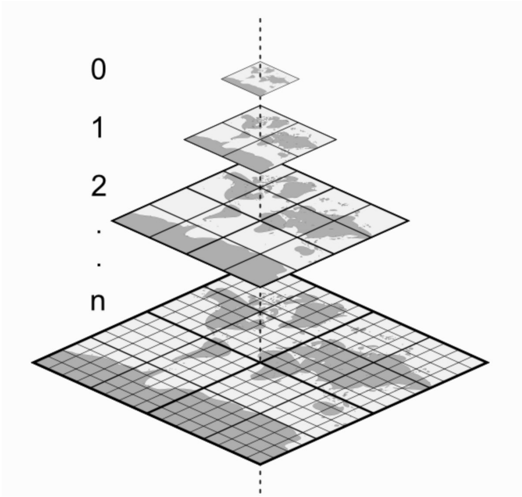
HIDALGO2
CENTRE OF EXCELLENCE

Geometry & Mesh & Partitioning



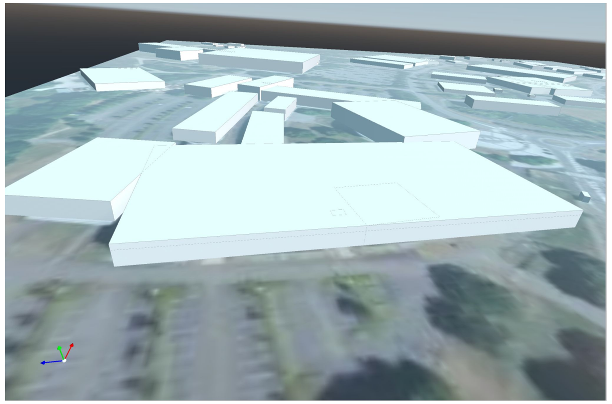
Geometry

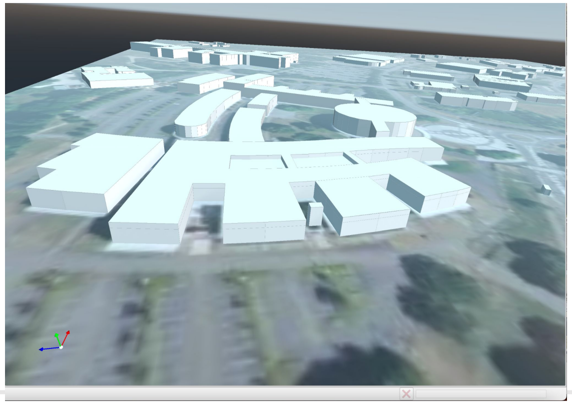


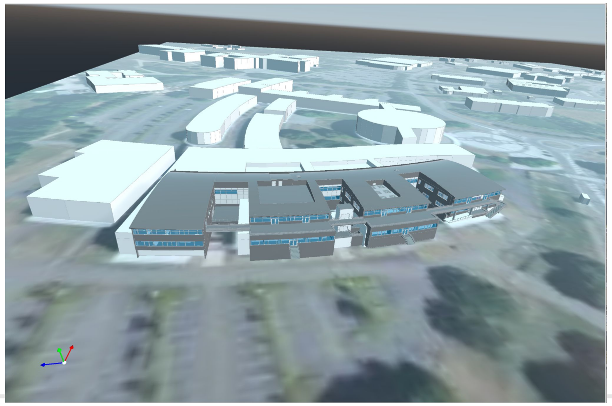


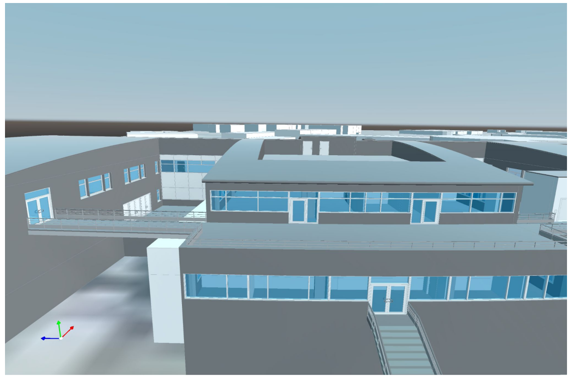
Geometric reconstruction of urban model

- **Challenge** : Create multi fidelity representation of buildings, terrain, vegetation, roads, …
- Tiled web map approach
  - Distributed data, adapt Level Of Definition (LOD)
  - Constraint : require gluing the tiles
- Buildings LOD
  - LOD-0 : Oriented bounding box
  - LOD-1 : Polygonal extrusion (+roof)
  - LOD-2 : IFC
- Terrain
  - Elevation from raster images
  - Include roads, parks, railways, rivers, …
- Vegetation (trees) - urban furniture
  - Limited information : position, height
  - Use prototypic geometry and parametrize





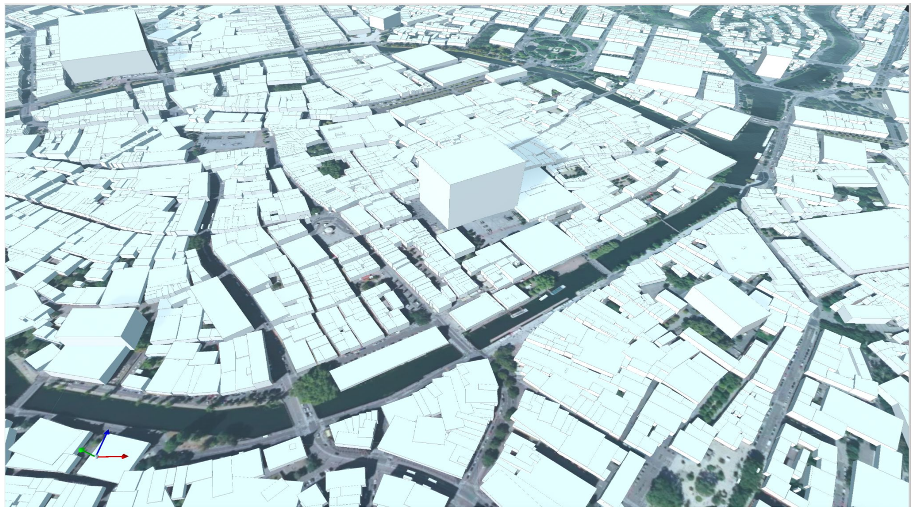




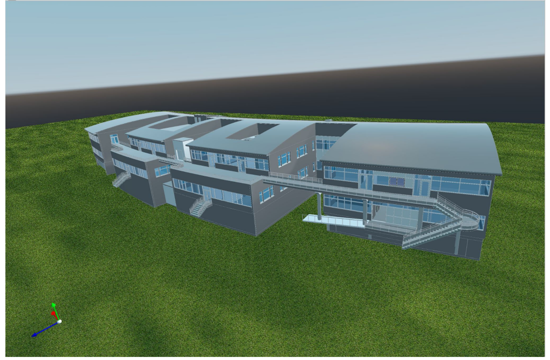
Mesh

Building meshes (LOD-0, LOD-1)

- Metadata fetched from the web (OpenStreetMap)
  - Multi polygons with holes
- Building LOD-0
  - Compute 2D oriented bounding box from building footprint
  - Extrusion applied with max height
- Building LOD-1
  - Composed of one or several parts defined by a polygonal extrusion
  - Apply union of all parts : generate one or several volumes for a building
- District/city
  - Collection of buildings
  - Apply union operations with building touching, or even intersecting
  - Optimization : building grouping from bounding box

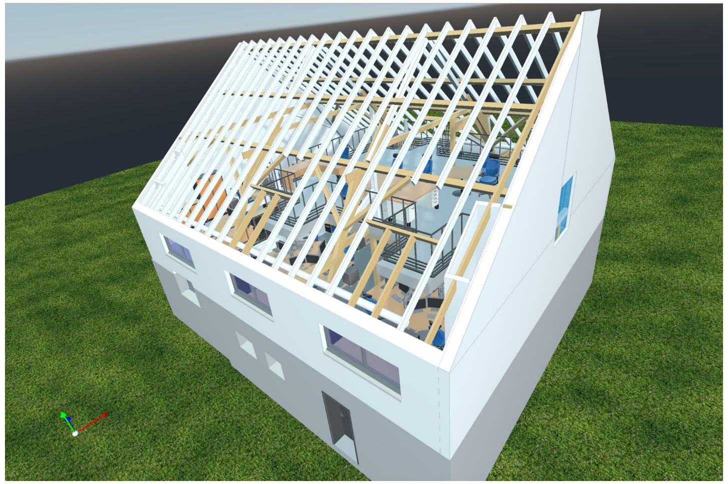


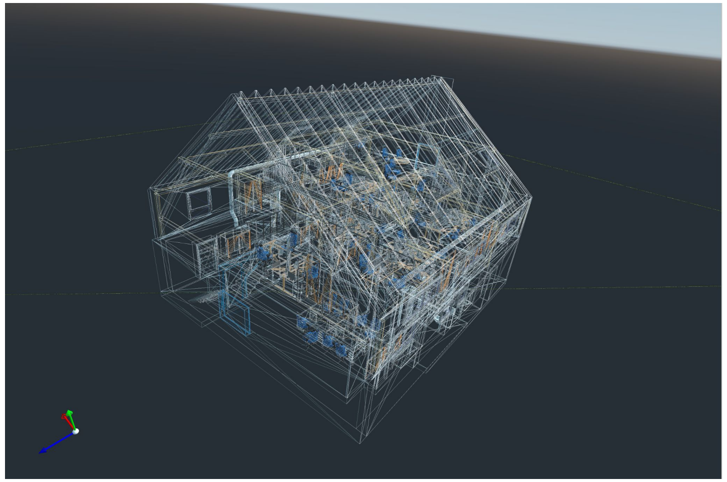
Mesh



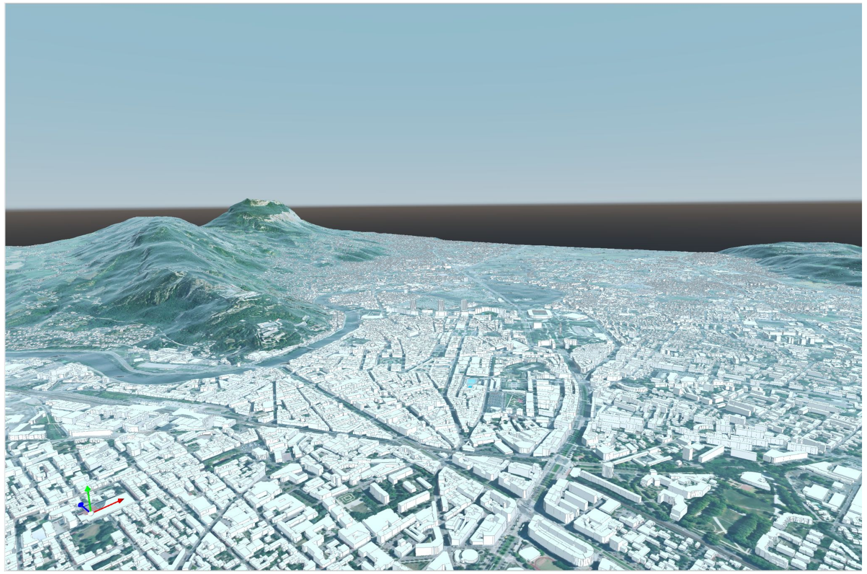
Building meshes (LOD-2)

- Geometry from Building Information Modeling (BIM)
  - IFC format
  - Surface model
  - Solid model : Swept, B-REP, CSG
- A geometry for each entity of building
  - Often non-conforming, not watertight
  - Allow to filter by entity type : structural, furniture, electric, plumbing, …
  - Mesh generator : Open-Cascade, Carve (IFC++), …
- **Challenge**
  - Conformal surface mesh
  - Volume mesh : watertight
  - Mesh Adaptation





Mesh

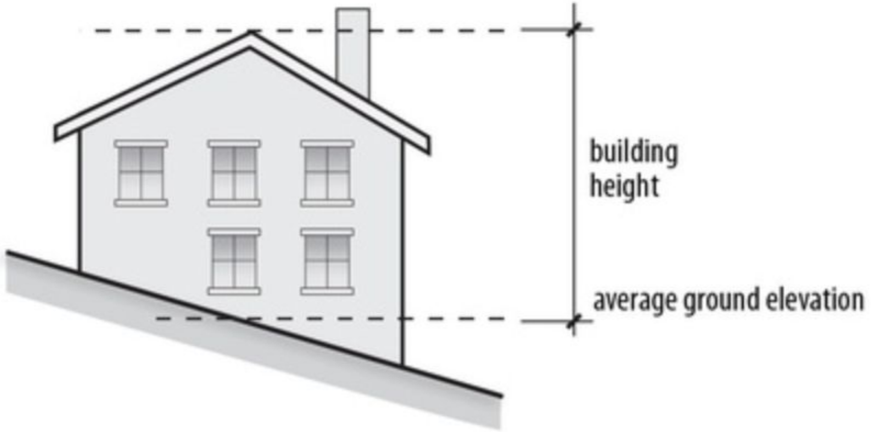


Terrain

1. Start by creating a uniform mesh (based on the raster image size)
2. Evaluate the elevation on each node
3. Compute isolines and mesh adaptation (reduce the number of element)
4. Glue each tile

Terrain + Buildings

- Conforming and watertight mesh
- Building (or any object) on a slope requires adapting the height
- Algorithm
  a. Compute difference of planar terrain and building footprint.
  b. Apply terrain elevation.
  c. Adapt building height.
  d. Insert building in the hole and repair mesh intersection.



Parallel Mesh Generation Strategy

Current strategy [MT]

1. From location, determine tiles used [Sequential]
2. Download GIS data (buildings, elevation) [Parallel MT]
3. Polygons repair [Parallel MT]
4. Terrain mesh generation [Parallel MT]
5. Union of building at tile junctions [Sequential]
6. Building mesh generation [Parallel MT]

Next Steps towards full parallel mesh generation [MPI+MT]

- Objectives : large city scale and beyond
- Use distributed computing (MPI)
- Create partitioning of tiles with overlapping (ensure fetching full building description)
- Create parallel mesh
  - Use MT strategy in each process
  - Overlapping zone allows creating full buildings without MPI Comm

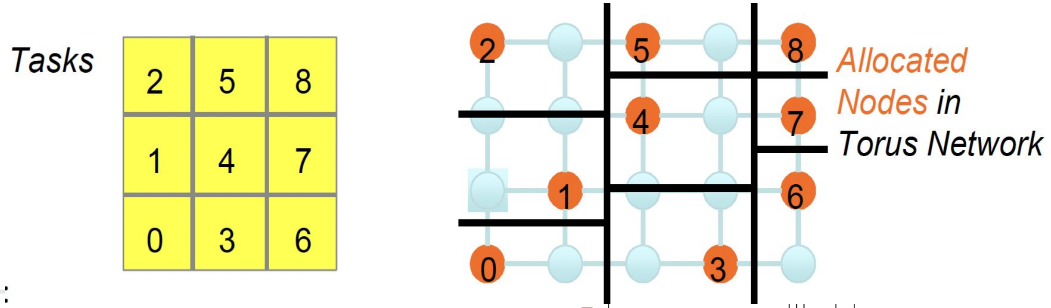
Mesh partitioning and simulation

- Input
  - Mesh of urban model (naturally partitioned thanks to tiled web map)
  - Number of partitions requested
  - Load balancing : weight parameter assigned to each building (depends on modeling complexity and solver)
- Output
  - Partitioned mesh used as input of urban building energy modeling

- Locality-aware partitioning
  - Building grouping
  - Interaction with the building's surroundings (solar shading, vegetation and river, …)
- Partitioning strategies :
  - Graph-based
    - Mesh connectivity
    - Toolkits : METIS, ParMETIS, SCOTCH, PT-SCOTCH, KaHyPar, Zoltan2, …
  - Geometric :
    - Uses coordinates of mesh entities
    - Toolkits : Zoltan2…
- Architecture-Aware Task Placement
  - Given a (possibly non-contiguous) allocation of nodes in a parallel computer, reduce application communication costs by assigning interdependent MPI tasks to “nearby” cores within the allocation
  - Important in extreme-scale systems

Zoltan2

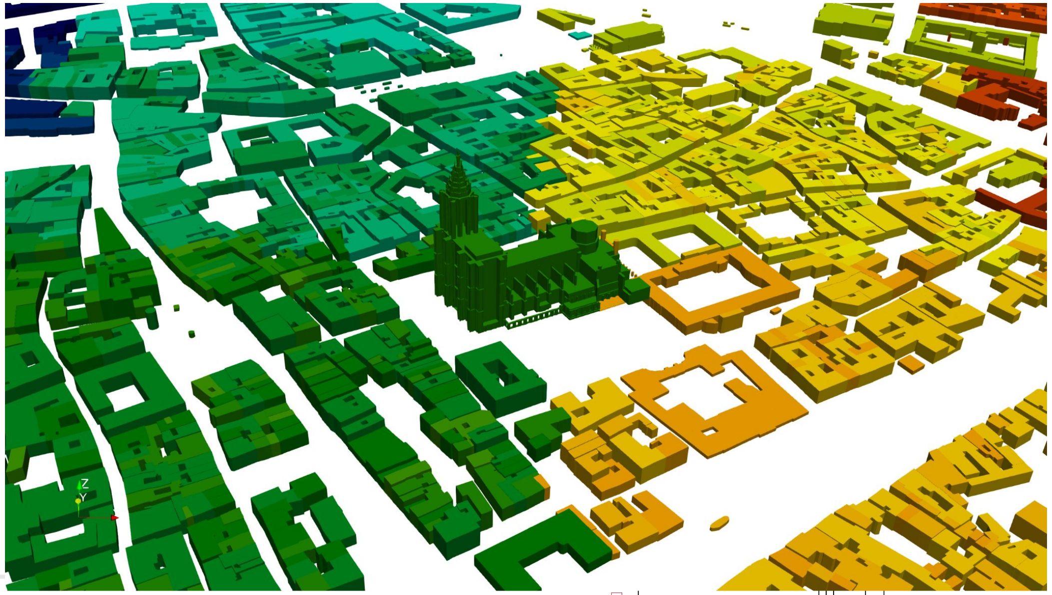
- Parallel partitioning : MPI+X (multicore,GPU)
- Geometric partitioning
  - MPI+OpenMP
  - Fast and scalable
- Graph partitioning via interfaces to PT-SCOTCH, ParMETIS, ….
- Architecture-Aware Task Placement



Mesh partitioning and simulation

Test case 0 : buildings mesh

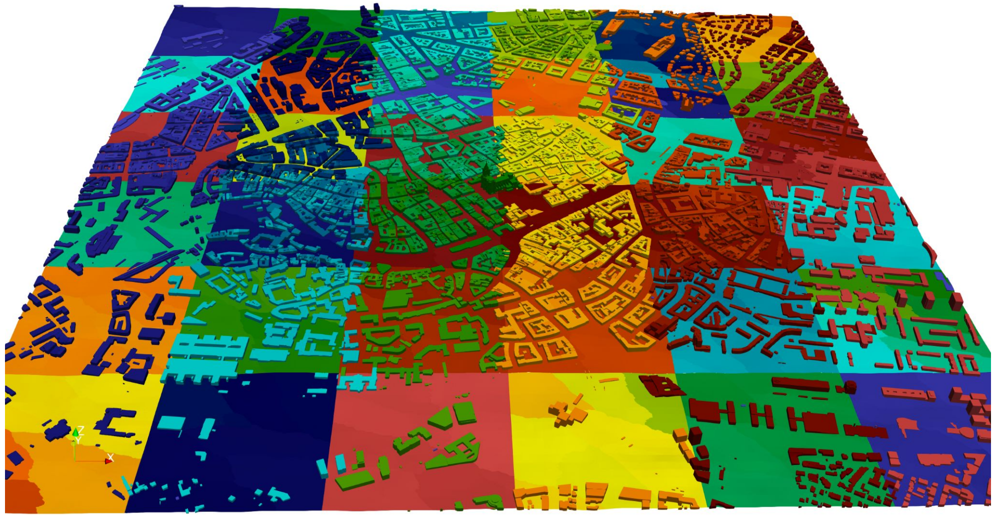
- Context
  - Building model has no interaction with environment
  - Only buildings mesh, no terrain, no vegetation …
- Partitioning methods
  - Use only the building listing (no spatial relation) and weights
  - Geometric partitioning
  - Weighted graph partitioning from a triangulation of building barycenter

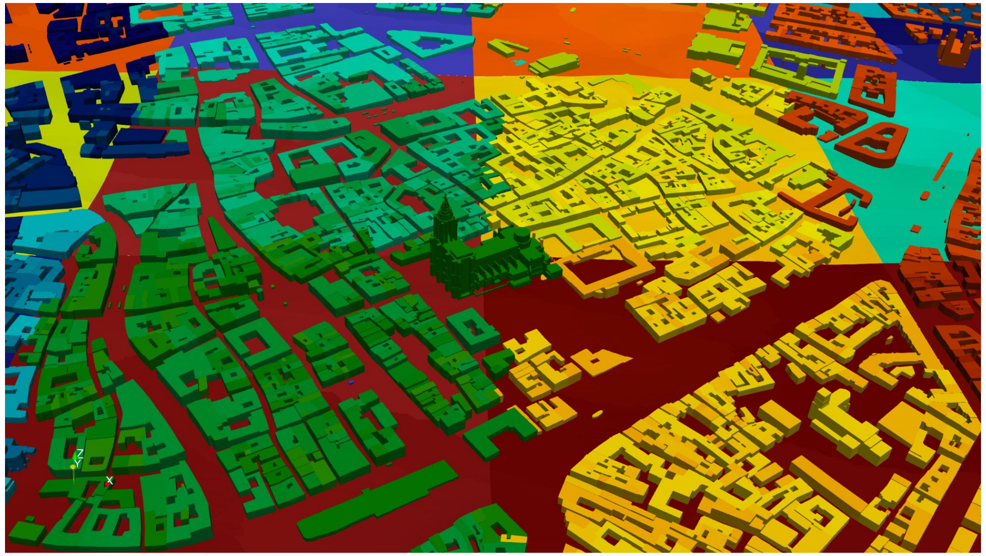


Mesh partitioning and simulation

Test case 1 : fully non-conforming mesh

- Context
  - Building model interact with environment and surrounding buildings
  - Building meshes are conforming and watertight
  - Terrain, vegetation mesh are non-conforming
- Partitioning methods
  - Geometric partitioning
  - Weighted graph partitioning
    - Buildings : Triangulation of building barycenter + Weighted + Building grouping
    - Terrain (and other structures) are partitioned independently and without constraint





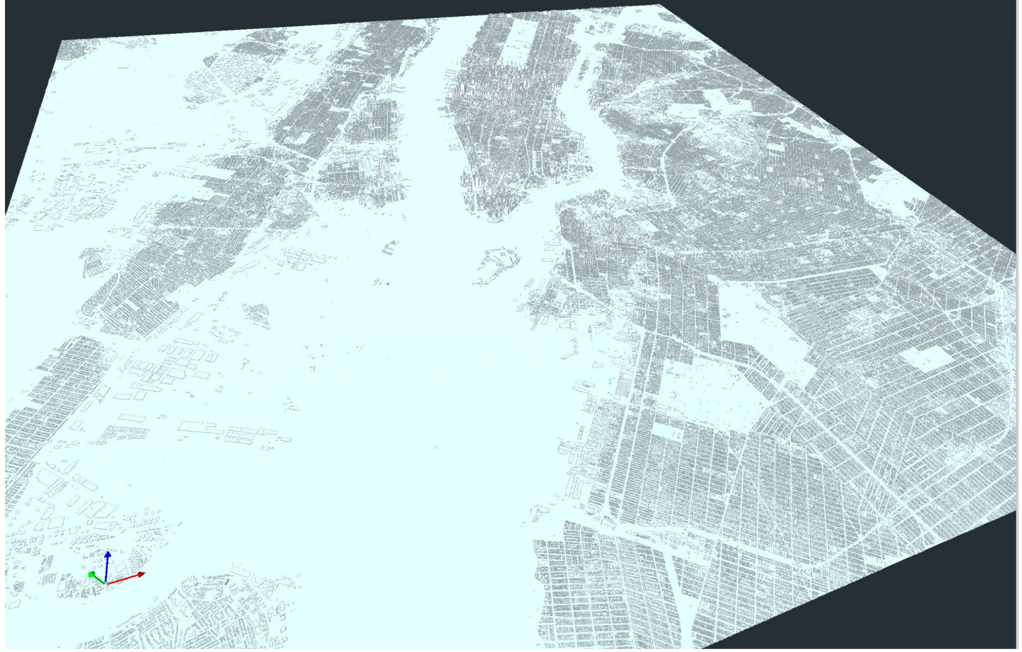
Mesh partitioning and simulation

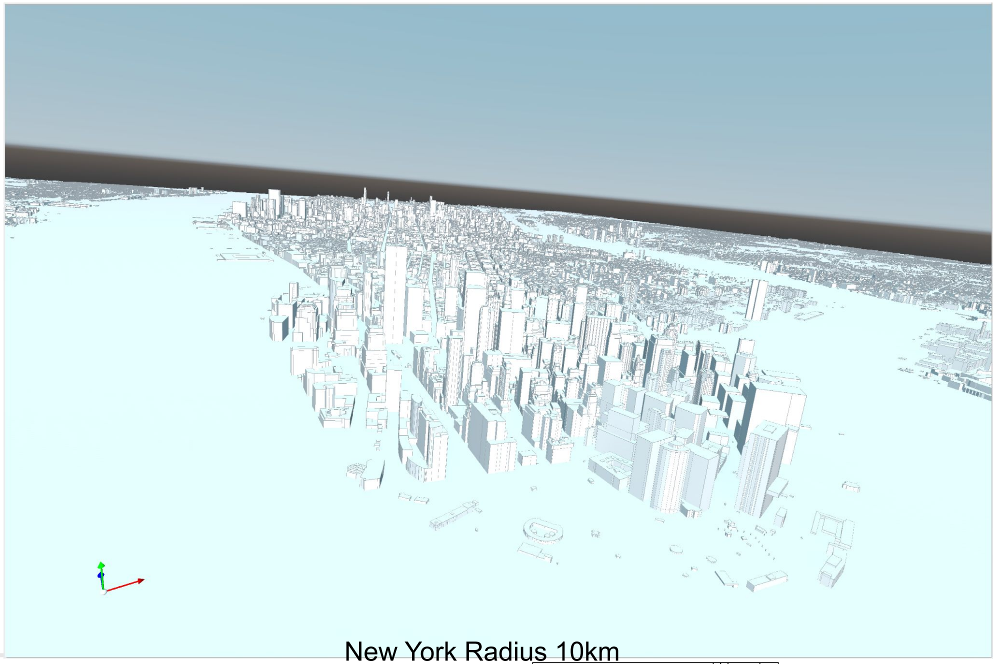
Test case 2 : full conforming mesh partitioning

- Context
  - Building model interact with the environment and surrounding buildings
  - Coupling pilots of Hidalgo 2 : UAP
- Weighted graph partitioning
  - Apply on full mesh
  - Start with buildings, then environment constraint with building partitioning
- Minimize communication cost compared to test case 1

Test case 3 : toward extreme number of partitions

- Multigrid approach
- Define coarse mesh (compute at GIS analysis phase)
  - District level with smart cutting (convex hull, …?)
  - Mesh as macro partitioning : each part (or grouping part) will define a subset of micro partitions
- Apply previous methodology in each macro partition

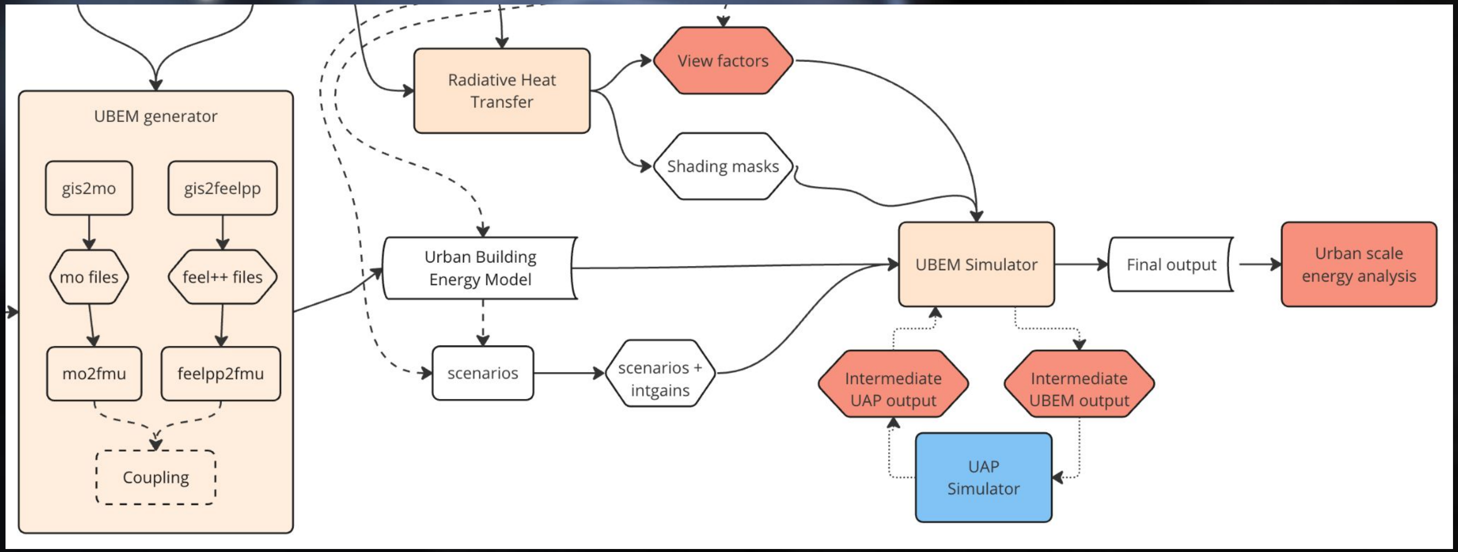




New York Radius 10km

HIDALGO2
CENTRE OF EXCELLENCE

Modeling & discretization



HIDALGO2 CENTRE OF EXCELLENCE

Simulation Tools

```
auto Vh = Pch<4>( mesh, markedelements(mesh, expr("<...>")) );
auto u = Vh->element(), v = Vh->element( g, "g" );
auto l = form1( _test = Vh );
l = integrate( _range = elements( support( Vh ) ),
               _expr = f * id( v ) );
l += integrate( _range = markedfaces( support( Vh ), "Robin" ), _expr = -r_2 * id( v ) );
l += integrate( _range = markedfaces( support( Vh ), "Neumann" ), _expr = -un * id( v ) );

auto a = form2( _trial = Vh, _test = Vh );
a = integrate( _range = elements( support( Vh ) ),
               _expr = inner( k * gradt( u ), grad( v ) ) );
a += integrate( _range = markedfaces( support( Vh ), "Robin" ), _expr = r_1*idt(u)*id(v));
a += on( _range = markedfaces( support(Vh), "Dirichlet" ), _rhs=l, _element=u, _expr = g );
a.solve( _rhs = l, _solution = u );
```

Feel++

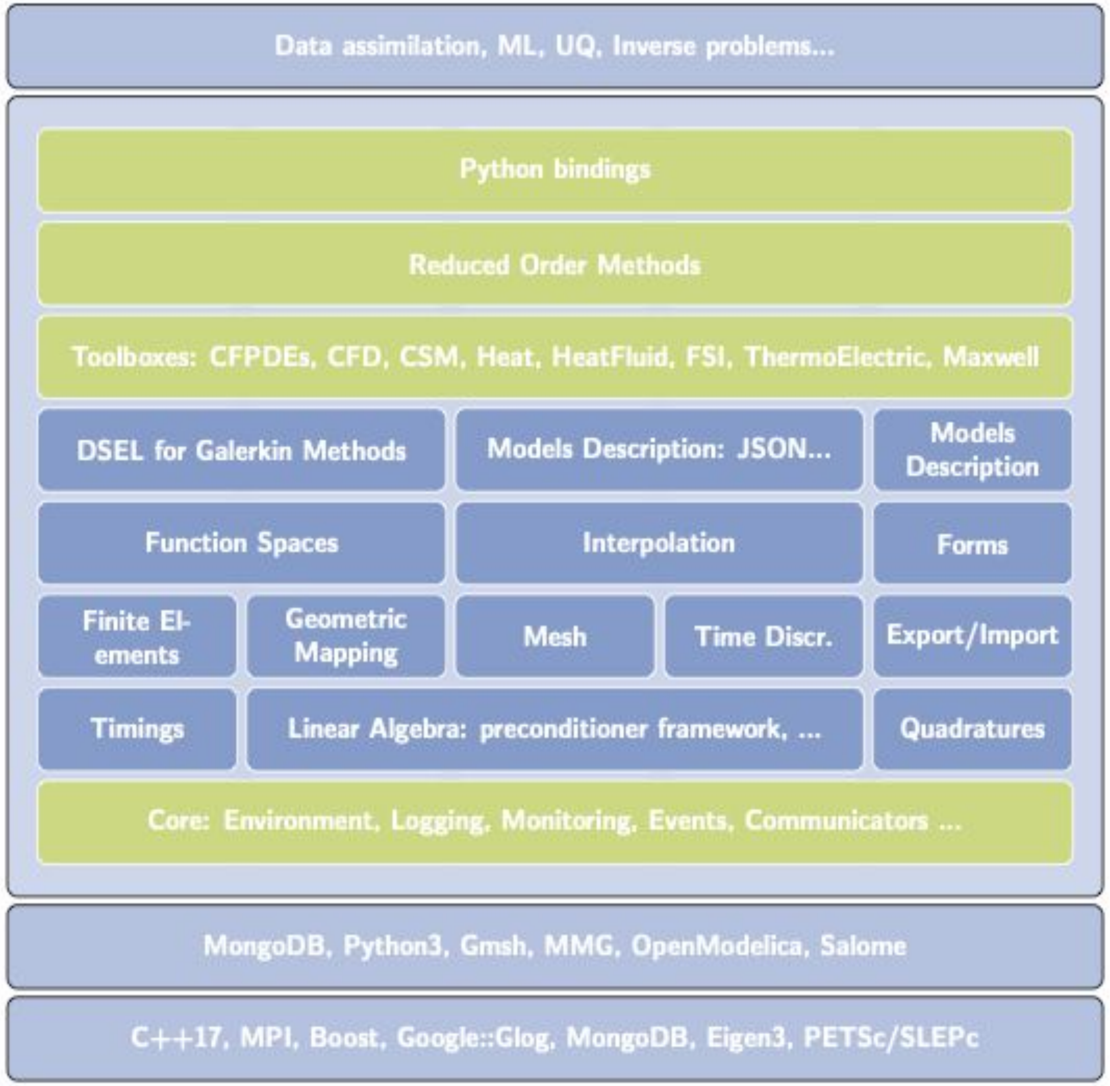
- Framework to solve problems based on ODE and PDE
- C++17 (C++20) with a Python layer using Pybind11
- Seamless parallelism with default communicator
- DevOps: CI/CD (docker/apptainer + ubuntu/debian packaging soon spack or guix-hpc)
- Tests: Hundreds of tests run in sequential and parallel C++ and Python
- Usage: Research, R&D, Teaching, Cloud Services

Simulation Tools

Feel++ : some features

- A large range of numerical methods to solve PDEs: cG, dG, hdG, rb/mor
- 0D+t, 1D(+t), 2D(+t), 3D(+t)
- DSEL for Galerkin methods in C++
- De Rham complex : arbitrary order
- Cross the bridge between symbolic and numeric computing
- Automatic differentiation
- Seamless interpolation
- WIP : Use Specx Task-based runtime system > Github

```
auto u = Xh->;element();
// expression handling
auto e3 = expr( "2*x*u*v:x:u:v" );
auto e3v = expr( e3, symbolExpr( "u", dxv( u ) ), symbolExpr( "v", Nx() ) );
// remeshing in seq and //
auto rm = remesh(_mesh=mesh,_params=<remeshing specs in json>);
// Fast marching : compute distance to range
auto distToBoundary = distanceToRange( _space=Vh, _range=boundaryfaces( mesh ) );
auto distToBoundaryNarrowBand =
   distanceToRange( _space=Vh, _range=boundaryfaces( mesh ),
                    _max_distance=3.*mesh->hAverage() );
```

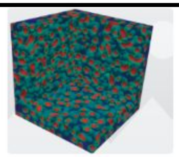


Simulation Tools : Feel++-Toolboxes

docs.feelpp.org/toolboxes/latest

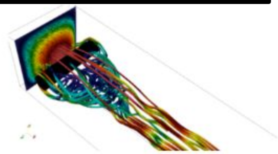
1. Core Manuals

The Toolbox Core Manuals describe (i) what is a toolbox (ii) how to setup simulation models thanks to JSON files and (iii) configure command line options of the toolboxes



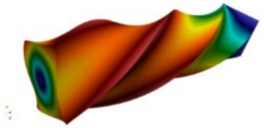
Coefficient Form PDEs

A coefficient form PDE toolbox to solve N coupled nonlinear PDEs



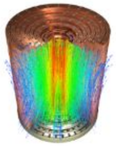
Computational Fluid Mechanics

The computation fluid mechanics solves from low to high Reynolds number flows and includes some turbulence models.



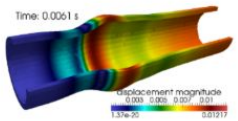
Computational Solid Mechanics

A toolbox to solve non-linear elasticity problems that supports (i) Large deformations, (ii) large displacements; (iv) Compressible, nearly incompressible materials and (v) Multi-material



Electric

A Toolbox that solves Gauss's Law equation and compute the electric potential and field of a given charge distribution.

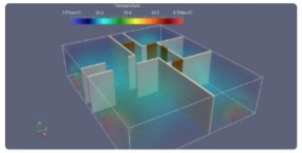


Fluid Structure Interaction

A toolbox for solving fluid-structure interaction problems using ALE formulation and staggered schemes such as Diriexlet-Neumann, Generalized Robin-Neumann, and Robin-Robin.

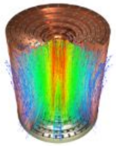
Heat and Fluid

A toolbox that solves the heat and fluid equations including convection, conduction effects. it is mostly used for aerothermal problems.



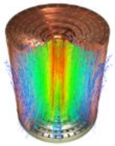
Heat Transfer

A toolbox to solve heat transfer problems including conduction, convection and black body radiation including non-linear problems and temperature dependent properties.



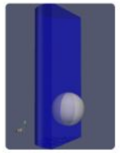
Hybridized Discontinuous Galerkin Toolbox

Hybridized Discontinuous Galerkin Toolbox case studies, supporting elliptic and parabolic problems in 2D and 3D including advection, diffusion and reaction, elasticity, Stokes and coupled problems.



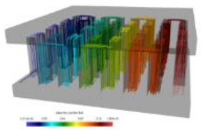
Maxwell

A toolbox for the simulation of electromagnetic fields in 3D including static, low frequency, and high frequency applications and non-linear materials.



Multifluid Flow Toolbox

A toolbox to solve multuid flow problems using the level set method



Thermo-Electric

A toolbox to solve the heat equation coupled to the electric currents equation including temperature-dependent material properties.

Masks and View factor

View factors

Describe the fraction of radiation leaving a surface and hitting a second surface

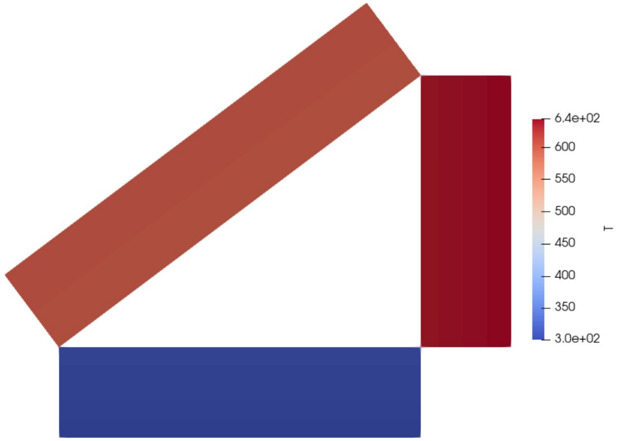
- Purely geometric quantities

Numerical method: Monte Carlo and ray tracing

- Geometries with obstructions
- Only gray surfaces for the moment, but specular could be considered

Challenges

- Efficient computation of in view factors
- Computation of view facto specular surfaces (severa bounces)
- Efficient storage



Solar masks

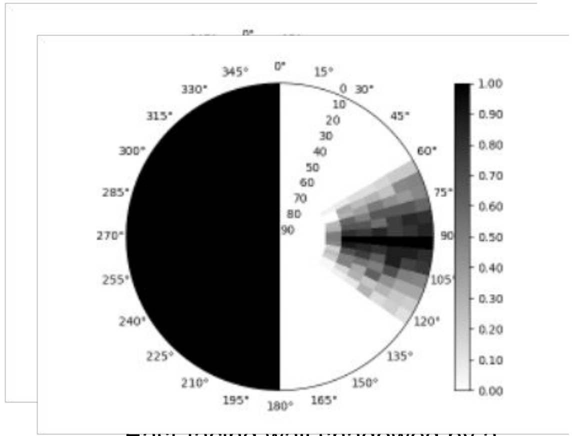
Describe the percentage of blocked solar radiation for each building surface (walls, roofs) and ray direction

- Caused by surrounding buildings, vegetation, urban furniture
- Can be computed per surface or per building

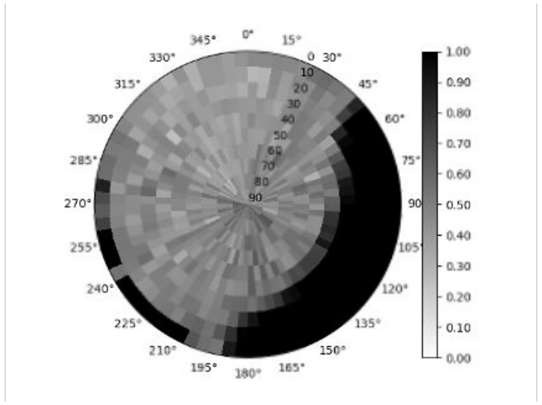
Numerical method: Monte Carlo and ray tracing

Challenges

- Computation on large-scale mesh and storage



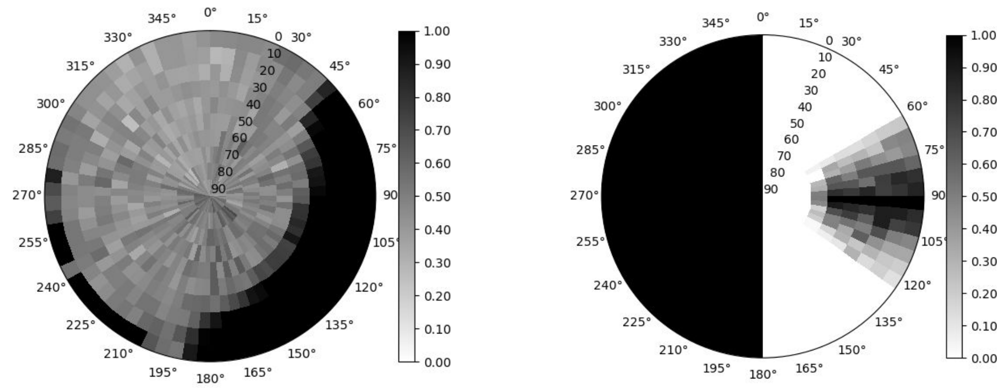
East-facing wall shadowed by a tall building in front of it

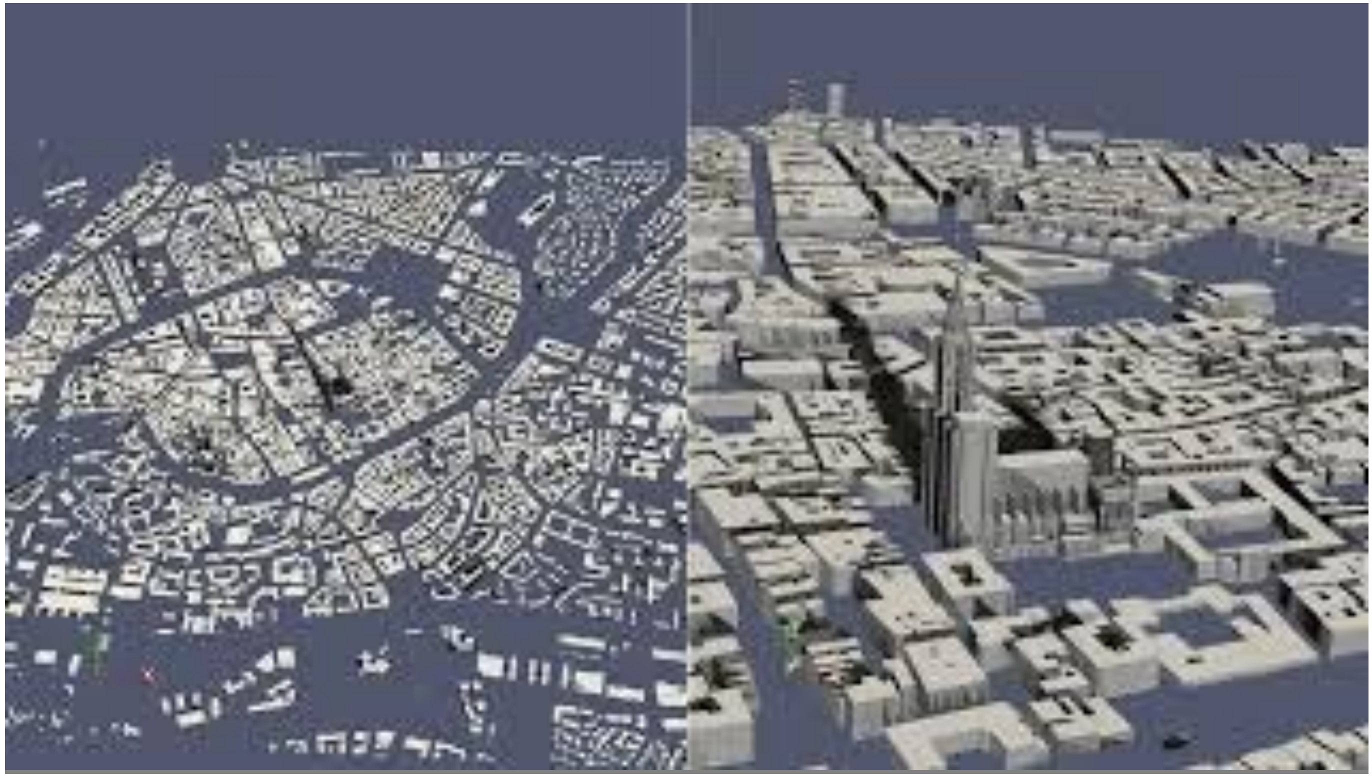


Shading mask for a whole building

Solar Masks computation

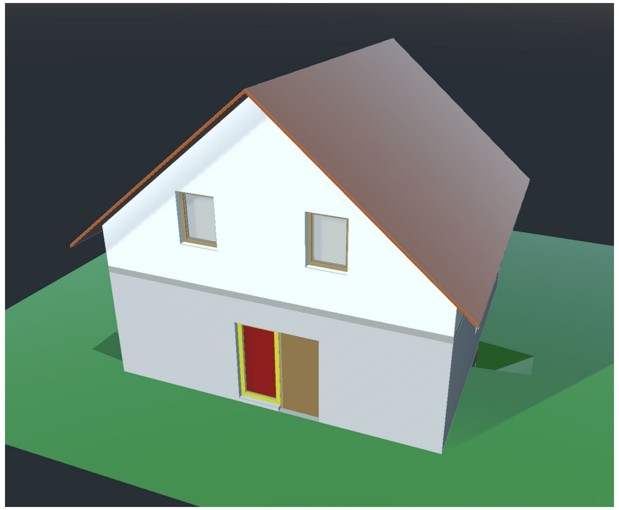
- Environments of buildings is critical and in particular exposition to the sun
- For each face of a building, compute solar masks using a MC approach for each position of the sun
- Very fast parallel computation of solar masks on CPU

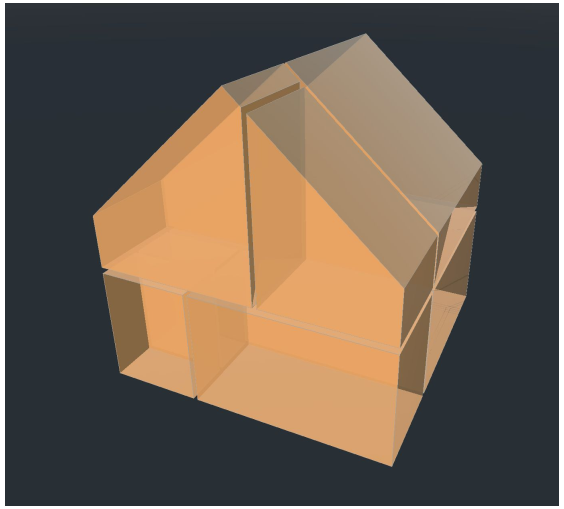


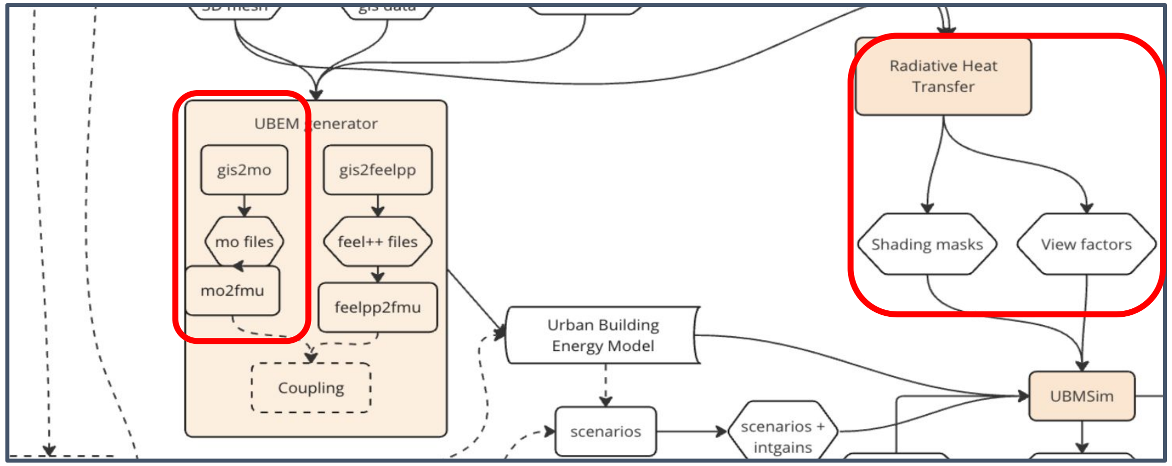


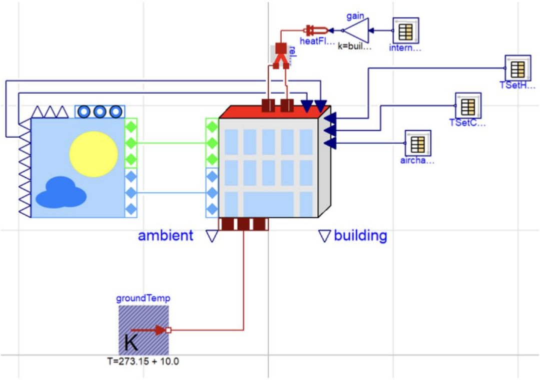
Multizone modeling using Modelica

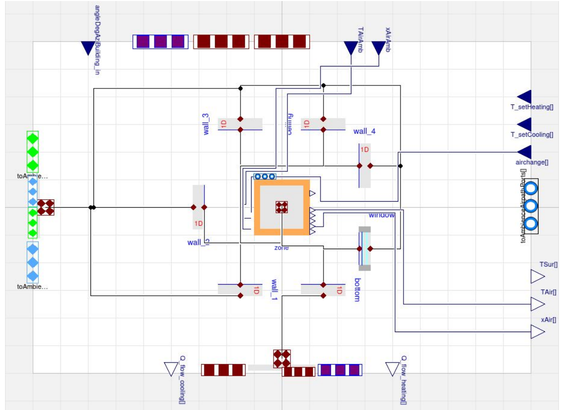
- Multizone model for building energy simulations
- Use Modelica language with extensive support for building energy modeling
- Multifidelity models: LOD-0, LOD-1
- Generate FMU (Mo transformed into C/C++)
- Sequential (some MT support)











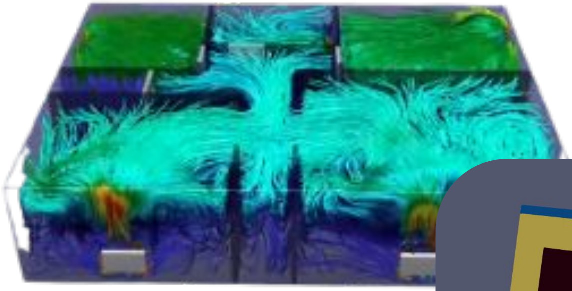
Finite Element modeling/ROM using Feel++

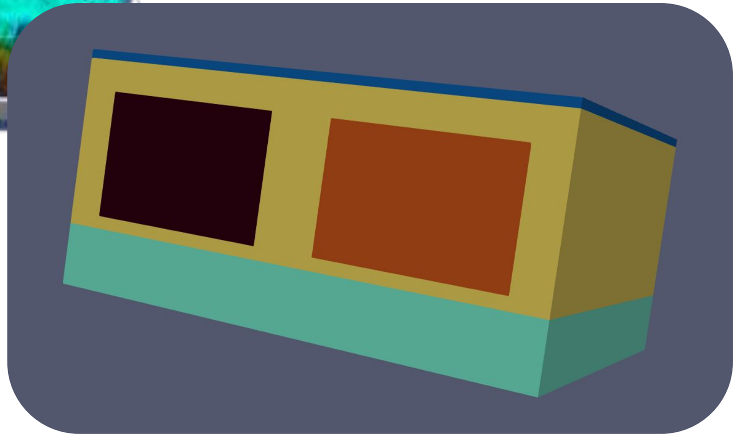
Predict energy consumption, thermal comfort and internal air quality
- Detailed description of internal air temperature
- Fast computations and scenario evaluation

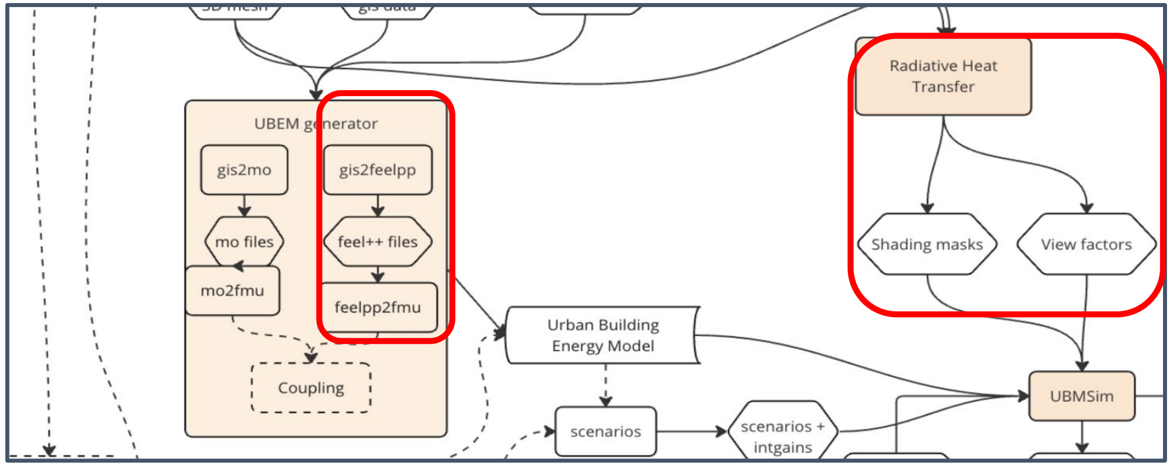
Models
- Steady state building model
- Unsteady model accounting for multi-layer walls and window openings
- Solar radiation and shading from the surrounding environment

Numerical methods
- Finite element and reduced basis methods
- Data assimilation for parameter estimation
- Parallel-in-time algorithms
- Model order reduction using eg RB

Challenges:
- Large number of parameters
- 3D-0D couplings







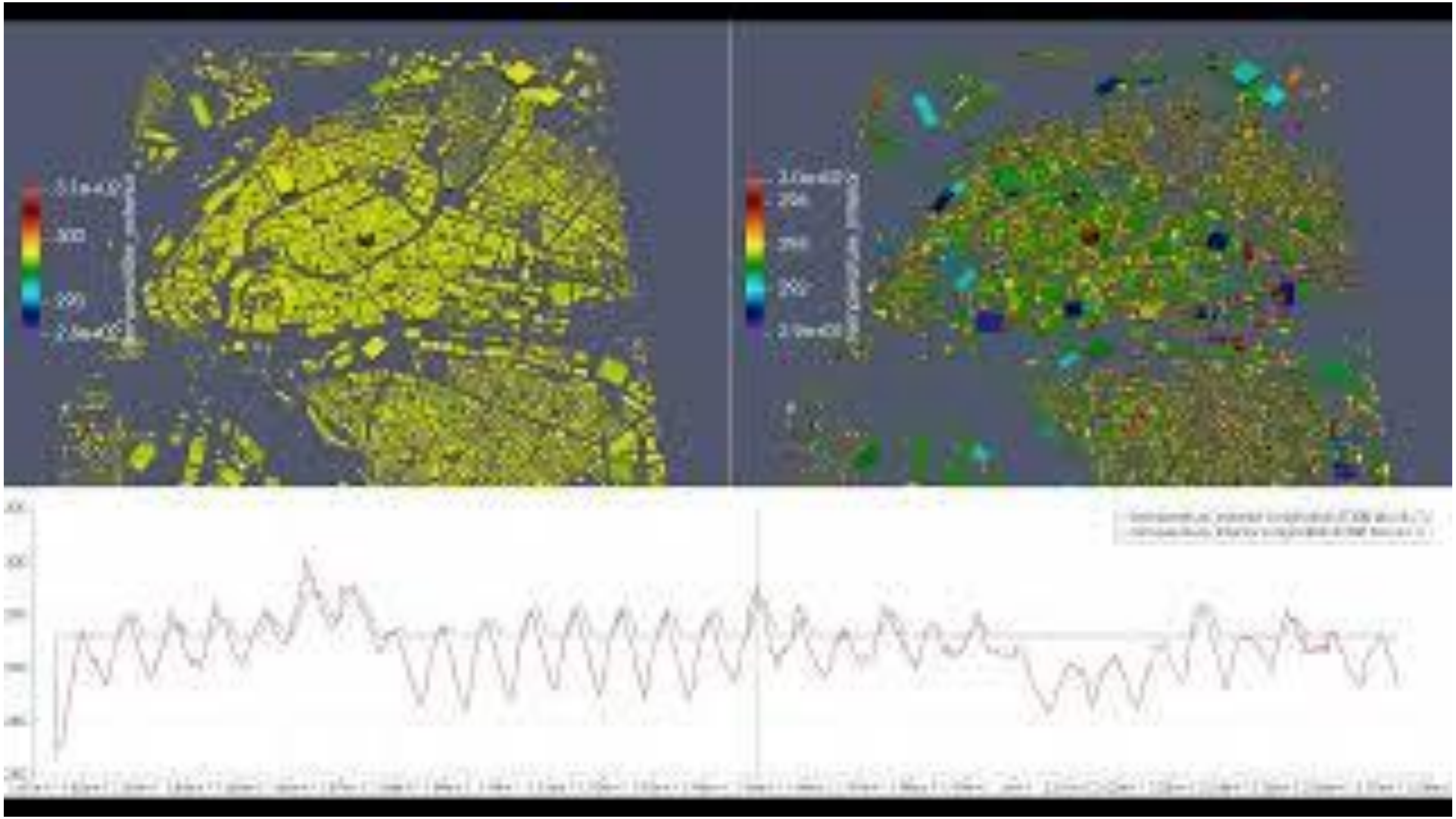
Challenges

Hybrid computing

- Strong link to Mesh Partitioning/Load balancing (see mesh section)
- Finite element method: use our DSEL to hide "details" (-> Mfem, Kokkos, Eigen, …)
- Reduced basis model (offline see above, online opencl implementation, large scale DB storage)
- Solar mask and view factors computing
  - Current CPU implementation very fast. testbed for specx (task-based runtime execution)
  - Well suited for GPU computing
  - Should we store the masks and view factors or recompute them in situ? (probably both options are relevant)
- Multizone models: currently a strong bottleneck as we do not control the generated code

UB in Strasbourg (more than 17000 buildings using LOD-0)

- 4x4 km square around the center of Strasbourg (about 17000 buildings) over 2 months
- We simulate the BEM using meteorology is from 2023
- We display the internal and external temperature starting in January.
- The simulation was executed on **EuroHPC JU Discoverer**
- **Software tools:** C++, FMU(building models), GitHub, Antora , Python



Post Processing I/O & Data Exchange

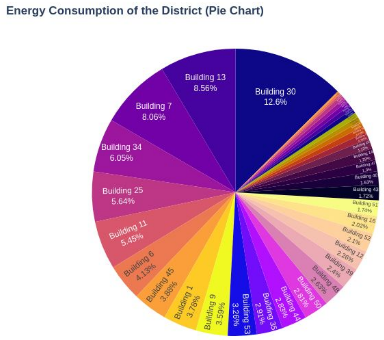
Post Processing

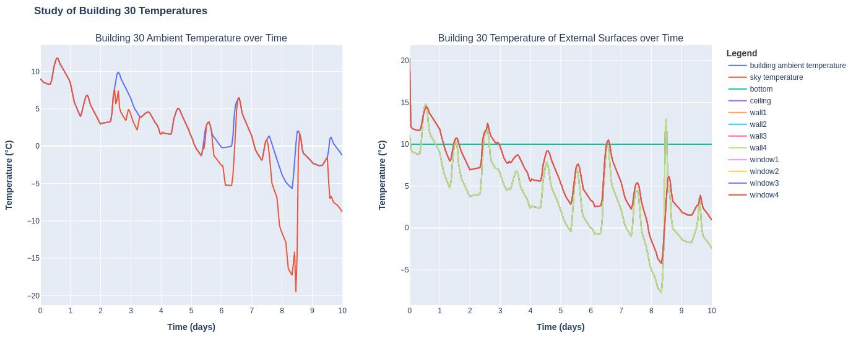
Save buildings simulation outputs

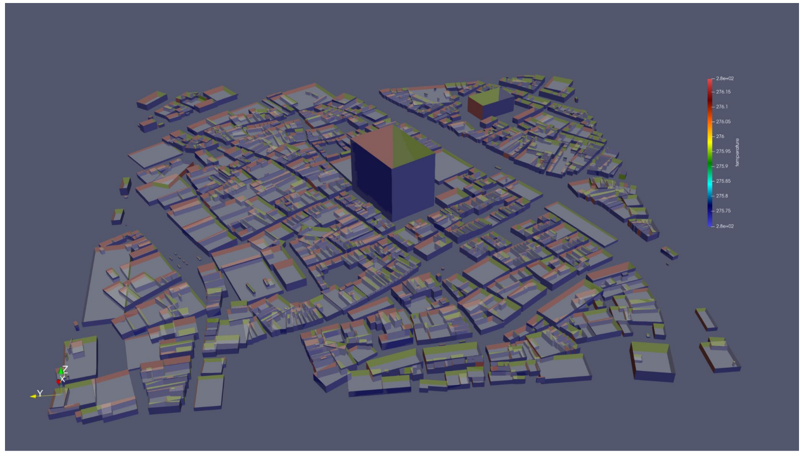
- Outputs : Temperature, Humidity, Energy, CO_2, NO_x, ...
- Frequency : at each time step (or over time points)
- Measures location :
  - On each building (global values)
  - On each building face
  - On each node/element of the (surface) mesh

Use cases

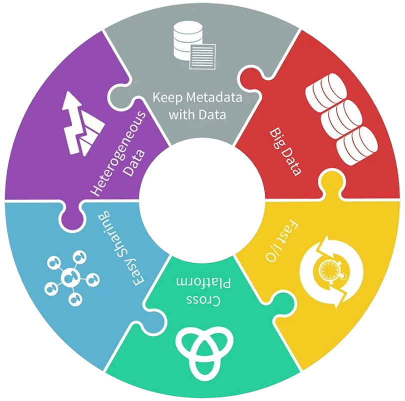
- Statistical analysis
  - Reporting generation, plot chart : line, bar, pie, …
  - Compute indicator : comfort, energy, …
- 3D visualization : ParaView, Ktirio, …
- Data exchange (coupling with another application)
- Restart simulation

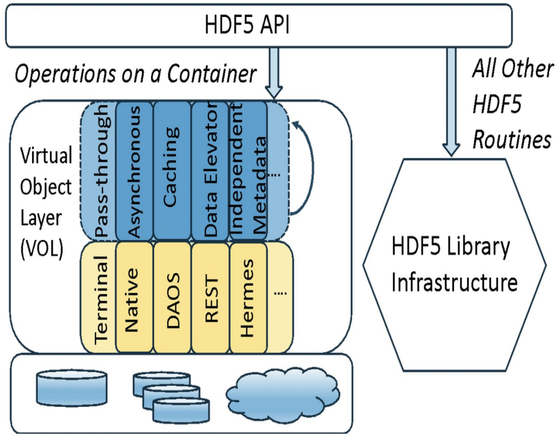






Post Processing : I/O



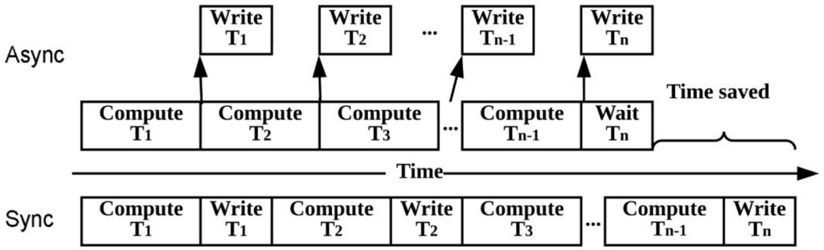


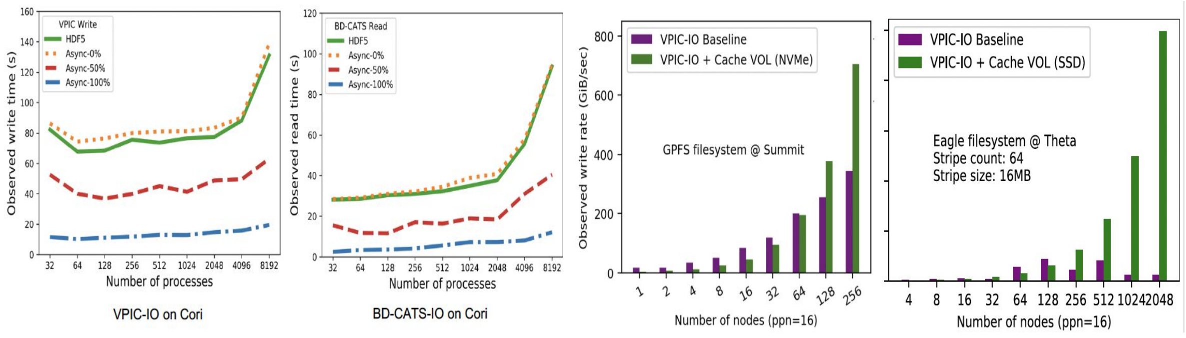
File formats

- CSV : Sequential text file, not really adapted to //
- JSON : Handle complex data, nice to describe metadata
- HDF5 : Parallel format with many features
- Ensight Gold : 3D Visualization format (binary and writing in // using mpi-IO)
- XDMF+HDF5 : 3D Visualization format

Scaling up

- IO is currently a bottleneck to scale up (many cores/nodes)
- One file per proc : generally not recommended
- Current solutions identified : HDF5, SIONlib, ADIOS
- Strategy envisaged
  - Asynchronous I/O [1] : background thread using task managers, hiding I/O latency
  - Caching with node-local memory and storage [2]
- HDF5 and Virtual Object Layer (VOL)
  - Same API, but allows different underlying storage mechanisms





Weak scaling

[1]: https://github.com/hpc-io/vol-async
[2]: https://github.com/hpc-io/vol-cache

Post Processing : Data Exchange

Multiphysics coupling : PDEs

- Coupling interface through mesh discretization
- Strong/weak coupling : one-way or two-way
- Our application : coupling Urban Building and Urban Air Pollution pilots

Requirements

- Data exchange between two (or more) simulation frameworks in HPC context.
- Communication peer-to-peer (objective is to reduce the use of file system)
- Unbalanced computational resource

Potential solution for coupling solution between UAP and UB

- MUSCLE3, CWIPI(Onera), MED-COUPLING(CEA,EDF)

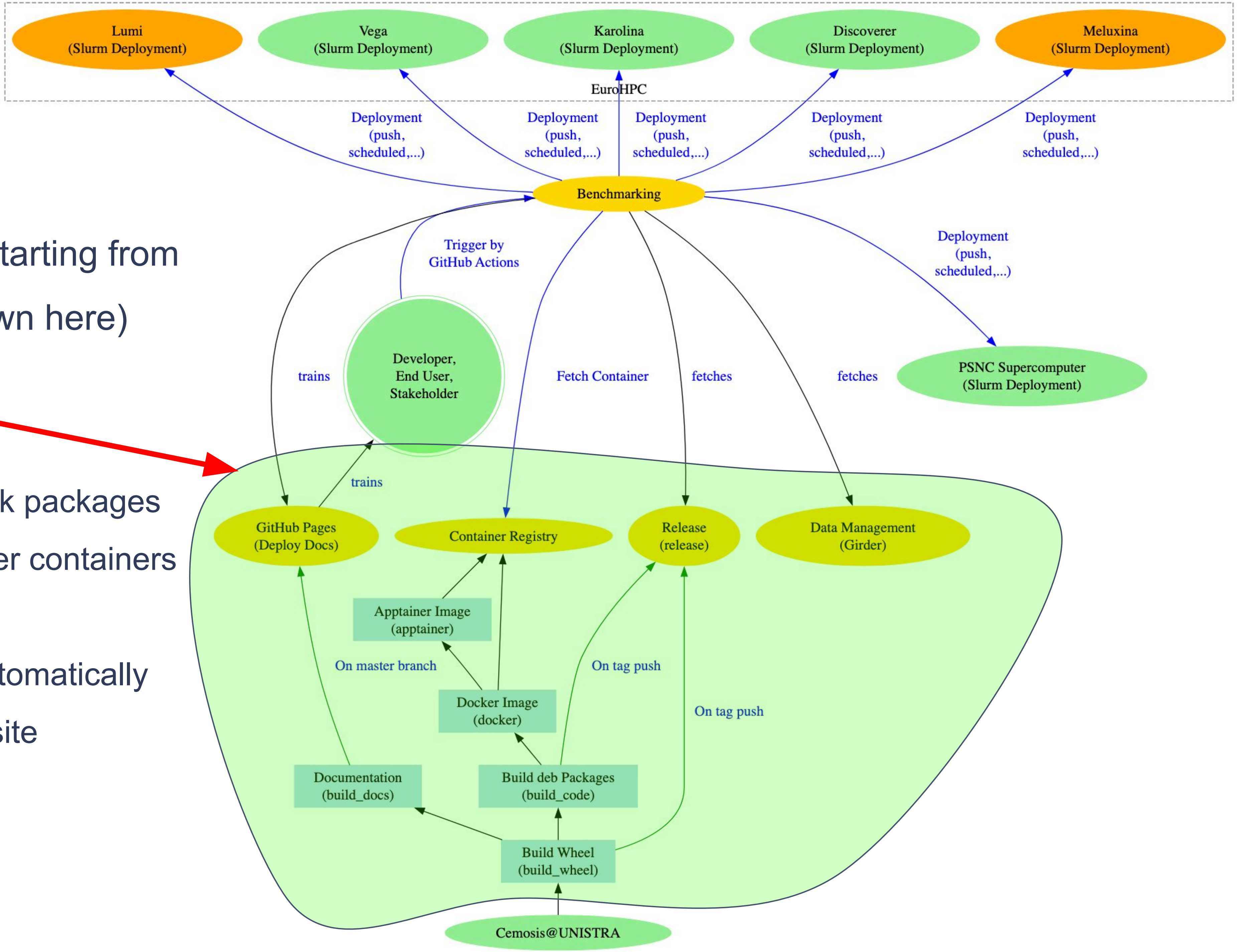
HIDALGO2

CENTRE OF EXCELLENCE

Post Processing I/O & Data Exchange

UB CI/CD

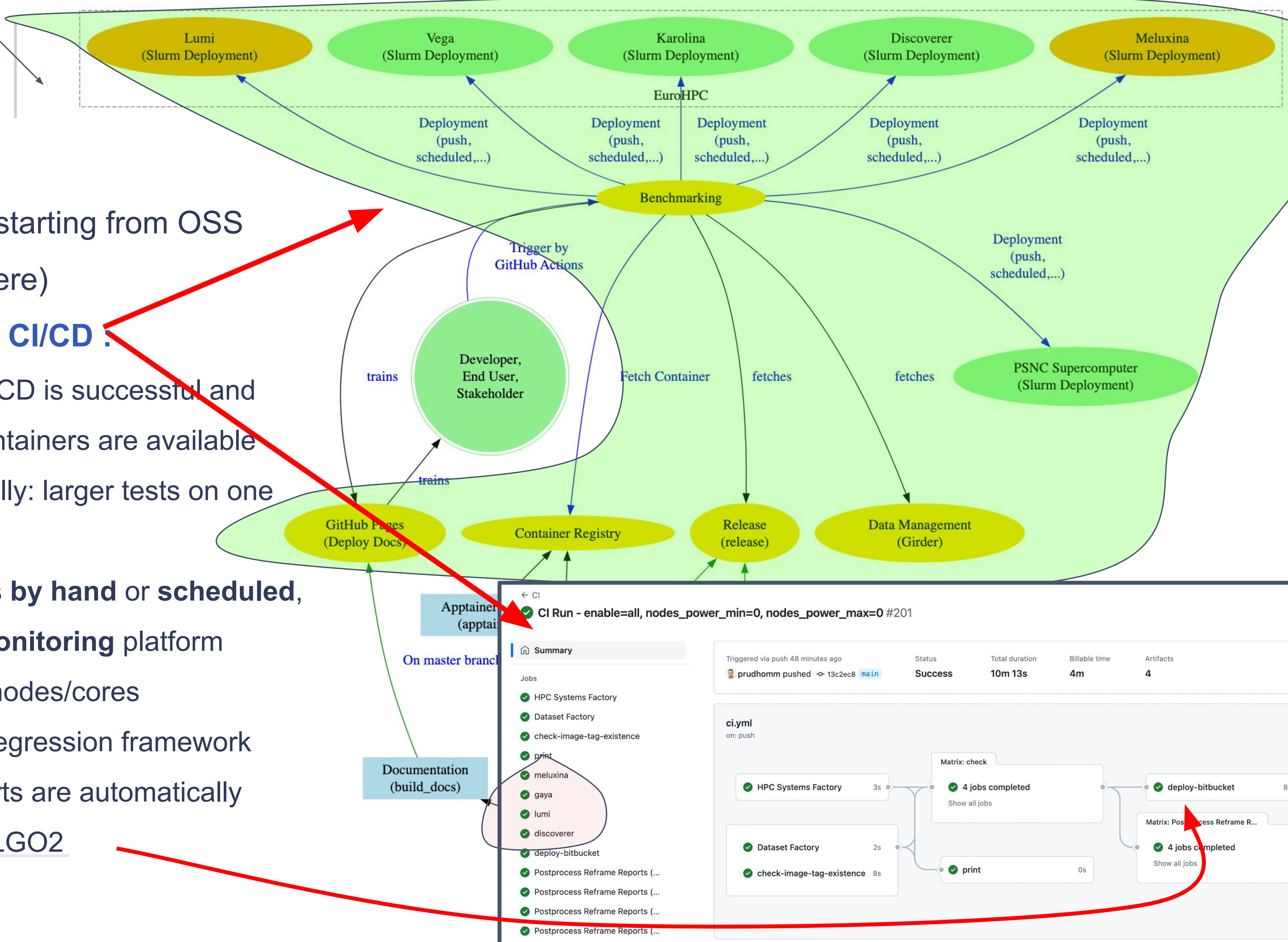
- A complex toolchain starting from OSS Feel++ (not shown here)
- **Standard CI/CD :**
  - **Packaging** : wheel, debian/ubuntu/spack packages and docker/apptainer containers
  - **Releases**
  - **Documentation** automatically deployed on a website



EuroHPC Joint Undertaking

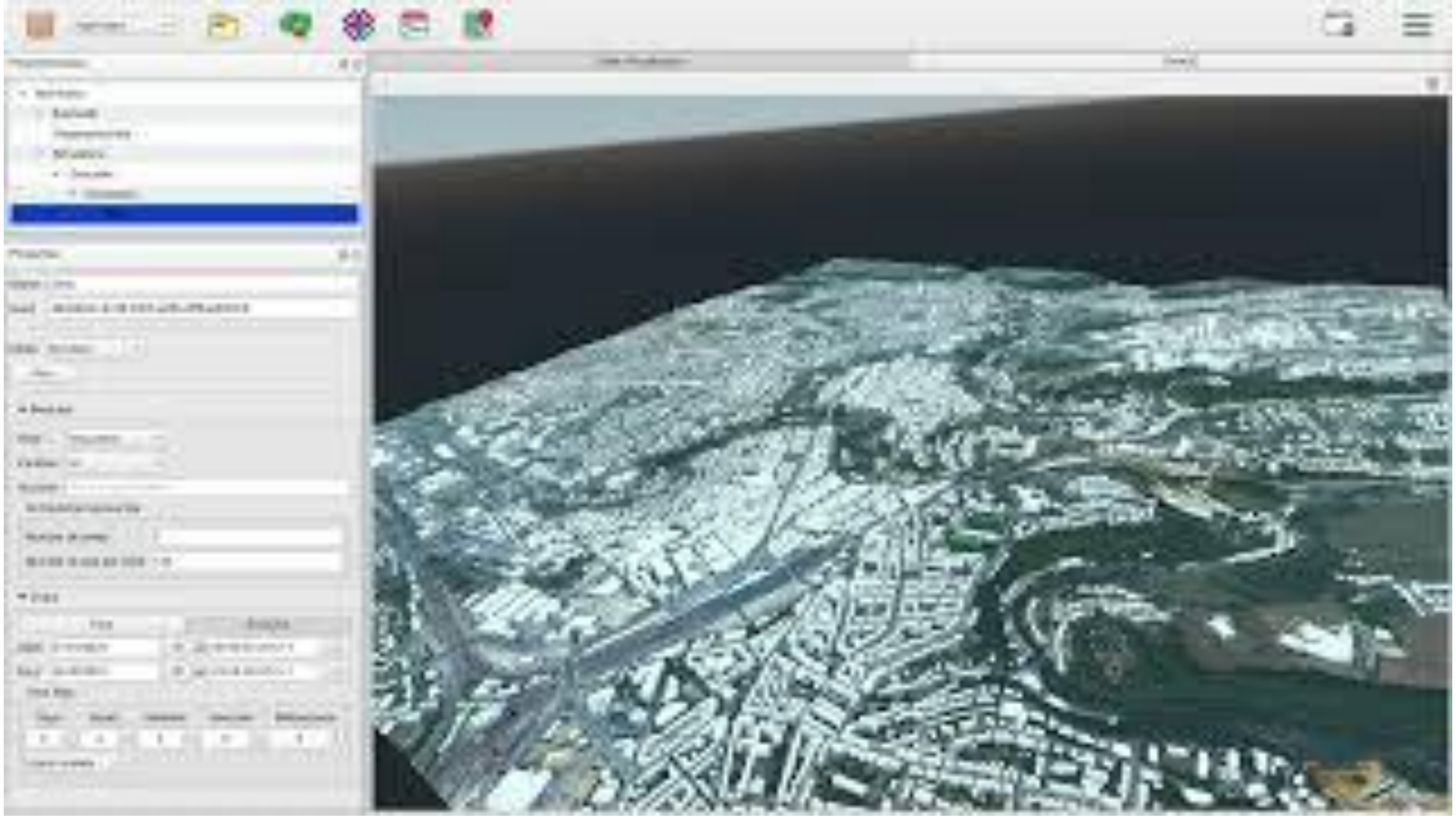
UB CI/CD

- A complex toolchain starting from OSS Feel++ (not shown here)
- **HPC Benchmarking CI/CD :**
  - If the Standard CI/CD is successful and new **apptainer** containers are available
  - Trigger automatically: larger tests on one node (128 cores)
  - Trigger larger tests **by hand** or **scheduled**, store results for **monitoring** platform possibly on more nodes/cores
  - Use **Reframe** as regression framework
  - Performance reports are automatically uploaded to HIDALGO2



UBEM CI/CD for HPC

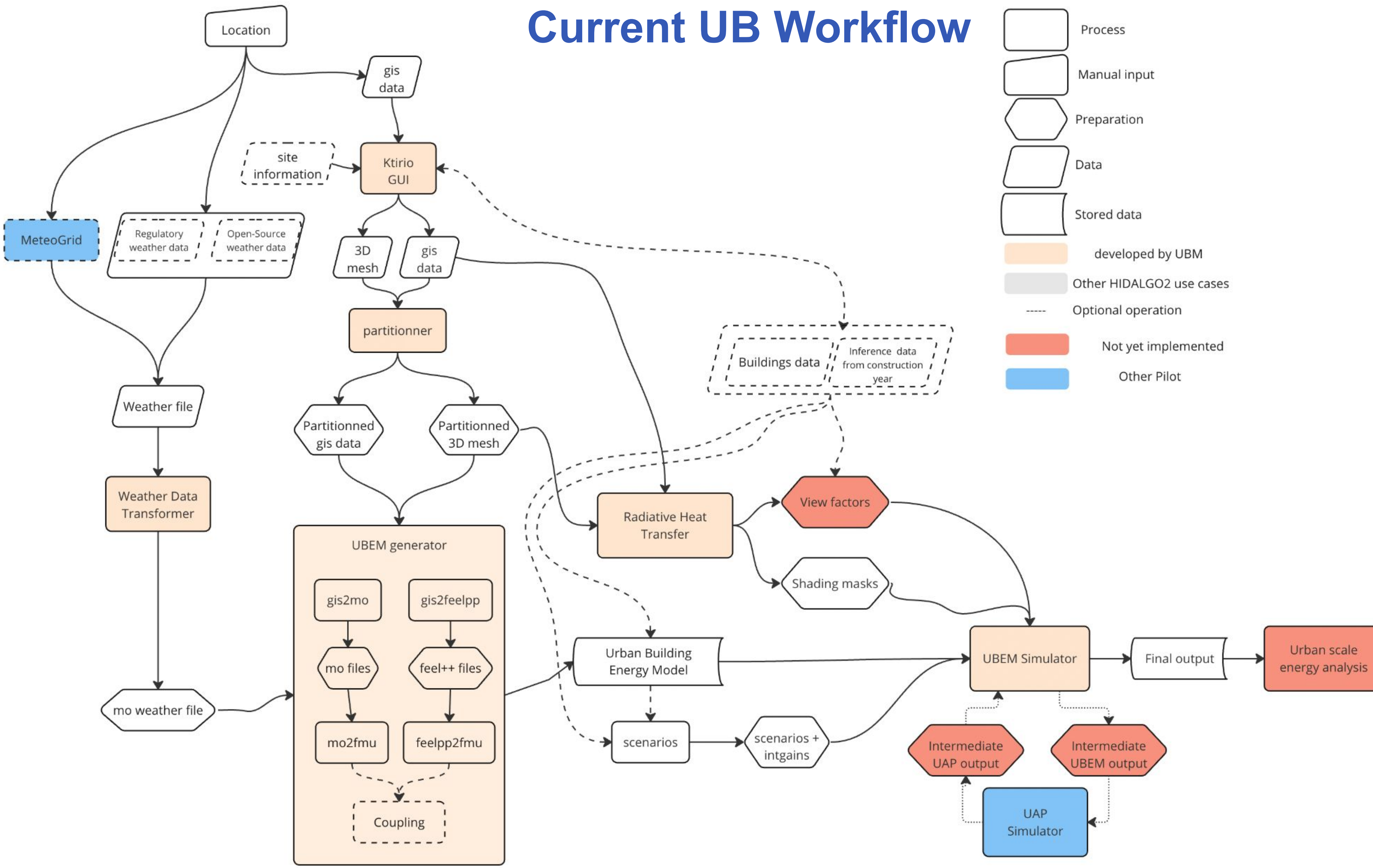
City of Luxembourg (4km square)



HIDALGO2 CENTRE OF EXCELLENCE

Conclusion

Current UB Workflow



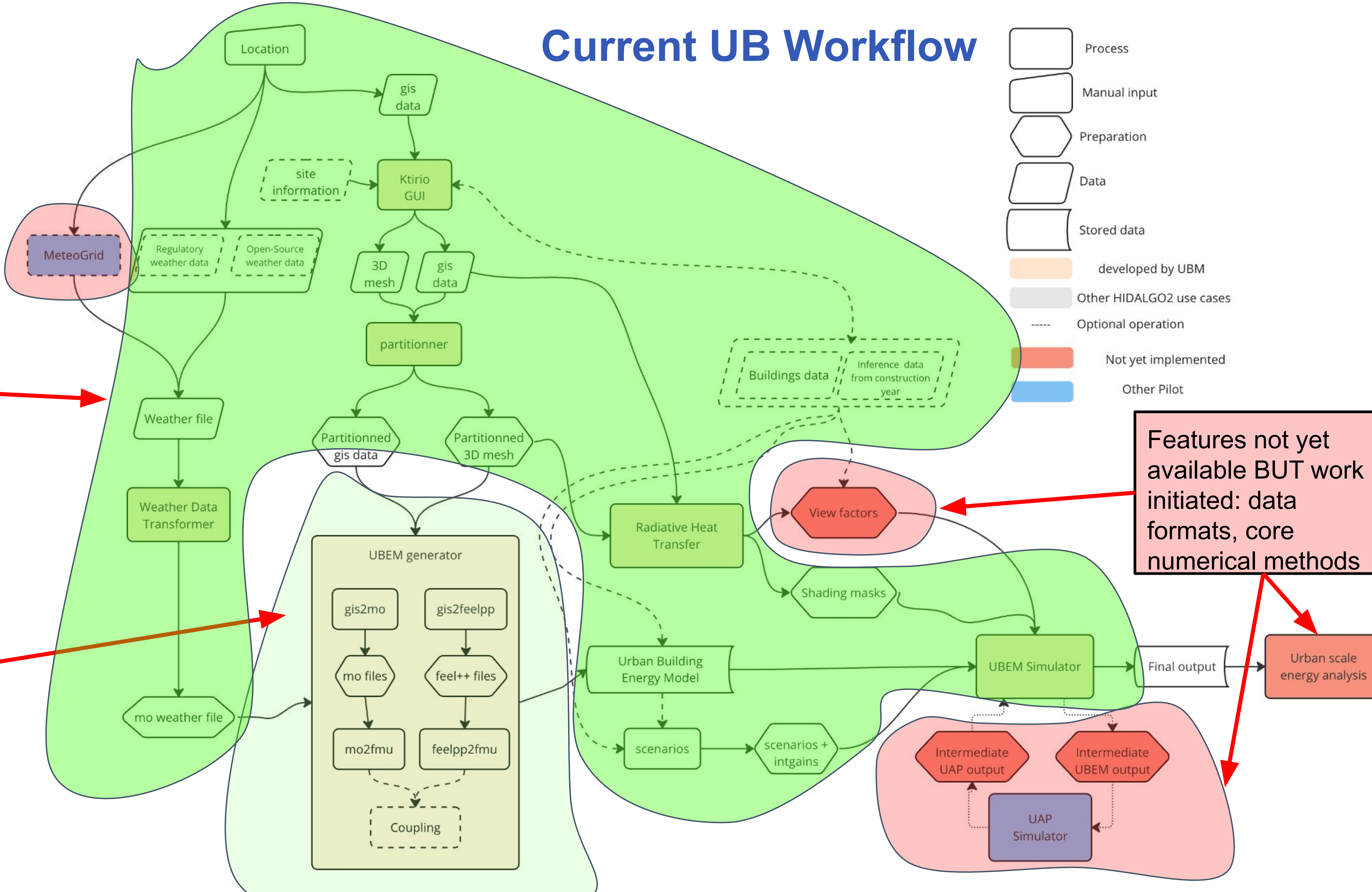
Current UB Workflow

Feature not yet available

New, work done in 2023

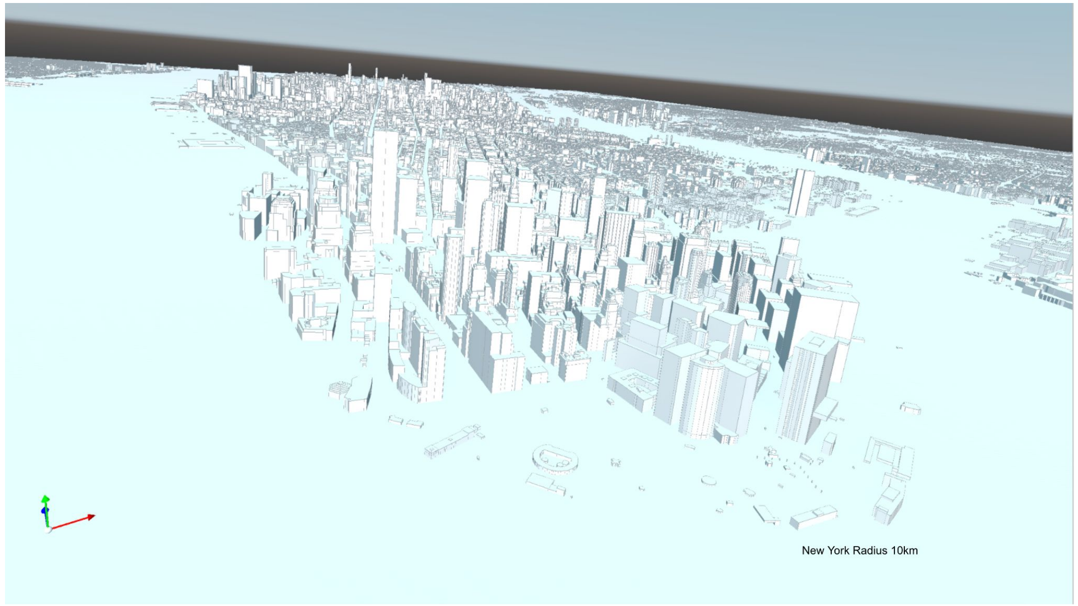
Some BEM FMU models available before 2023 for single buildings Modifications and additions in 2023

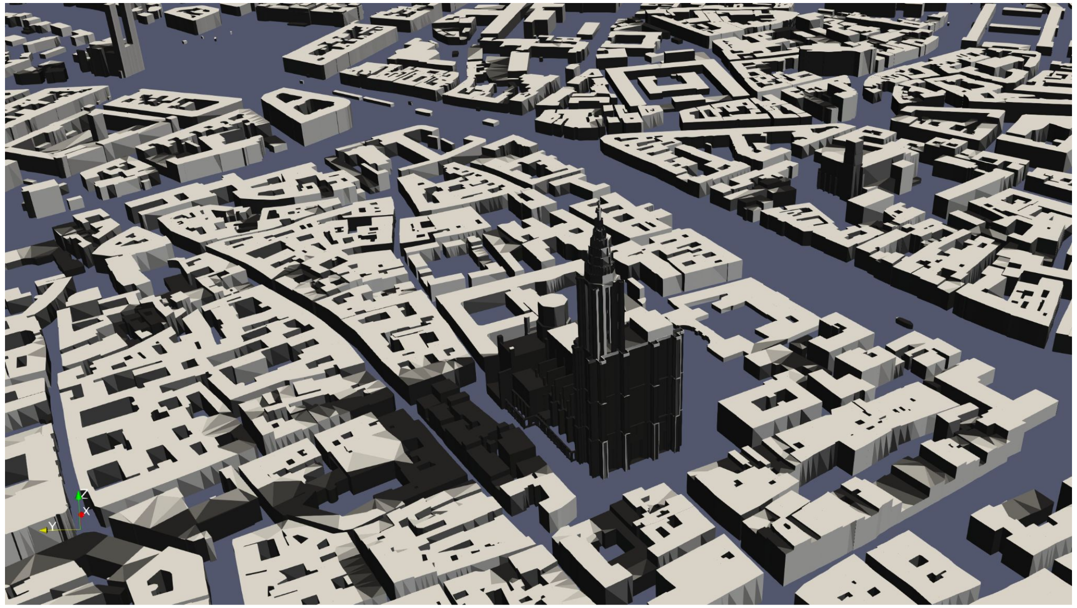
Features not yet available BUT work initiated: data formats, core numerical methods



Summary UB Pilot

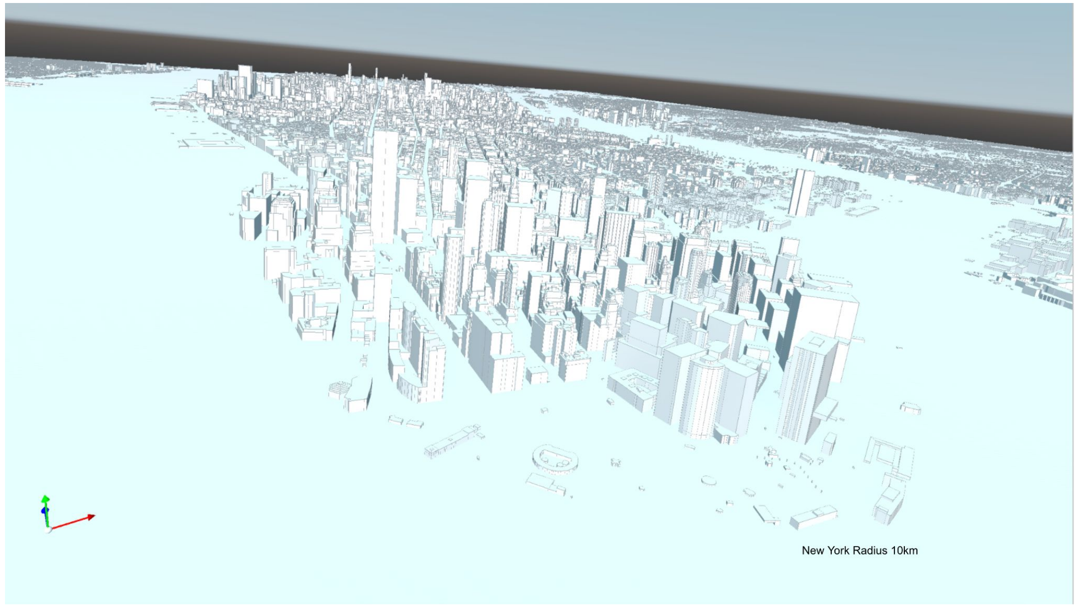
- **Framework Status:**
  - LOD-0 energy simulation(E/S) - Fully Operational at city scale
  - LOD-1 for solar masks(S/M) - operational, E/S implemented, in tests
  - LOD-0 (E/S) and LOD-1 (S/M) Coupling - finalizing
- **New Use Cases Identified:**
  - **UB UC#1 (initial one)**: City-Scale Energy Simulation
  - **UB UC#2:** Solar Masks Computation for best location of solar cells
  - **UB UC#3:** Diffuse Energy Demand Response at Scale
- **Deployment Status:**
  - Available via Apptainer on EuroHPC systems: **LUMI**, **Karolina**, **Discoverer**, **Meluxina** and PSNC/Altair
- **CI/CD Pipeline:**
  - CI/CD pipeline integrated and automated for container updates and releases
  - **Performance and verification tests at large scale** in place for quality assurance
  - Automatically **deploys codes and reports to Hidalgo2 and Castiel2** repositories

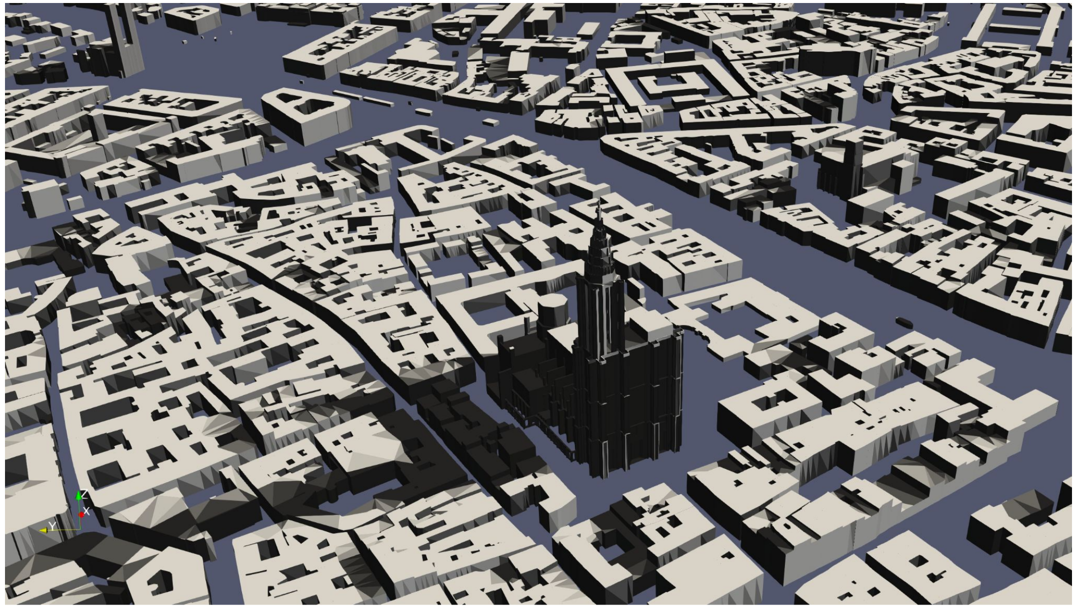




UB Next Steps

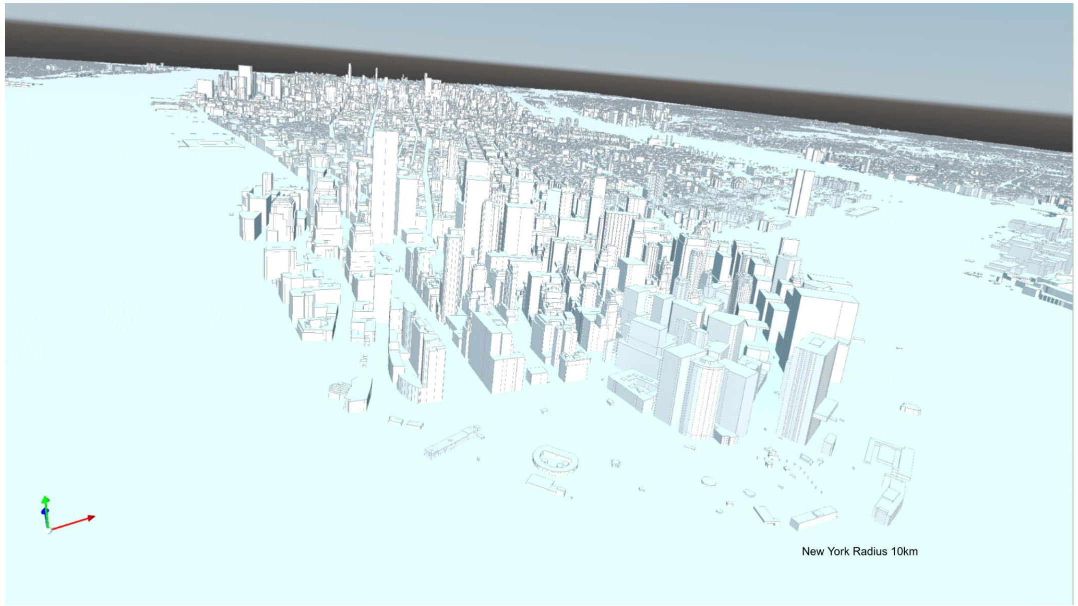
- **Modeling**
  - Finalize multifidelity (coupling lod-0 and lod-1)
  - Add roofs, vegetation, roads, urban furniture (
  - Compute view factors (interaction between building surfaces)
  - Add occupation scenario and energy policies
- **Computing / Performance**
  - Improve I/O performance, several options are under considerations
  - Implement ensemble runs based on Feel++
- **CI/CD**
  - Automate as much as possible from client app to visualisation and reporting
  - Port to remaining EuroHPC platforms and GitLab
  - Sustainable data management
- **Coupling**
  - **UAP:** generate common building surface mesh common to both pilots to exchange data
- **IA and ML**
  - Discover hidden useful information in massive datasets

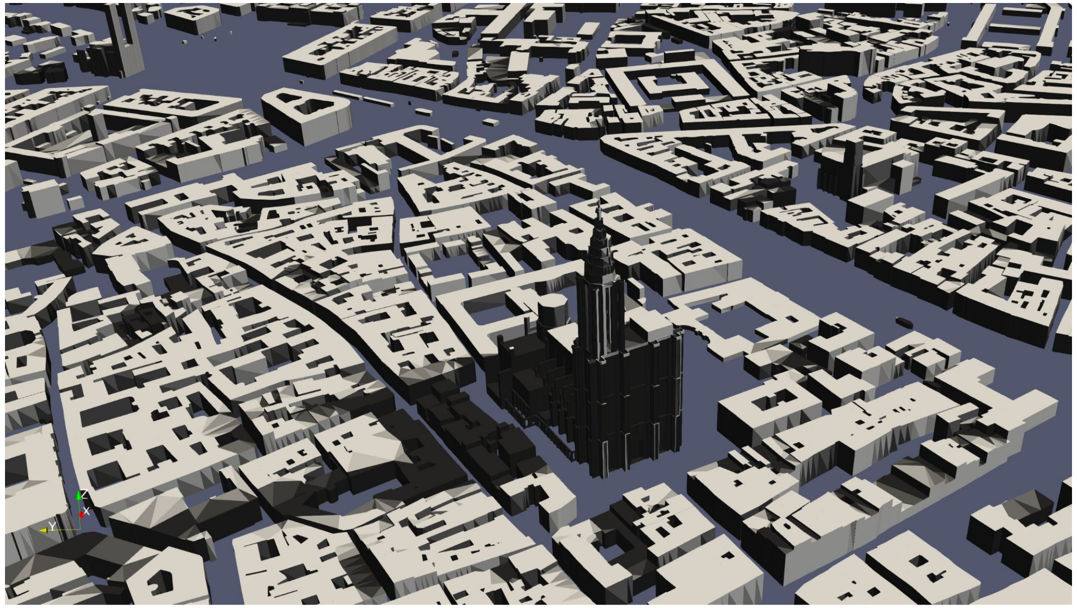




Identified Challenges/Bottlenecks

- Large-scale mesh generation : surface mesh and volume mesh (adapted for CFD)
- Parallel mesh adaptation
- Multi fidelity representation (geometry and physics)
- Mesh and physic-based/LOD partitioning and load balancing
- I/O
- Multiphysics coupling





Thank you for your attention

www.hidalgo2.eu

e-mail: office@hidalgo2.eu

Contact Cemosis

Christophe Prud'homme
christophe.prudhomme@cemosis.fr

Vincent Chabannes
vincent.chabannes@cemosis.fr

Acknowledgments

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and Poland, Germany, Spain, Hungary, France, Cyprus under grant agreement number: 101093457.

This publication expresses the opinions of the authors and not necessarily those of the EuroHPC JU and Associated Countries which are not responsible for any use of the information contained in this publication.

Co-funded by the European Union

EuroHPC Joint Undertaking

Disclaimer