



# The ERATO Systems Biology Workbench

A. M. Finney<sup>1,2</sup>, M. Hucka<sup>1,2</sup>, H. M. Sauro<sup>1,2</sup>, H. Bolouri<sup>1,2,3,4</sup>, J. Doyle<sup>1,2</sup>, H. Kitano<sup>1,2,5</sup>

1. ERATO Kitano Symbiotic Systems Project, M-31 Suite 6A 6-31-15 Jingumae Shibuya-ku, Tokyo 150-0001, Japan

2. Control and Dynamical Systems 107-81, Caltech, CA 91125, USA 3. Science and Technology Research Centre, U. of Hertfordshire, AL10 9AB, UK

4. Division of Biology 216-76, Caltech, CA91125, USA 5. Sony Computer Science Labs, 3-14-13 Higashi-gotanda Shinagawa-ku, Tokyo 141-0022, Japan

## Introduction

The ERATO Systems Biology Workbench (SBW) is an extensible software framework enabling the creation of an integrated, easy-to-use software environment that allows the sharing of models and resources between simulation and analysis tools for systems biology. In this poster we summarize the project goals and current status.

## Goals of the SBW Project

Integrate a variety of software tools that implement different approaches to modeling, parameter analysis, and other related tasks

Enable the construction of complex hierarchical multicellular models from arrays of model components

Enable the interaction with biologically-oriented databases containing data, models and other relevant information

## Principles of the SBW project

Be open source: use a variant of the GNU library license

Be portable to Windows and Linux

Use current and emerging standards such as Python 2 and the Systems Biology Markup Language (SBML)

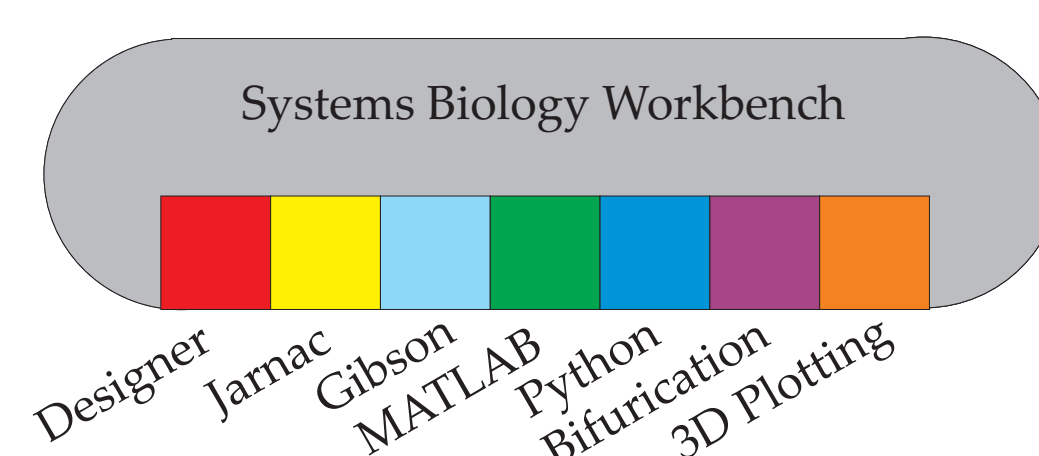
Ensure tools are easy to use

Reduce the time spent by developers both creating software infrastructure and creating tools that exist in a similar form in other packages, allowing developers to concentrate on the development of new algorithms and models

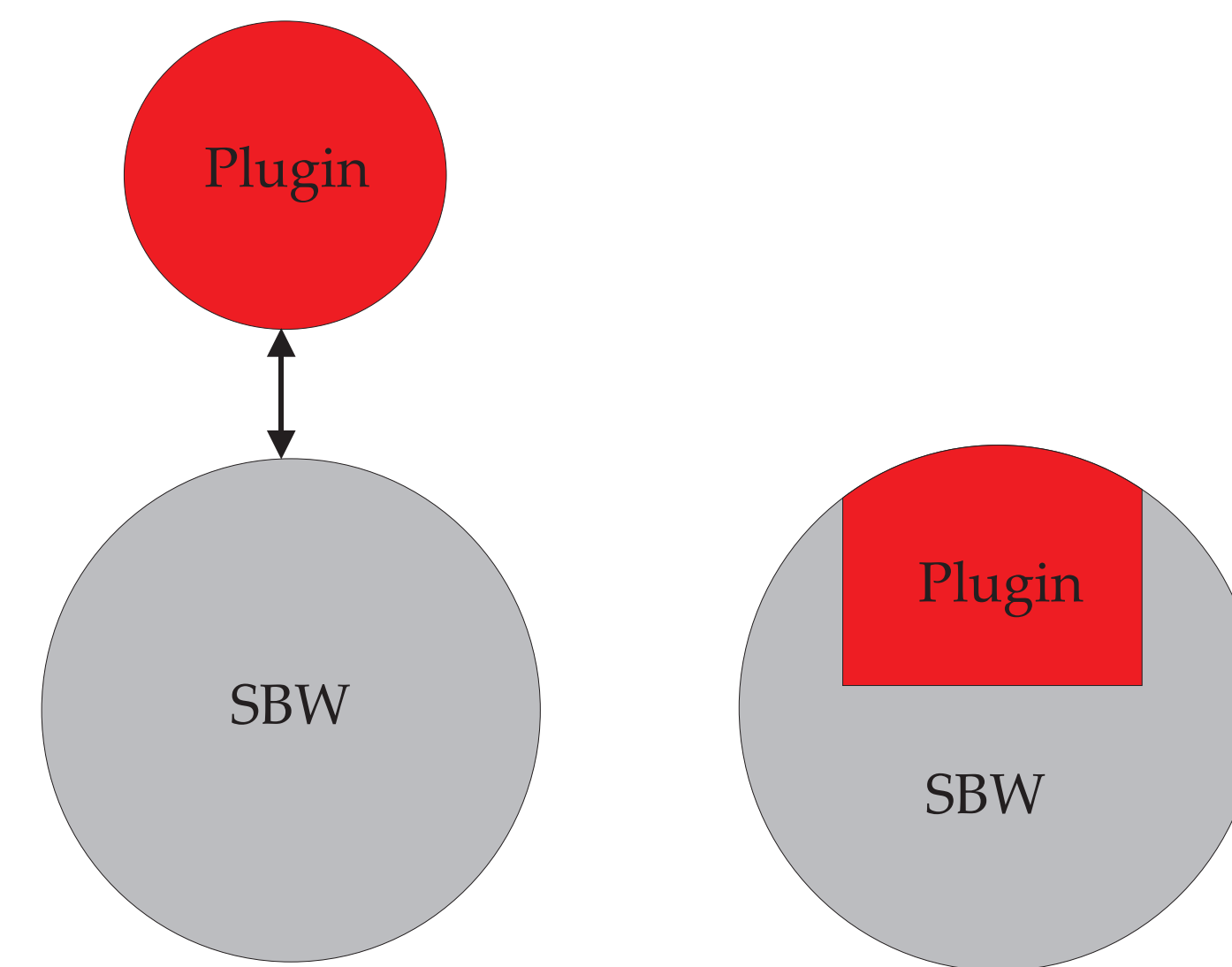
Use the workbench as a vehicle for collaboration between developers of bioinformatics technology

## Architecture

To facilitate modularization and extensibility, SBW is a software framework that enables separately compiled modules or plug-ins to be combined to form a functioning application. Currently, plug-ins can either be applications, dynamic libraries or Java JAR files. Plug-ins can be created in several programming languages. SBW currently employs components written in Delphi and Java, and we have written plug-ins in C++. We plan to create bridges to other languages including Python.



Intercommunication between plugins is performed by a simple message passing mechanism. Plugins are registered with a directory. This directory indexes plugins by category. A category groups together plugins with a common interface.



Plug-ins can either be in separate processes from SBW or be part of the main SBW process.

Currently the most important category of plug-in is "Analysis". Plug-ins in this category perform an arbitrary function given a model. An example of a plug-in in this category is Jarnac, a tool written by Herbert Sauro that performs time-based simulation, steady-state analysis and Metabolic Control Analysis.

## Systems Biology Markup Language

A key feature of SBW is the use of Systems Biology Markup Language (SBML) to communicate models between plug-ins. Plug-ins in the "Analysis" category receive models in SBML form. SBML was and continues to be developed in close collaboration with the teams developing BioSpice, DBSolve, E-Cell, Gepasi, Jarnac, VCell and StochSim as well as members of the ERATO Kitano team. SBML is being developed in several stages. One stage has been completed and a second stage is planned.

## SBML Level 1

This level was frozen on 1st March 2001. It enables biochemical networks to be described as consisting of species (e.g. molecules), reactions (with rate equations) and compartments. In addition, mathematical rules linking the properties of these components can be specified.

## SBML Level 2

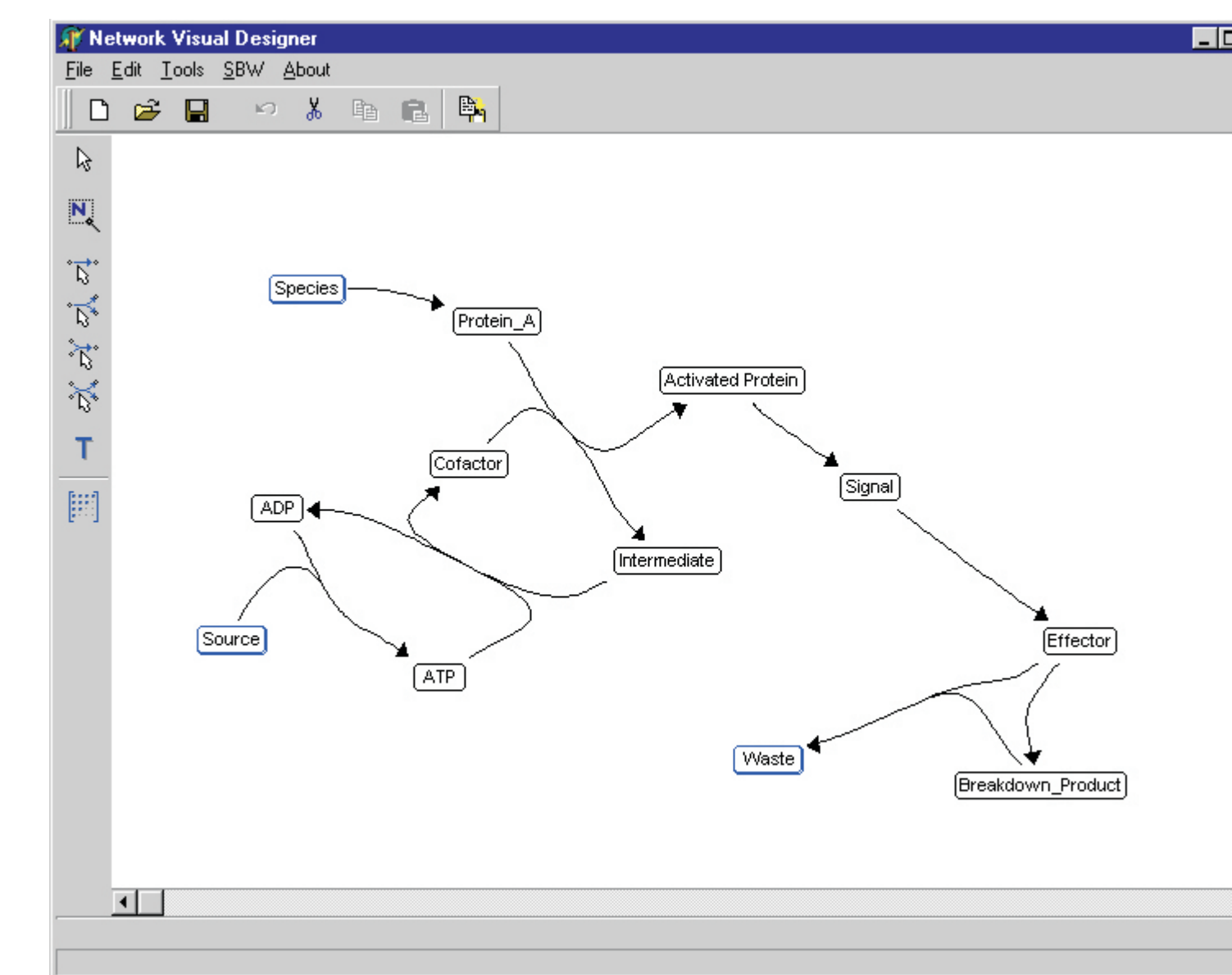
We expect this level will extend level 1 to enable complex models, including multicellular and gene regulatory models, to be constructed hierarchically from submodels and arrays of components. Compartments will probably have a simple spatial attributes. Models will probably be able to specify rules of connection between compartments.

## Plug-ins

The following applications and libraries will be adapted to be plugins for SBW in three phases of the SBW project. We will write adaptors that allow the following applications and libraries to be interfaced to the SBW framework. The workbench is in the first phase of development.

## Phase 1

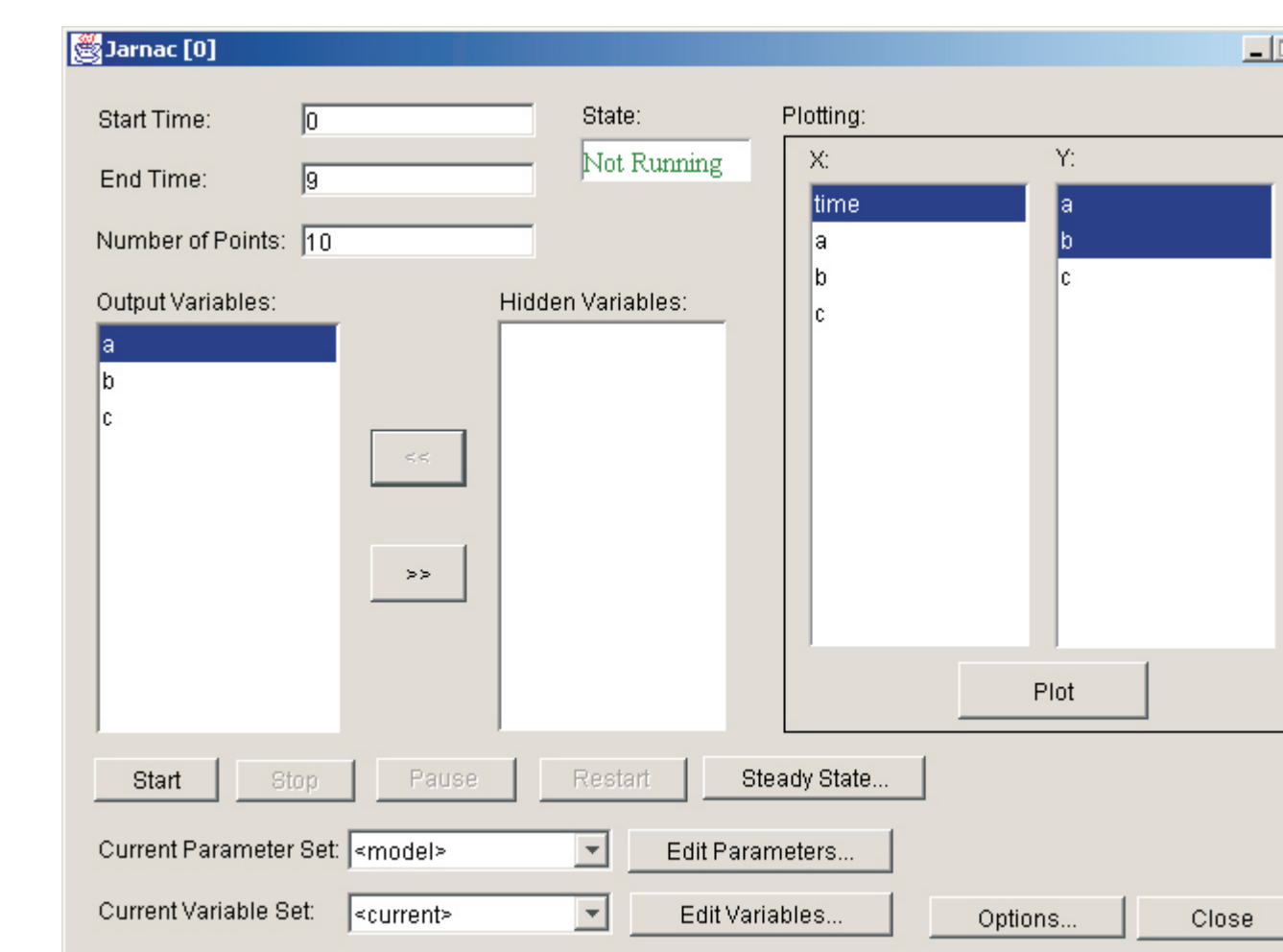
*Designer*



This plug-in provides a GUI for drawing biochemical pathways. The Designer exports the model in SBML format. The SBW integration allows the user to select an analysis to be invoked on the model.

*Jarnac*

This plug-in provides the following ODE-based analyses: Time-Based simulation, Steady state analysis and Metabolic Control Analysis.



The above GUI provided by the Jarnac integration allows the user to access a subset of the functionality in Jarnac.

*Gibson*

This plug-in will provide stochastic time-based simulation. This stochastic simulator which employs a performance enhanced version of the Gillespie algorithm written by Michael Gibson. This plug-in will employ a GUI very similar to that used for the Jarnac integration above.

*MATLAB*

This plug-in will translate an SBML model into ODEs for export into MATLAB.

*Python*

This plug-in will allow a user to control SBW via the Python scripting language.

*Optimization*

This allows the fitting of model parameters to experimental data.

## Phase 2

*Bifurcation*

We will integrate a standard bifurcation tool.

*3D Plotting*

We will integrate 3D plotting functionality to support time based simulation and bifurcation analyses.

## Phase 3

*Multi-mode simulation*

We will develop a new time-based simulation component that will enable very fast yet accurate simulation of biochemical networks. This simulator will combine Stochastic and ODE techniques using novel algorithms developed collaboration with leading researchers in the field.

*Transparent database access*

We will develop a several plugins to enable the user to download models from bioinformatics databases.

## Modular and Multi-cellular models

As part of phase 3 we will enable SBML Level 2 support by:

Extending and developing model capture tools to support hierarchical models

Developing a tool that translates SBML Level 1 from Level 2 to enable existing analyses to be applied to Level 2 models.

## Packaging

We plan to create a Windows installation as part of phase 1 and a Linux installation as part of phase 2. These packages will include plug-ins as they become available and tutorials with working models supplemented later by online documentation. We plan to set up web resources appropriate to open source development, including a public bug database.

## Acknowledgments

This project is funded by ERATO Kitano. Major plug-ins of the current SBW (Designer and Jarnac), which pre-existed this project, were developed independently by Herbert Sauro.

The System Biology Workbench - <http://www.cds.caltech.edu/erato/>  
ERATO Kitano Project - <http://www.symbio.jst.go.jp/>  
Jarnac - <http://members.tripod.co.uk/sauro/biotech.htm>