

VCell: an Open Framework for Computational Cell Biology

James C. Schaff^{1,*}, Ion I. Moraru¹, Leslie M. Loew¹.

¹Center for Cell Analysis and Modeling, University of Connecticut Health Center
*email: schaff@neuron.uhc.edu

The Virtual Cell (*VCell*; <http://vcell.org>) is a unique software environment for computational cell biological research developed at the Richard D. Berlin Center for Cell Analysis and Modeling (CCAM) at the University of Connecticut Health Center. CCAM is a NIH Technology Center for Networks and Pathways and a NIH-designated National Research Resource. Current capabilities include modeling and simulation of biochemical pathways coupled with electrophysiology, membrane transport, and diffusion/advection within either spatial or compartmental contexts and with deterministic and stochastic formulations. The VCell software supports a workflow for modeling and simulation which abstracts unnecessary complexity of the underlying math and physics while automatically managing dependencies, constraints and modeling artifacts (e.g. simulation results, generated math models). However, as we continue to support more users, enter into important new collaborations, propose new capabilities, and promote external contributions, we recognize the need to introduce changes to the organization of the VCell software that will support this growth.

The entire software is being transformed into a standards-based (OSGi), modular, extensible, open source plug-in architecture. We present the new architecture and the current set of 42 software components (plug-ins) that has been created as a proper decomposition of the previous Virtual Cell code. These plug-ins have been iteratively refined to achieve better separation of concerns, reduced coupling, and better encapsulation of implementation details. For example, the mathematical expression handling and the unit handling capabilities were completely refactored into reusable components that hide their implementations and provide stable, high level interfaces in preparation for publication as open source components. These components also gave us opportunities to define “extension points” (i.e. locations for future components to plug into the framework). The expression handling allows for alternate implementations of the expression handling and a facility to plug in more sophisticated “Symbolic Processors” (computer algebra systems) for advanced services like more powerful expression simplification and symbolic equation solving. Using these patterns the entire modeling and simulation software is being decomposed into reusable components and reassembled into the next-generation, extensible VCell, as well as smaller special-purpose applications (e.g. model-assisted analysis of FRAP experiments). VCell is available at <http://vcell.org> and is supported by NIH P41-RR013186.