

UNICORE-based Workflows for the Simulation of Organic Light-Emitting Diodes



Stefan Bozic

Ivan Kondov, Velimir Meded and Wolfgang Wenzel
Karlsruhe Institute of Technology, Germany



- Project MMM@HPC overview
- The Challenges
- Integration concept: UNICORE
- GridBean and UNICORE Workflow
- Simulation of Organic Light Emitting Diodes (OLEDs)
- Data Exchange
- OpenMolGRID
- OLED Workflow
- Sharing the Workflow
- Conclusions and outlook

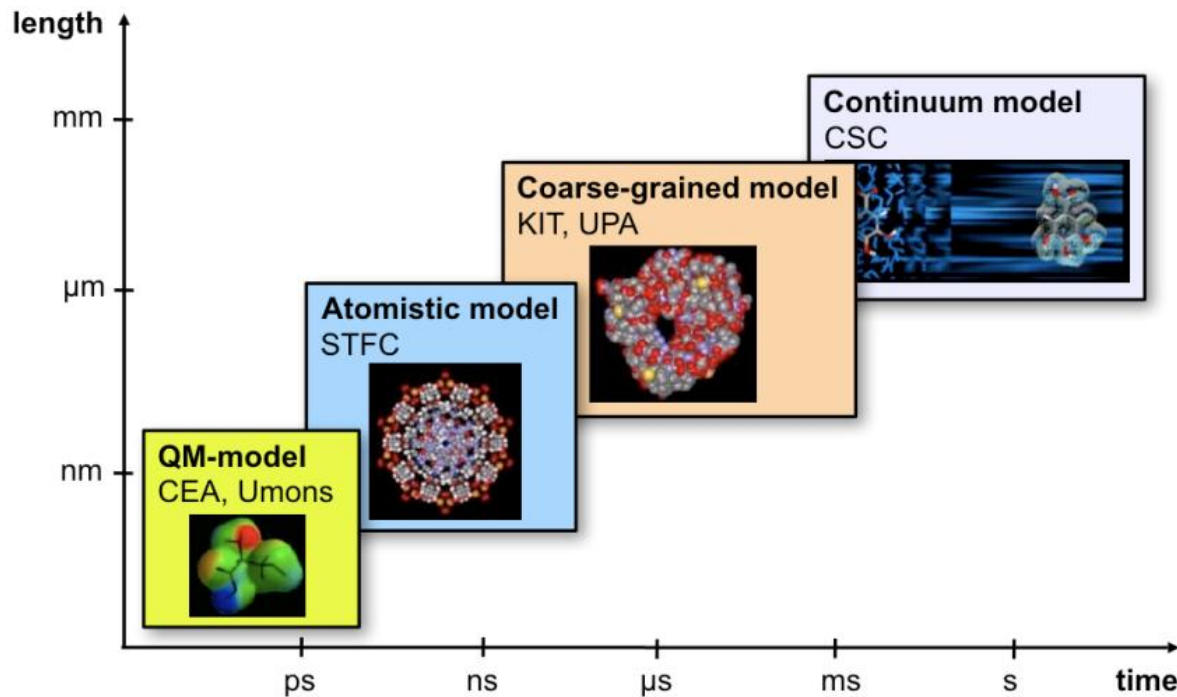


www.multiscale-modelling.eu

- **HPC centres:** CINECA, CSC, KIT and KIST (Korea)
- **Modelling and code developing groups:** University Mons, CEA, CSC, STFC, University Patras, KIT
- **Industrial partners and users:** CEA, SONY, KIT, project MINOTOR
- **Cooperating projects:** PRACE, MINOTOR, D-Grid and NGI-DE



The challenges



- Integration on different size and time scales to address real-life problems in nano-materials science
- Develop an easy to use solution for non-experts: industrial and experimental groups

Reusability

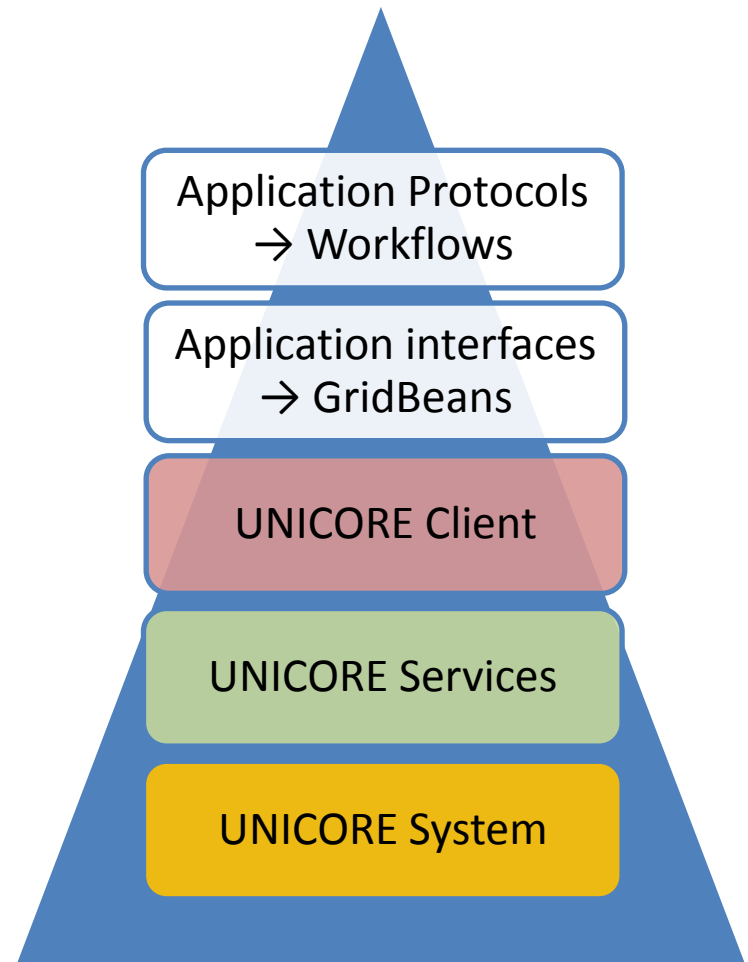
Data complexity

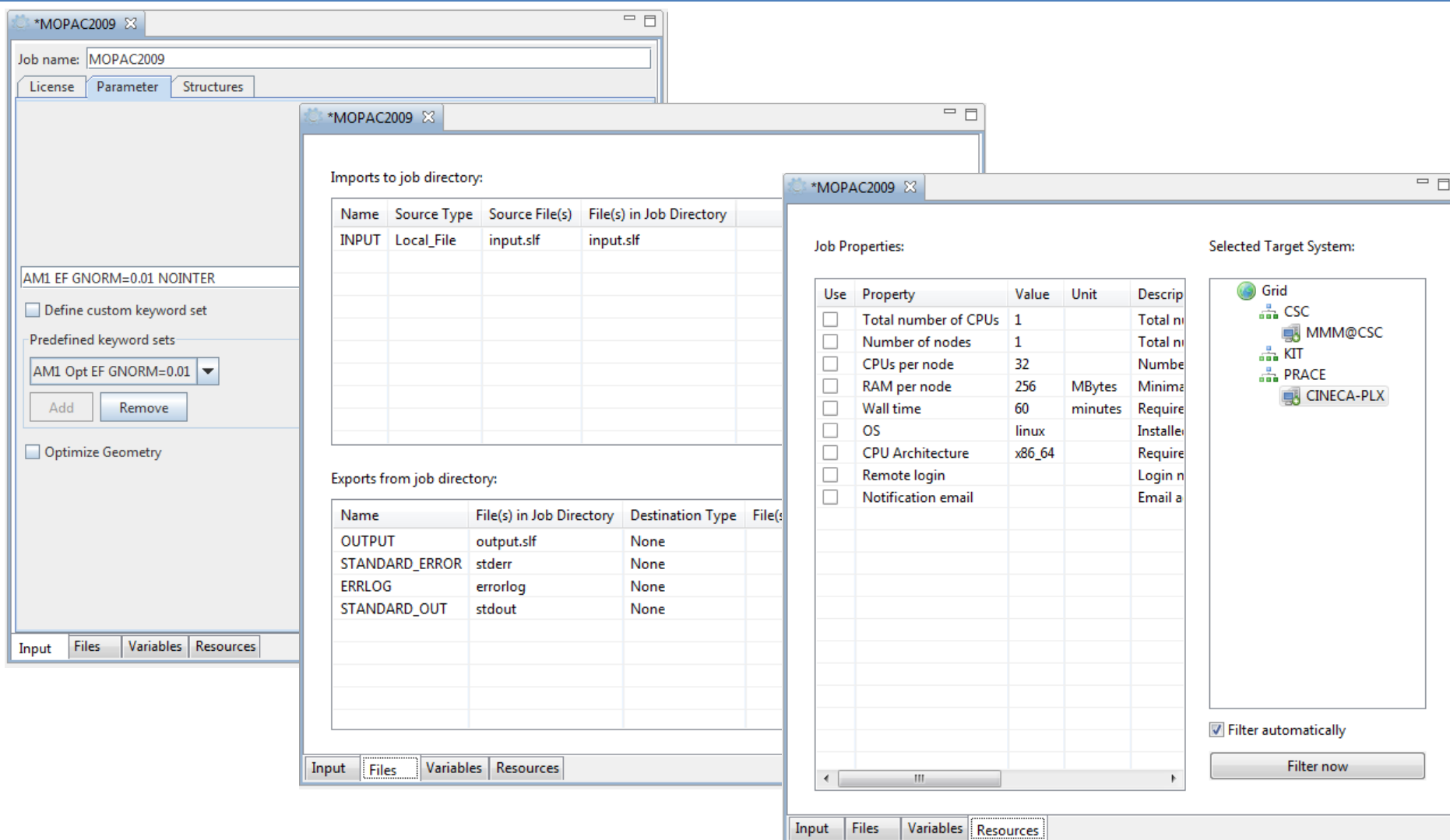
Solution for licensing issues

Security & Reliability

Capacity & Capability

- Provision of simulation tools and services that can be combined in many different application workflows
- Adaptable, reusable and extendable interfaces & workflows based on UNICORE
- Access to distributed HPC/HTC resources via UNICORE services





The screenshot displays three overlapping windows from the MOPAC GridBean application, all titled '*MOPAC2009'.

Left Window (Job Configuration):

- Job name: MOPAC2009
- Tabs: License, Parameter, Structures
- Parameter tab is active, showing:
 - AM1 EF GNORM=0.01 NOINTER
 - ☐ Define custom keyword set
 - Predefined keyword sets:
 - AM1 Opt EF GNORM=0.01 (selected)
 -
 - ☐ Optimize Geometry
- Bottom tabs: Input, Files, Variables, Resources

Middle Window (Imports/Exports):

Imports to job directory:

Name	Source Type	Source File(s)	File(s) in Job Directory
INPUT	Local_File	input.slif	input.slif

Exports from job directory:

Name	File(s) in Job Directory	Destination Type	File(s)
OUTPUT	output.slif	None	
STANDARD_ERROR	stderr	None	
ERRLOG	errorlog	None	
STANDARD_OUT	stdout	None	

Bottom tabs: Input, Files, Variables, Resources

Right Window (Job Properties):

Job Properties:

Use	Property	Value	Unit	Descrip
<input type="checkbox"/>	Total number of CPUs	1		Total n
<input type="checkbox"/>	Number of nodes	1		Total n
<input type="checkbox"/>	CPUs per node	32		Numbe
<input type="checkbox"/>	RAM per node	256	MBytes	Minima
<input type="checkbox"/>	Wall time	60	minutes	Require
<input type="checkbox"/>	OS	linux		Installe
<input type="checkbox"/>	CPU Architecture	x86_64		Require
<input type="checkbox"/>	Remote login			Login n
<input type="checkbox"/>	Notification email			Email a

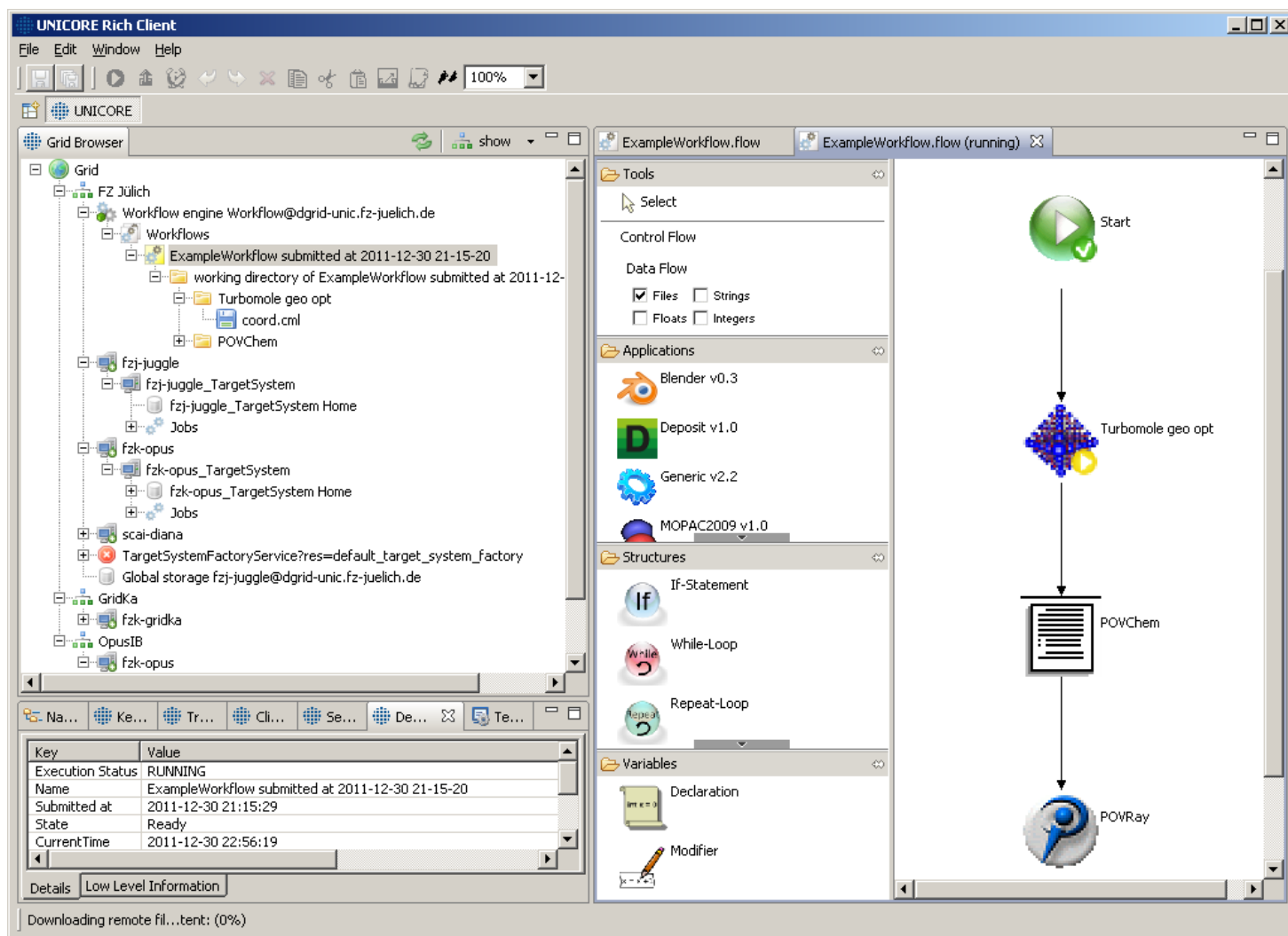
Selected Target System:

- Grid
 - CSC
 - MMM@CSC
 - KIT
 - PRACE
 - CINECA-PLX

☒ Filter automatically

Bottom tabs: Input, Files, Variables, Resources

Application flow: Example



Data flow: Example

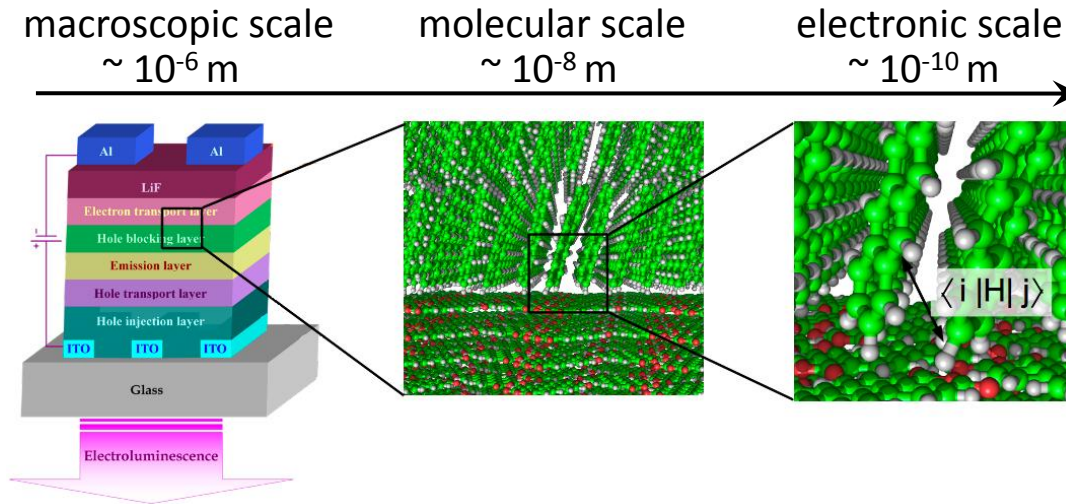
The screenshot displays the UNICORE Rich Client interface. The main window is titled "UNICORE Rich Client" and features a menu bar (File, Edit, Window, Help) and a toolbar. The interface is divided into several panes:

- Grid Browser:** A tree view on the left showing the hierarchy of grid resources. It includes "CINECA" (with sub-entries for Workflow engine, Workflows, CINECA-PLX, CINECA-SP6, and Storage factory) and "FZ Julich" (with sub-entries for Workflow engine, DGIREF_SCC, fzk-juggle, fzk-opus, and TargetSystemFactoryService).
- Workflow Editor:** The central area displays a workflow diagram titled "OLED workflow.flow". The diagram shows a sequence of tasks: "Turbomole1" (a blue diamond icon) followed by "Deposit1" (a green square icon). "Turbomole1" has two output ports labeled "stderr" and "stdout". "Deposit1" has five output ports labeled "errorlog", "last.pdb", "output.slf", "stderr", and "stdout". Below "Deposit1" is a "POVRay1" task (a blue sphere icon) which has three output ports labeled "*.png", "stderr", and "stdout".
- Tools and Structures:** A vertical pane on the right contains a "Tools" section with "Select", "Control Flow", and "Data Flow" (checked). Below this are "Applications" (Blender v0.3, Deposit v1.0, Generic v2.2) and "Structures" (If-Statement, While-Loop, Repeat-Loop). A "Variables" section includes "Declaration" and "Modifier".
- Terminal View:** A pane at the bottom right for executing commands.
- Details:** A pane at the bottom left showing a table of key-value pairs for the selected workflow.

Key	Value
Name	Workflows
State	Ready
Type	Enumeration
URI	https://grid.cineca.it:9111/CINECA-WORKFLOW/services/WorkflowFactory?res=default_workflk

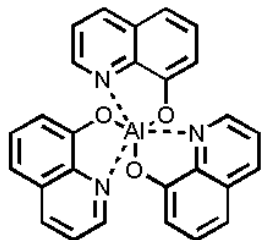
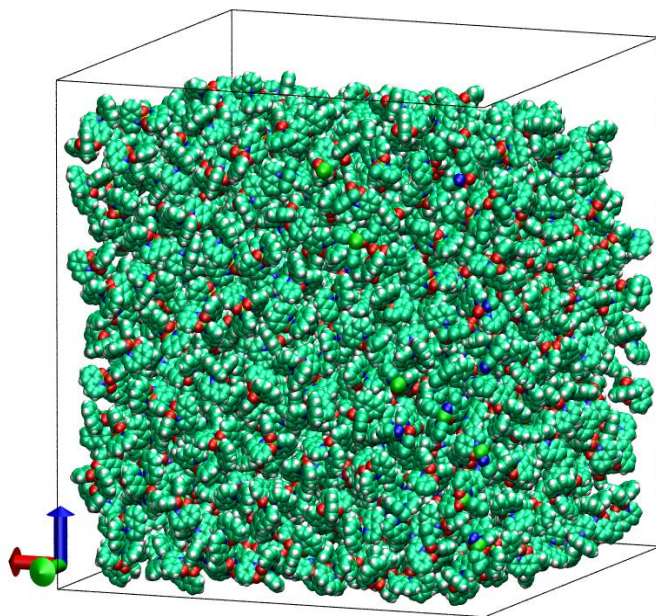
refreshing Grid: (0%)

OLED: architecture and modelling tools



continuum model (FEA)	coarse-grained model (CG)	Atomistic model (MM)	QM model (QM)
Elmer	ToFeT (KMC)	DEPOSIT	MOPAC
FEAP	End-bridging MC	LAMMPS	TURBOMOLE
	Transporter	DL_POLY	BigDFT

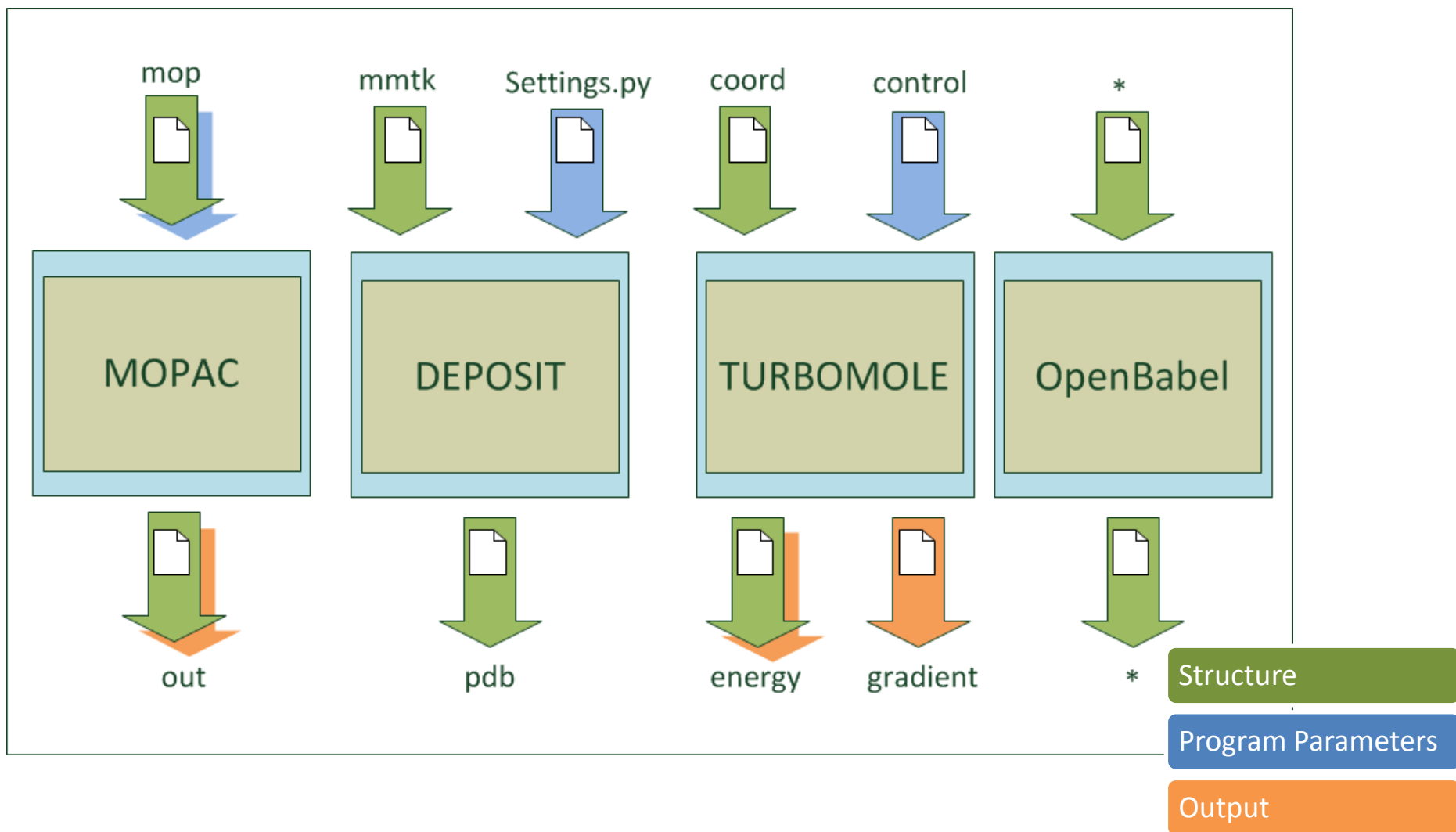
OLED: simulation protocol for charge transport in Alq3 disordered films



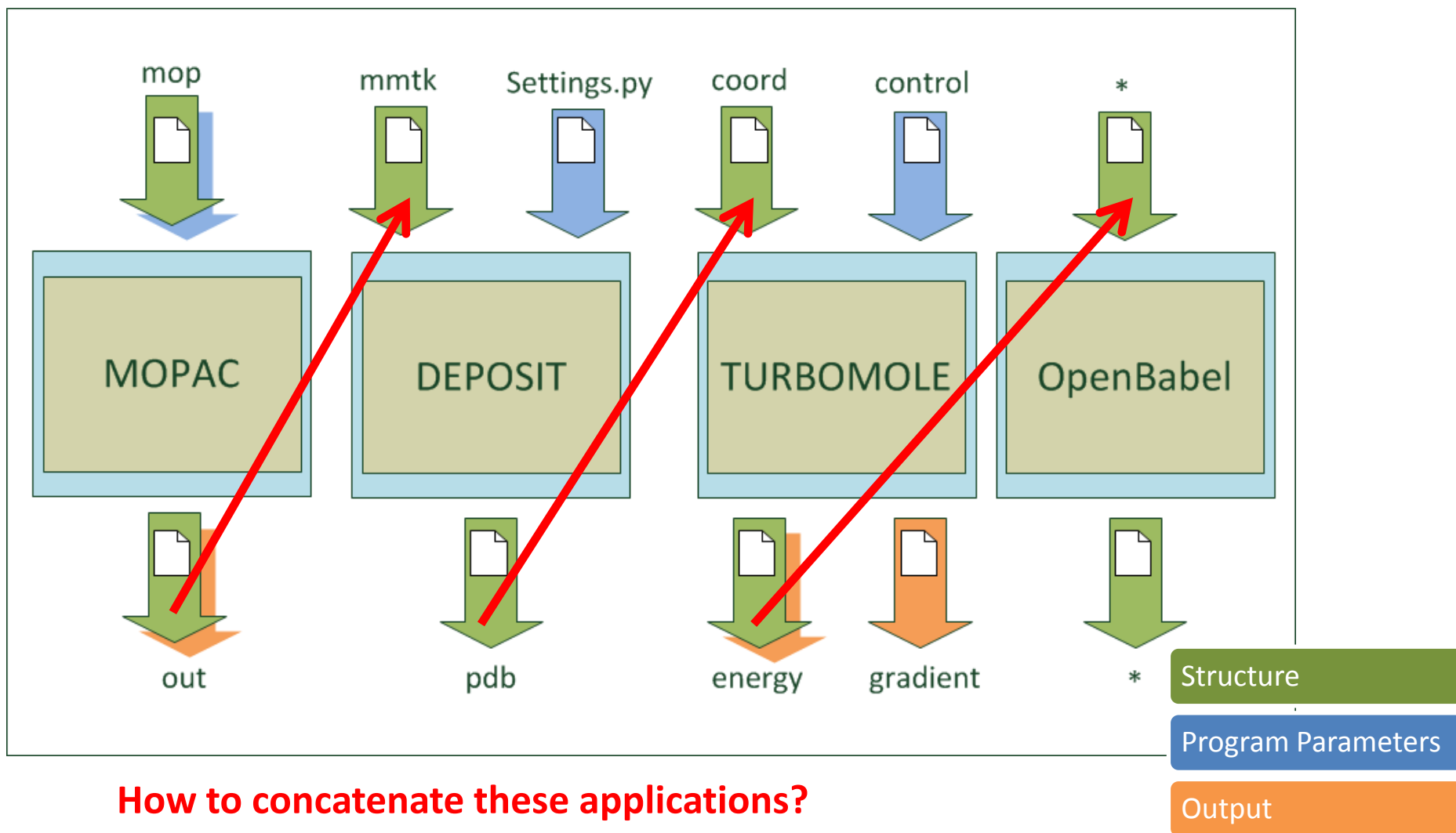
J. J. Kwiatkowski, J. Nelson, H. Li,
J. L. Bredas, W. Wenzel, and C.
Lennartz, Phys. Chem. Chem. Phys.,
2008, 10, 1852–1858.

- Film deposition (or MD)
 - Generate disordered film morphologies
- QM calculations of hopping sites
 - Calculate HOMO, LUMO, LUMO+1 etc energies.
 - Electronic couplings reorganization energies
 - Calculate charge hopping rates
- Kinetic Monte Carlo (KMC)
 - Calculate charge (electron-hole) mobility
 - Calculate current density

Application Input/Output files



Application Input/Output files

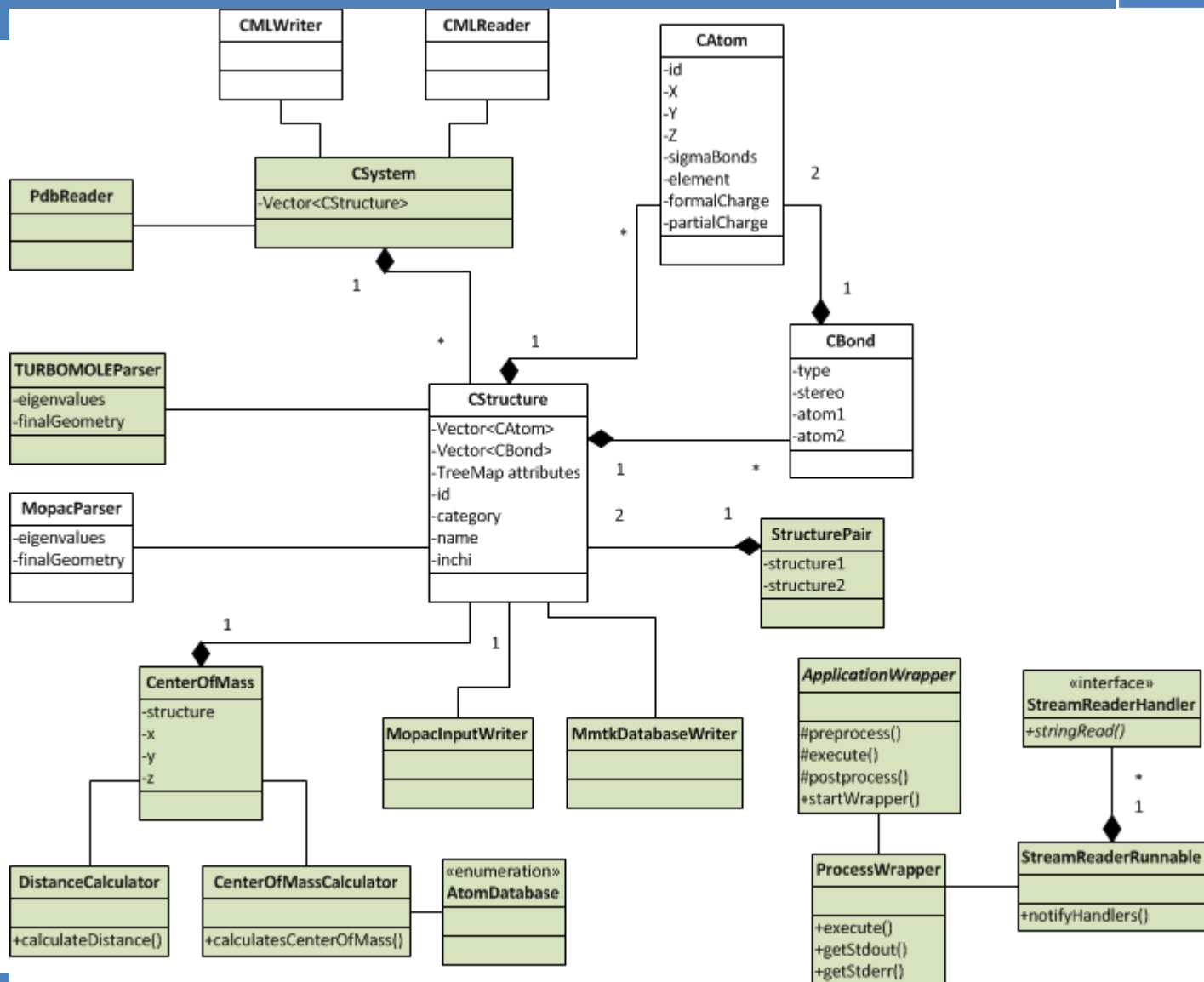


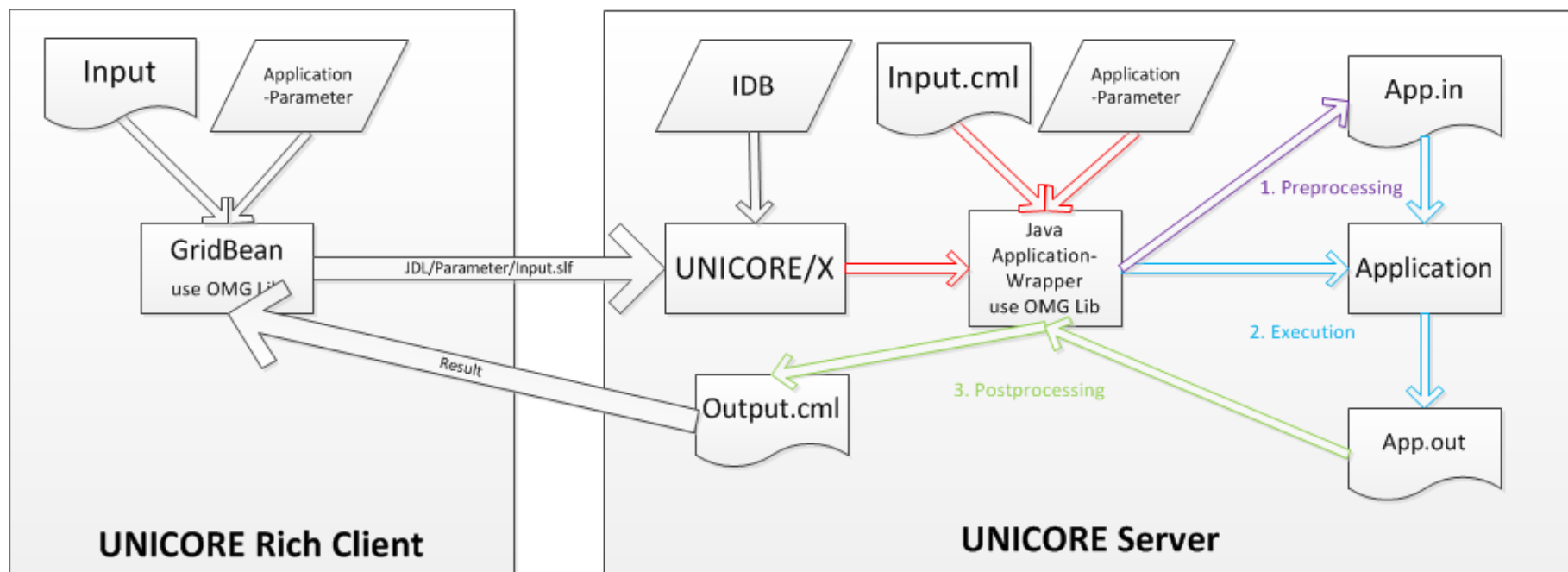
How to concatenate these applications?

- Structure Data exchange format: Chemical Markup Language (CML)
 - is the *de facto* XML for chemistry
 - Extendable with additional domain specific attributes (charges, temperature, density) in CML dictionaries
- Data flow management with the OpenMolGRID library

S. Sild et al., LNCS 3470, 464, Springer (2005); S. Sild et al., J. Chem. Inf. Model., 46, 953 (2006).

 - Read, write and convert chemical file format
 - Provides a data model for molecular information
 - Application Wrapper API
- Open Babel
 - Read, write and convert chemical file format

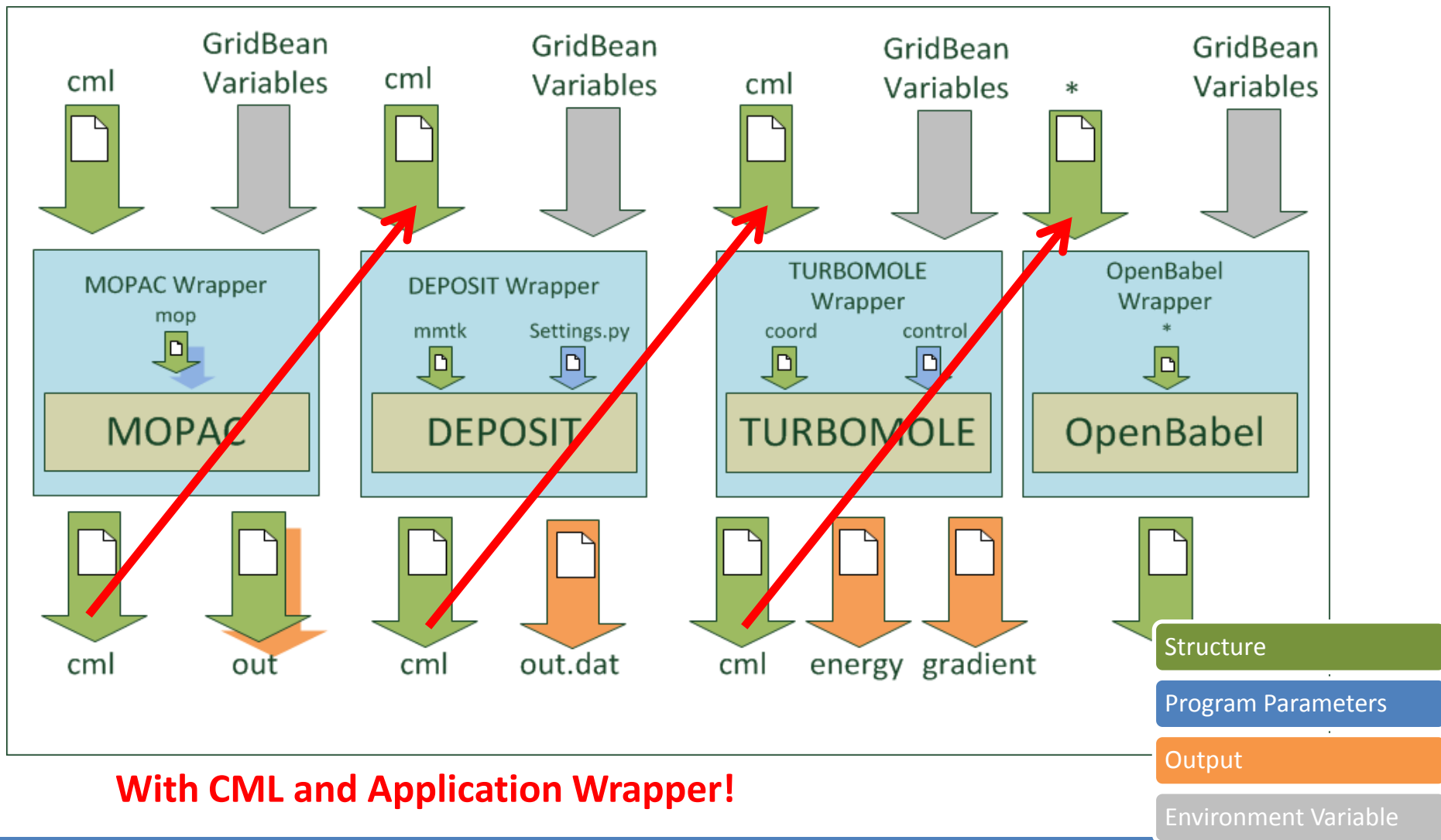




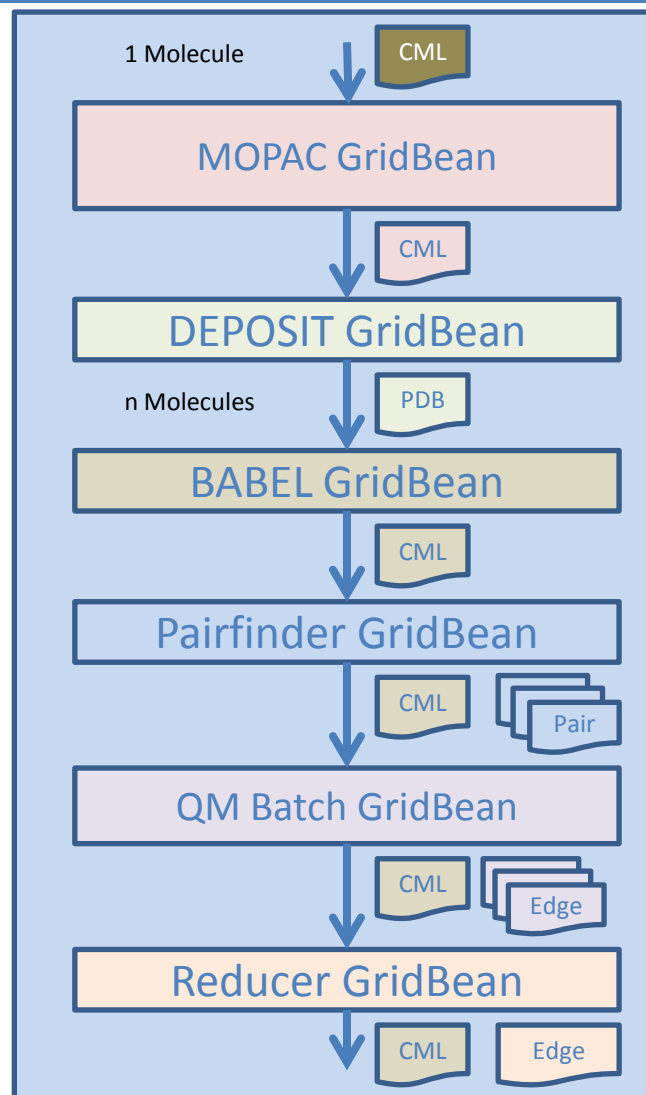
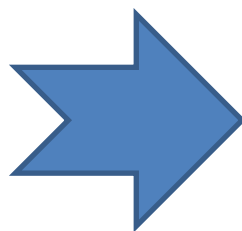
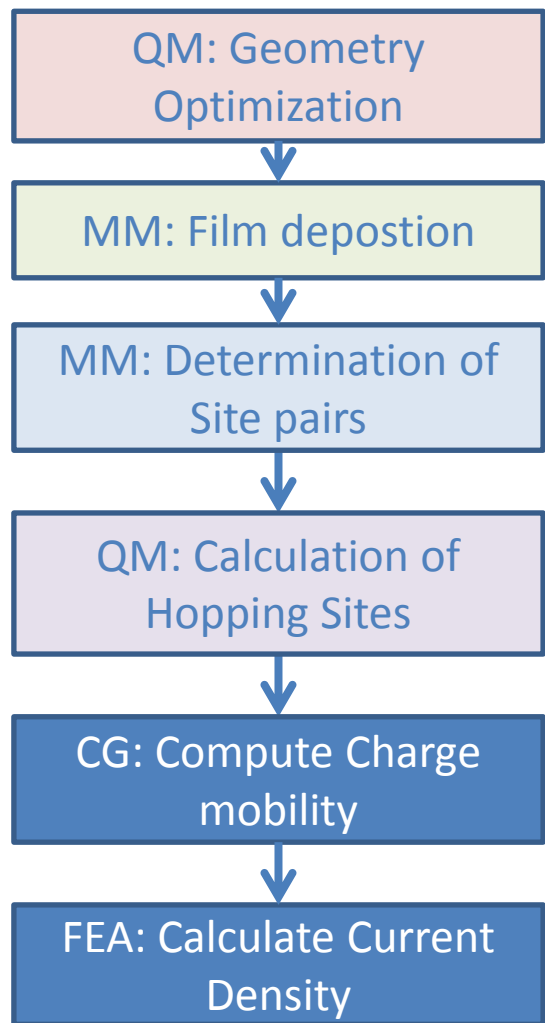
Application Wrapper Lifecycle

- 1. Preprocessing:** Validation of App. Parameter, Generation of App. specific Input Files
- 2. Execution:** Run App(s) in separate Process(es), Monitoring of stdout/stderr (allows interaction with the application)
- 3. Postprocessing:** Error Handling, Parsing App Output, Creation of Workflow Data

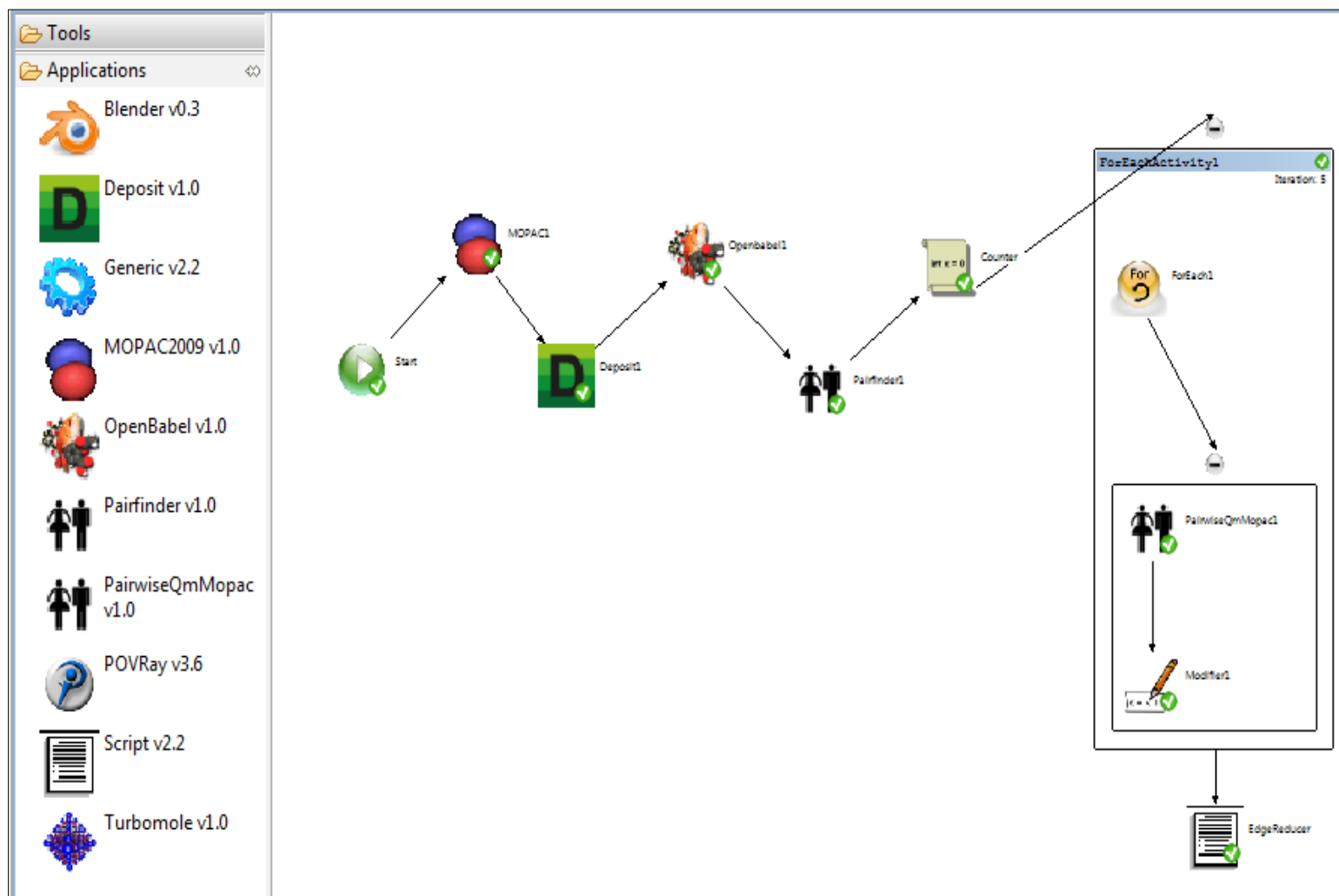
Application Wrapper Input/Output files



Simulation protocol



UNICORE Workflow



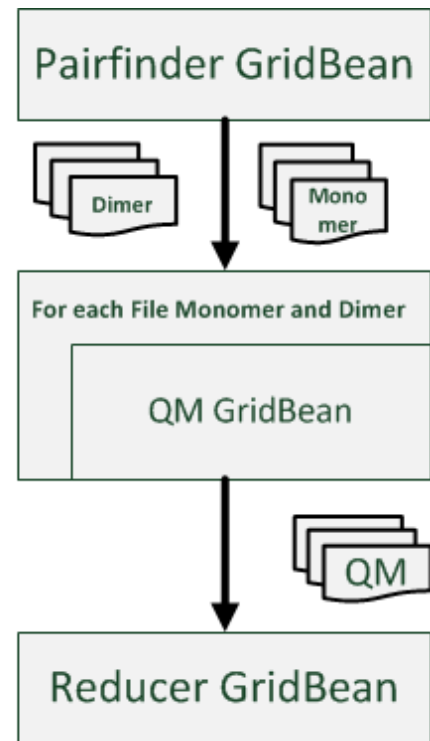
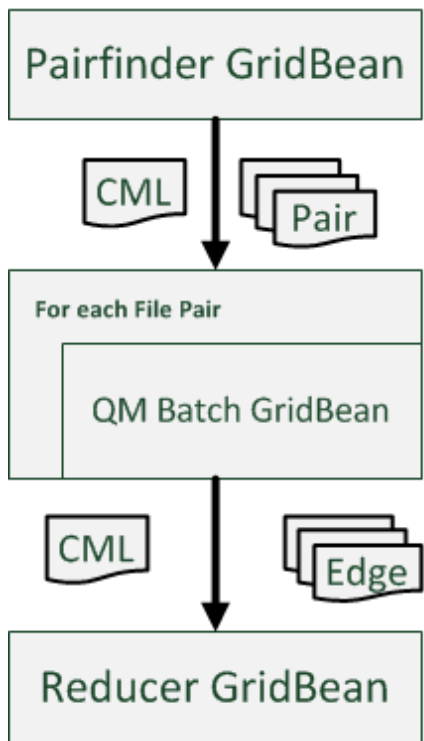
A deposition film has **1.000 – 1.000.000** Hopping Site candidates (Molecule Pairs)
Each Hopping Site needs 3 QM calculations (2 Monomers and 1 Dimer)

QM-Batch Jobs

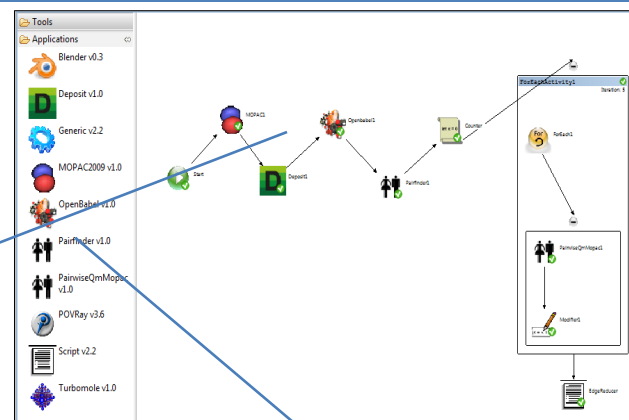
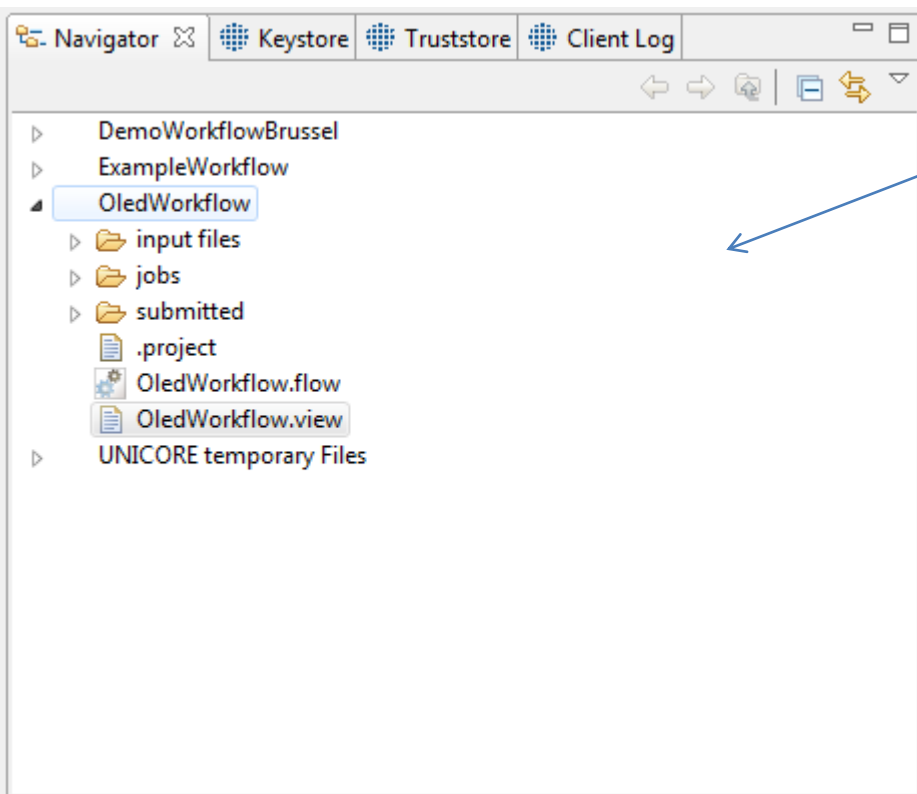
- Short QM calculations (~ 0.5 - 5 min)
- Each Pair file: includes n Molecule Pairs
- Each Job: n * 3 QM calculations
- Less Jobs: Minimize service overhead
- Each QM Code needs additional GridBeans & Wrapper

QM-Single Jobs

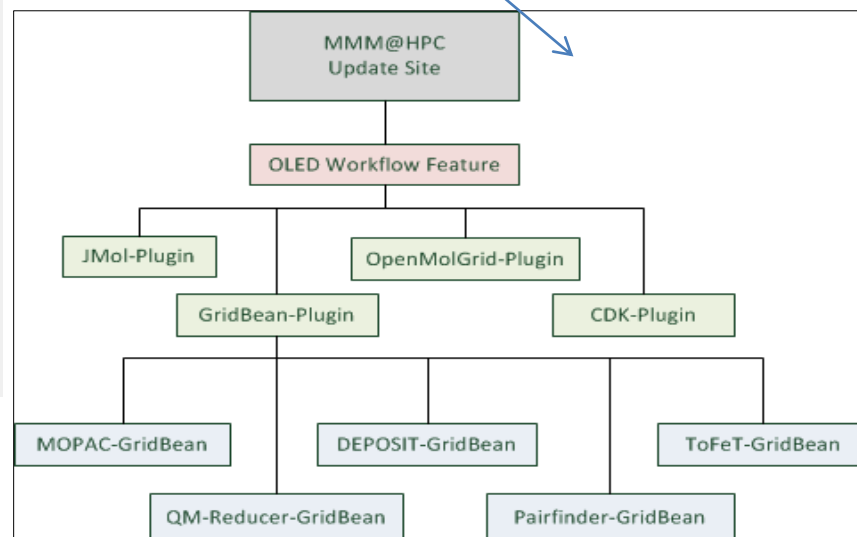
- Long QM calculation (30 min +)
- Each Job: one QM calculation
- Many Jobs & Files
- Reuse of QM GridBeans
- Each QM Code needs a special Reducer



1) Workflow Project



2. GridBeans in MMM@HPC Update Site



- With UNICORE we provide an optimal low-effort/low-cost solution for multiscale modelling
- GridBeans → App Interfaces
- Workflows → Simulation protocol
- Data Exchange in WFs between applications handled with CML, Open Babel and OpenMolGRID
- Different approaches on the calculation of Hopping Sites

Current work

- Integration of the CG and FEA steps into the OLED Workflow
- Elmer, DL_POLY and BigDFT GridBeans
- Simulation of whole OLED devices
- Workflows for Molecular Electronics, Carbon Molecular Devices and Li-Ion Batteries

Acknowledgments



- All consortium partners in MMM@HPC
- Funding from the EC



- Partner projects, supporting infrastructures and software

