

Process Bigraph Schema: A framework for multi-scale, multi-algorithmic modeling

Eran Agmon, PhD

Center for Cell Analysis and Modeling, Department of Molecular Biology and Biophysics, Department of Biomedical Engineering
University of Connecticut School of Medicine

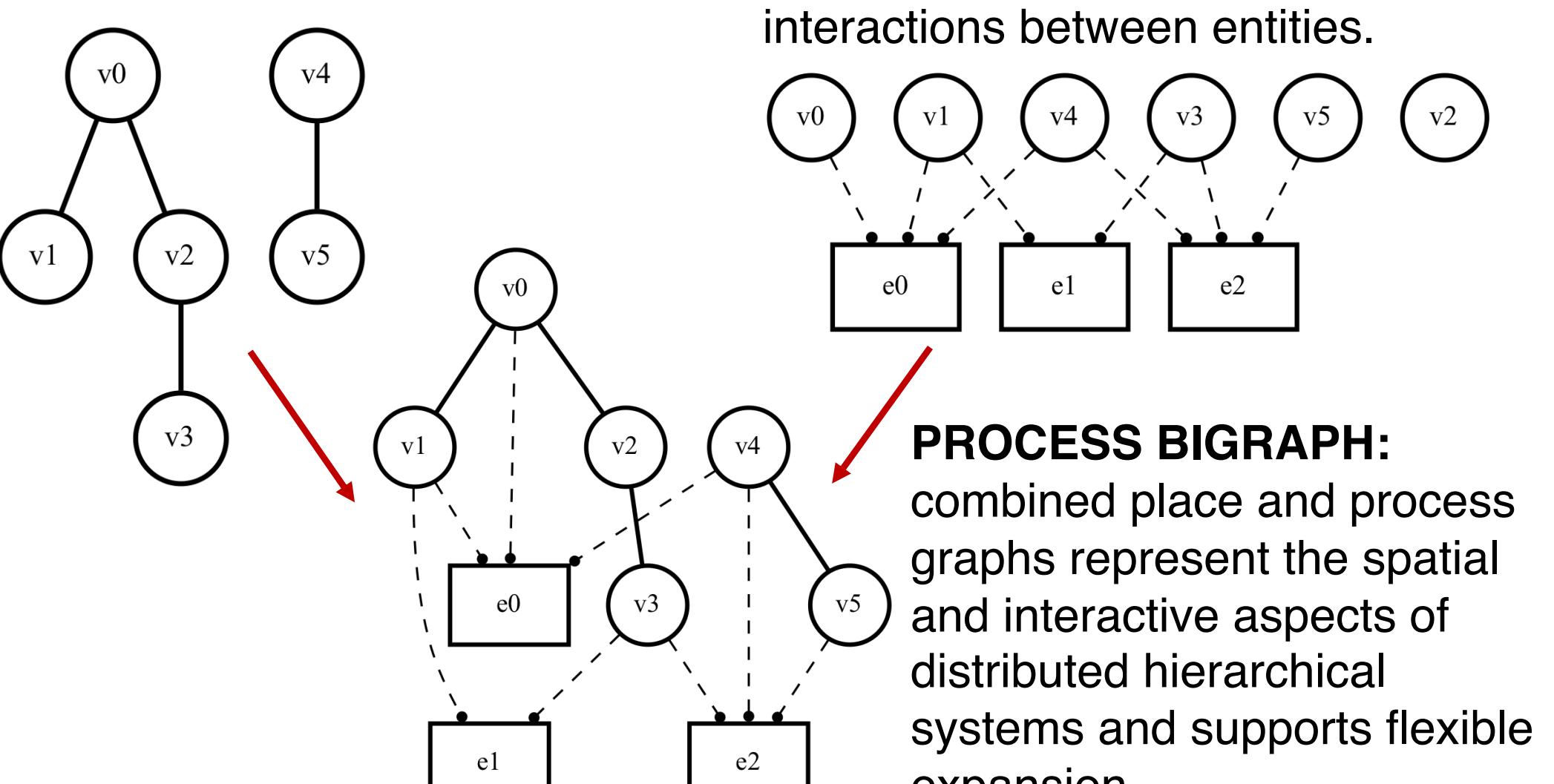


Multiscale modeling in biology poses a challenge akin to piecing together a multi-dimensional puzzle due to the diverse data and models it requires. To harmonize efforts across different domains of biology, a robust infrastructure is needed to manage dynamic data structures and control decentralized information flow. Our prior work on Vivarium [1] offered a software solution for interfacing and orchestrating heterogeneous models across scales – using a framework called *Process Bigraphs*.

The practical application of this framework calls for a standard exchange format that bridges modeling methodologies and software systems. Process Bigraph Schema (PBS) aims to be this standard, providing a shared API with a robust type system and a JSON-based format for model consistency and interoperability across simulators. PBS's composition protocol allows users to declare process interfaces, connection patterns, and orchestration methods, which can be flexibly composed together in integrative simulations.

PROCESS BIGRAPHS

PLACE GRAPH: a directed rooted tree, represents spatial nesting or other hierarchy between variables.



SCHEMA: BASIC ELEMENTS

STORES: Hold the system variables, which can be of any declared type from float/int to arrays, tables, images.

```
{ "store1": { '_type': 'float' }, }
```

HIERARCHY: JSON nesting structure supports placing stores within stores in the place graph.

```
{ "store1": { "store1.1": { "store1.1.1": 'Any', "store1.1.2": 'Any', "store1.1.3": { "store1.1.3.1": 'Any', }, }, "store1.2": 'Any', } }
```

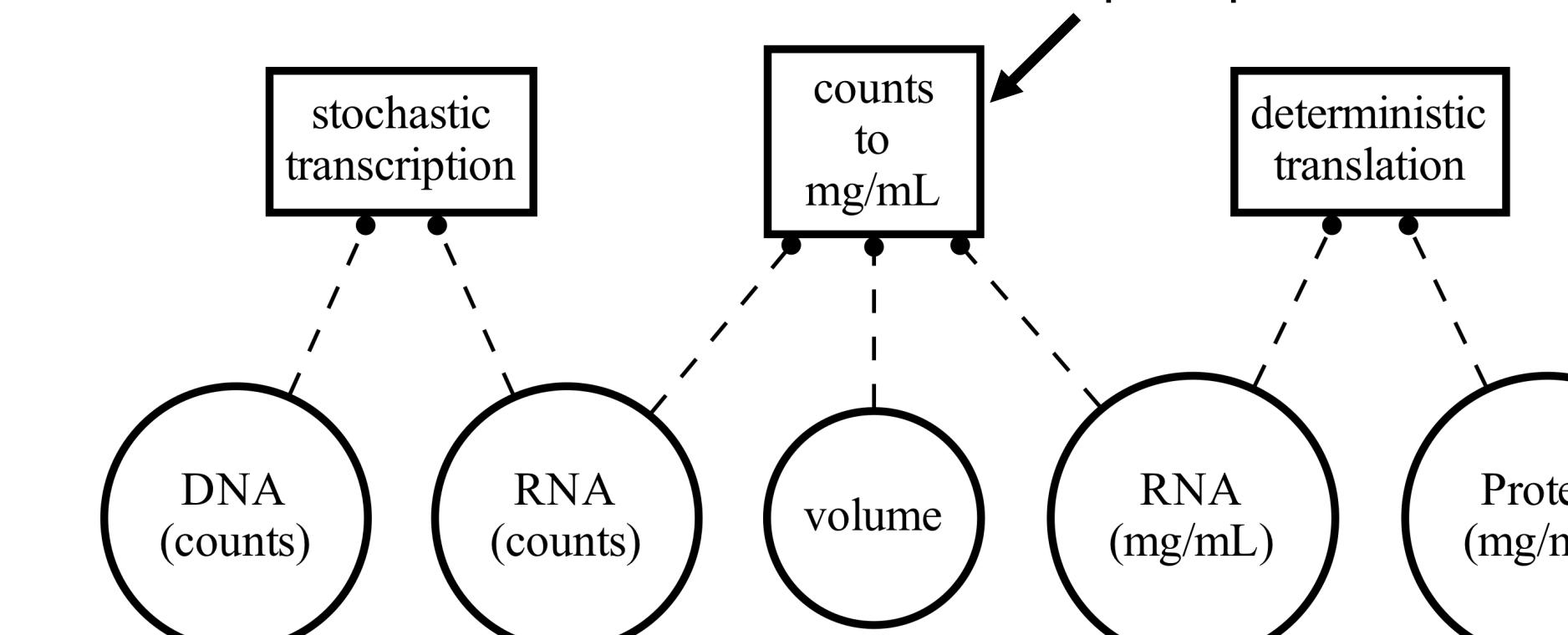
PROCESS INTERFACE: A standard port declaration allows any process that supports the API to plug into a composite simulation.

```
{ "process1": { '_type': 'process:core', '_address': 'URI:000000', '_config': { 'parameter': 'Any' }, '_ports': { 'port1': 'Any', 'port2': 'Any', } }
```

SCHEMA: COMPOSITION

```
{
  'stochastic transcription': { '_type': 'process:stochastic', 'wires': { 'DNA': 'DNA (counts)', 'RNA': 'RNA (counts)' } },
  'counts to mg/mL': { '_type': 'adapter:counts-concs', 'wires': { 'counts': 'RNA (counts)', 'concs': 'RNA (mg/mL)', 'volume': 'volume' } },
  'deterministic translation': { '_type': 'process:deterministic', 'wires': { 'RNA': 'RNA (mg/mL)', 'Protein': 'Protein (mg/mL)' } },
}
```

an “adapter” process

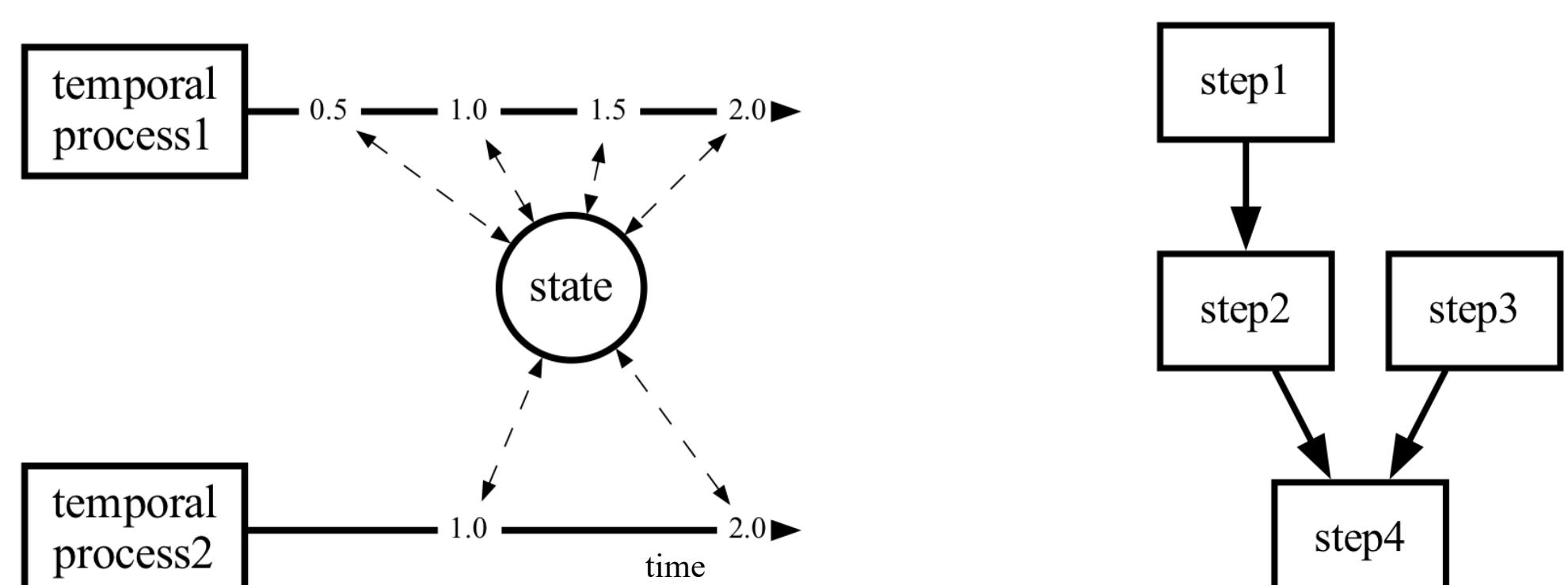


SCHEMA: ORCHESTRATION

A composite simulator’s orchestration method determines the order by which processes are triggered. Including this in the schema allows for flexible declaration of distributed control flow.

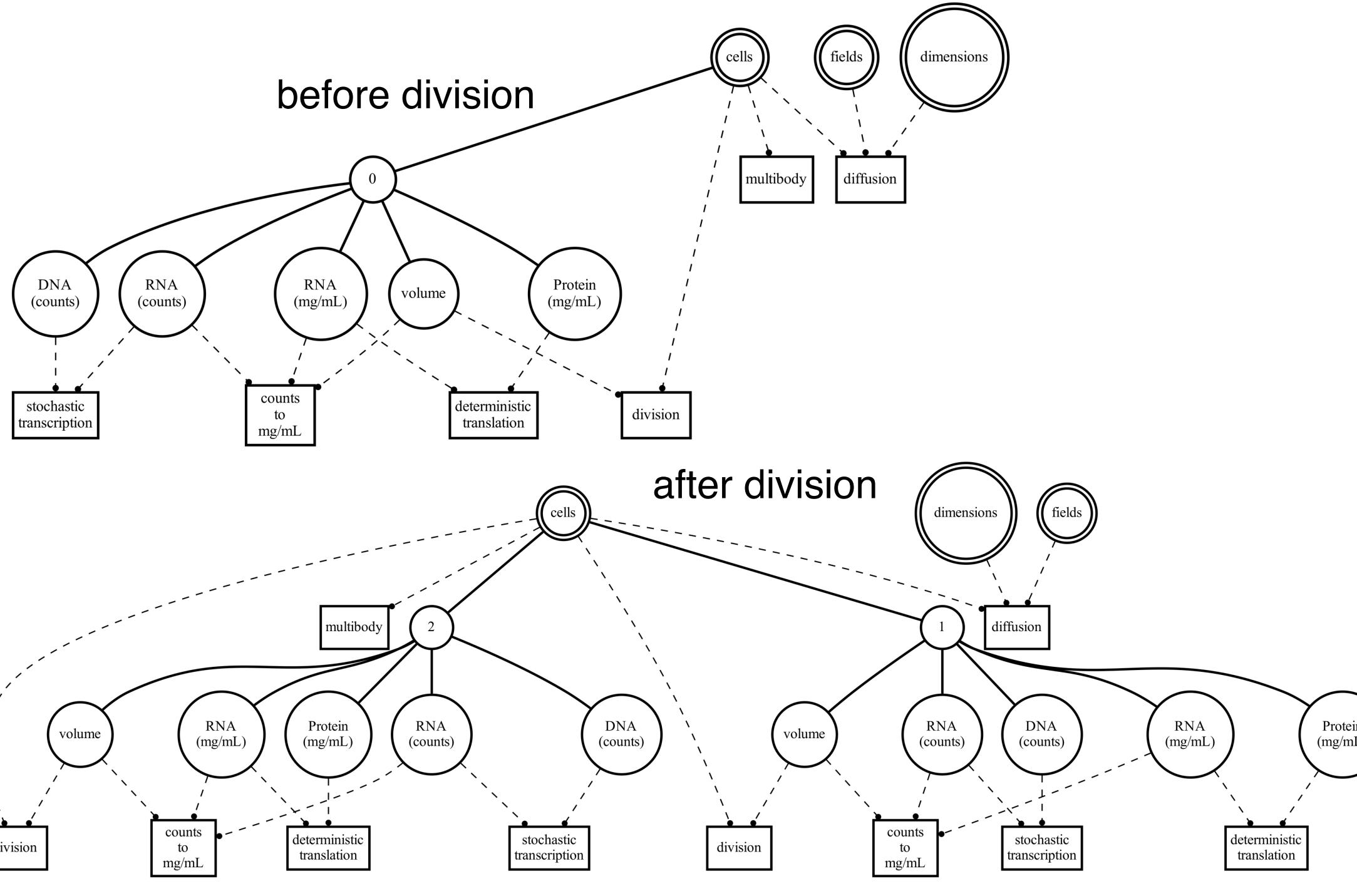
MULTI-TIMESTEPPING

Temporal processes provide the composite with their preferred timesteps, which can be updated adaptively in a discrete-event simulation.



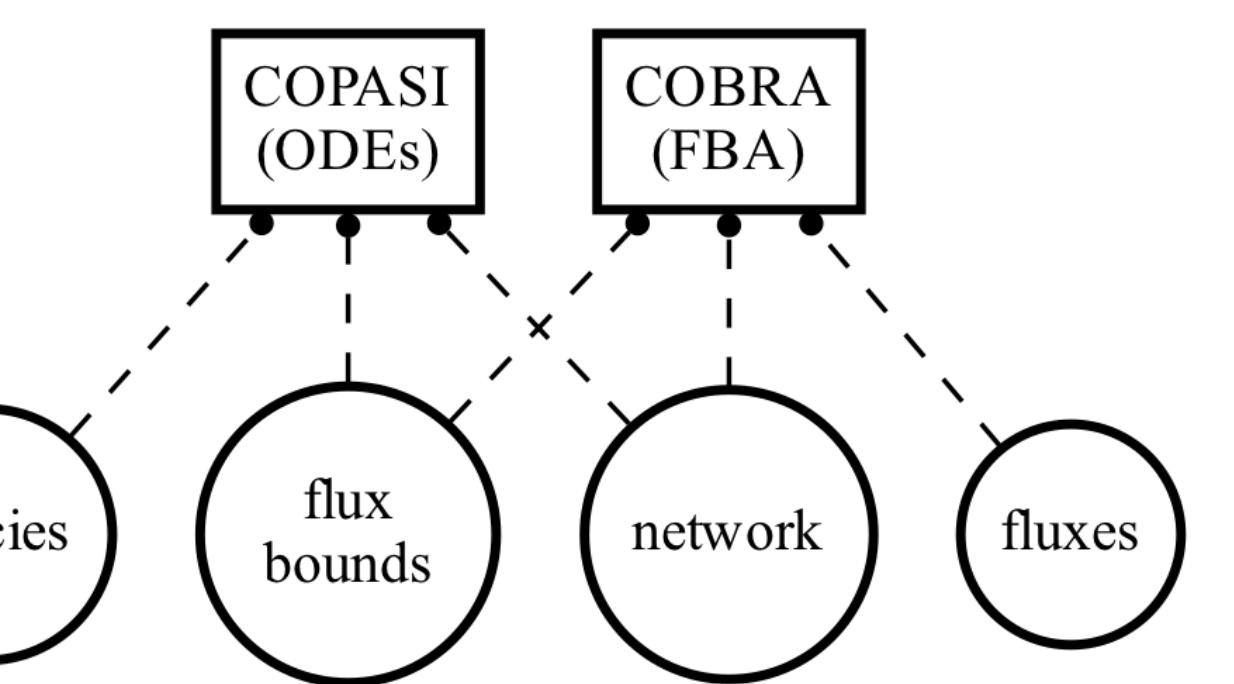
EVENT-DRIVEN GRAPH REWRITE

Processes can trigger graph rewrites during runtime, which includes adding, removing, or moving stores and process. In this example, a “division” process triggers the division of cell 0, generating fresh new cells 1 and 2.



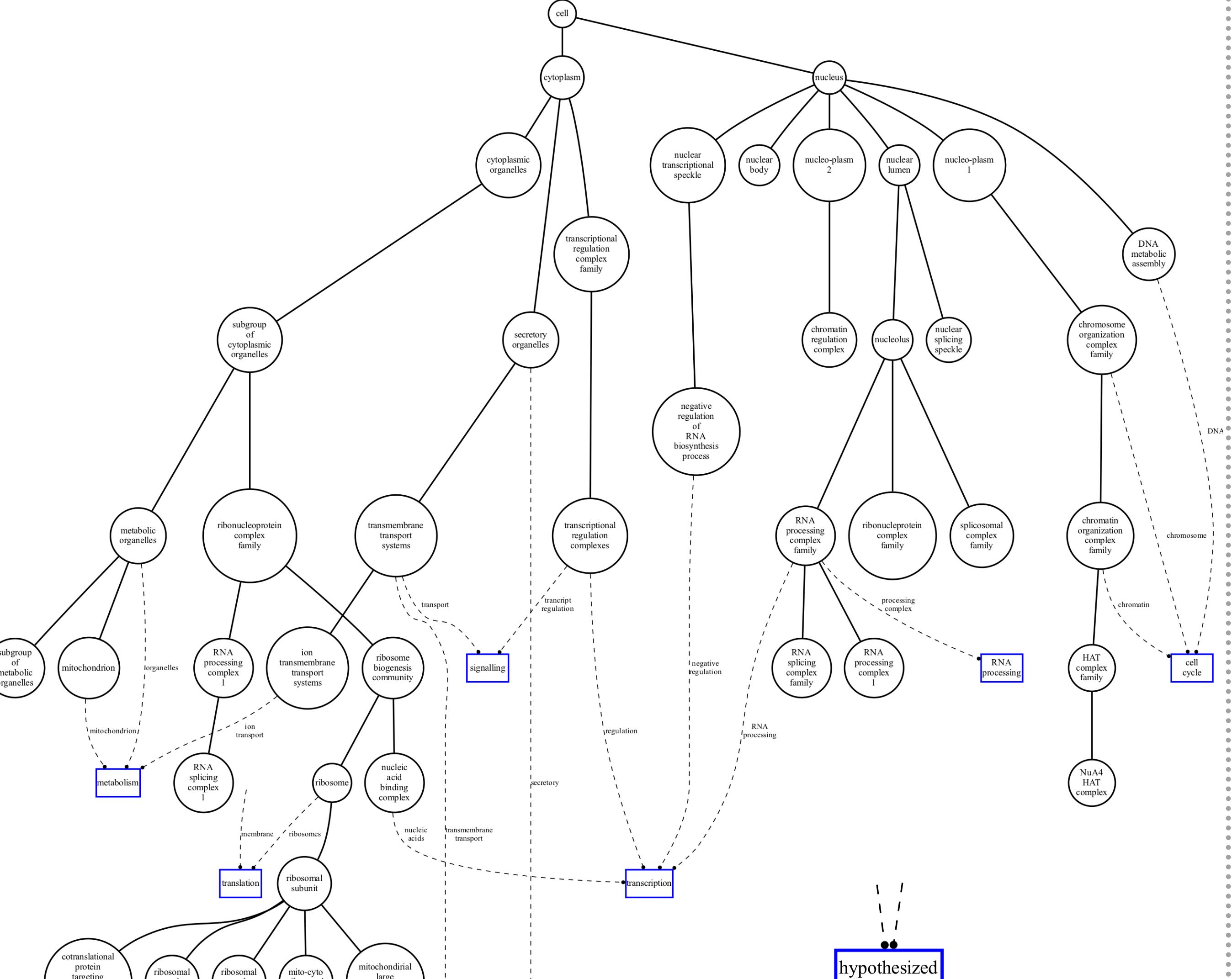
APPLICATION: CUSTOM HYBRID SIMULATIONS

Existing simulation frameworks can be combined into custom hybrid simulators that re-use the best available software in a modular fashion. This provides a foundation for an open-source ecosystem of interchangeable simulation modules that can be flexibly plugged together.



APPLICATION: COMPOSING STRUCTURE WITH FUNCTION

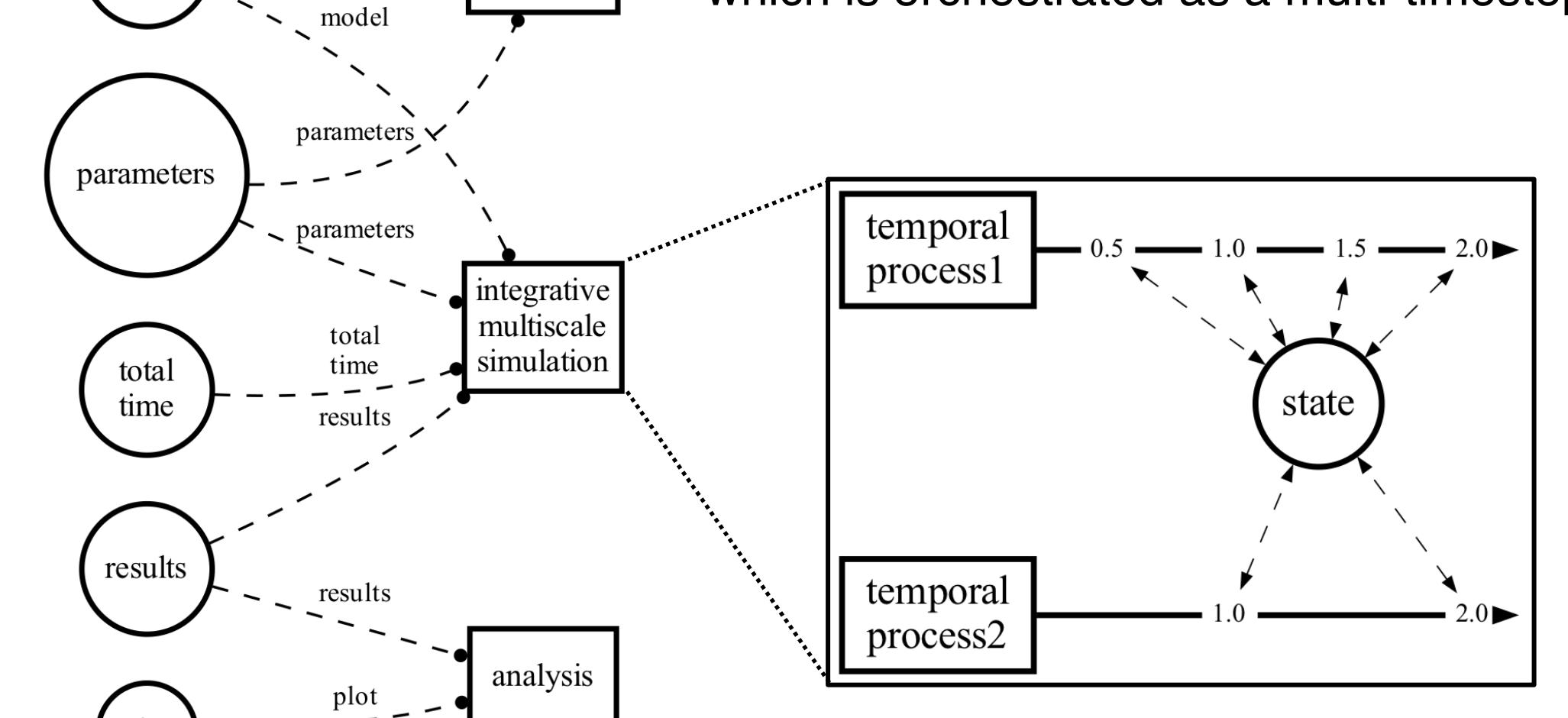
Structural nodes that represent different biological compartments can be wired together with processes that represent biological functions. Process bigraphs makes this flexible, supporting the composition of many different hypothesized mechanisms each with the most appropriate mathematical representation.



*Full model is not yet functional. Multiscale map from Qin et al. “A multi-scale map of cell structure fusing protein images and interactions.” *Nature* (2021)

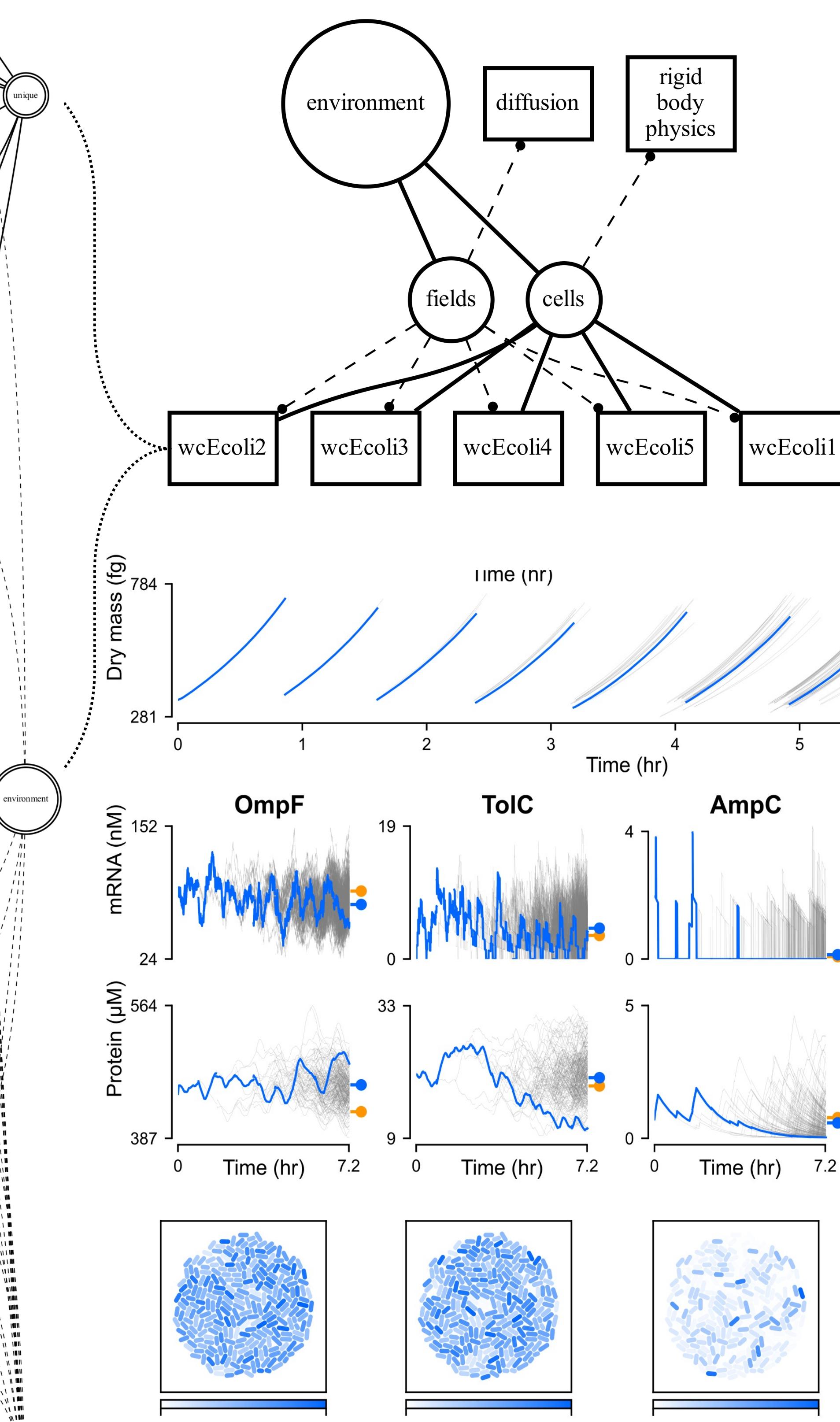
APPLICATION: SIMULATION EXPERIMENT DESIGN

PBS can declare simulation experiments in the style of SED-ML, in which an integrative simulation is just one step. This example simulation experiment is a composition of two orchestration methods: a workflow (left) which includes a step for a full integrative simulation (below), which is orchestrated as a multi-timestep co-simulation.



WHOLE-CELL/WHOLE-COLONY E.COLI MODEL

In this recently completed project, each *E. coli* whole-cell model (wcEcoli) becomes an agent in a spatial, multi-cell simulation [2].



REFERENCES

- Agmon, E., et. Al. 2022. Vivarium: an interface and engine for integrative multiscale modeling in computational biology. *Bioinformatics*.
- Skalnik, C.S., Cheah, S., Yang, M.M., et al., 2023. Whole-cell modeling of *E. coli* colonies enables quantification of single-cell heterogeneity in antibiotic responses. *To appear in PLoS CB*.
- PBS is realized through three software libraries:
 - bigraph-schema (<https://github.com/vivarium-collective/bigraph-schema>)
 - process-bigraph (<https://github.com/vivarium-collective/process-bigraph>)
 - bigraph-viz (<https://github.com/vivarium-collective/bigraph-viz>)

CONTACT

Email: eagmon@uchc.edu