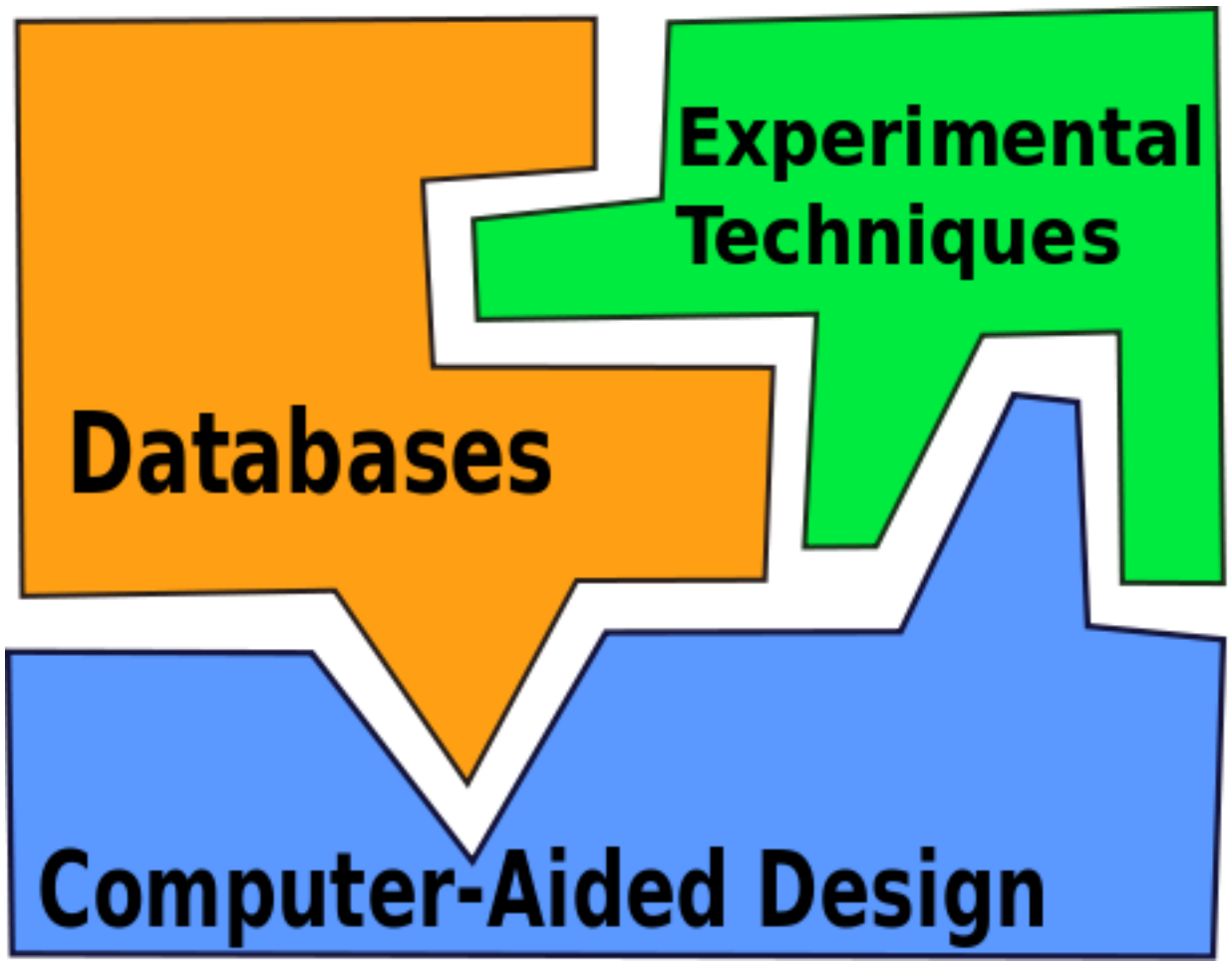
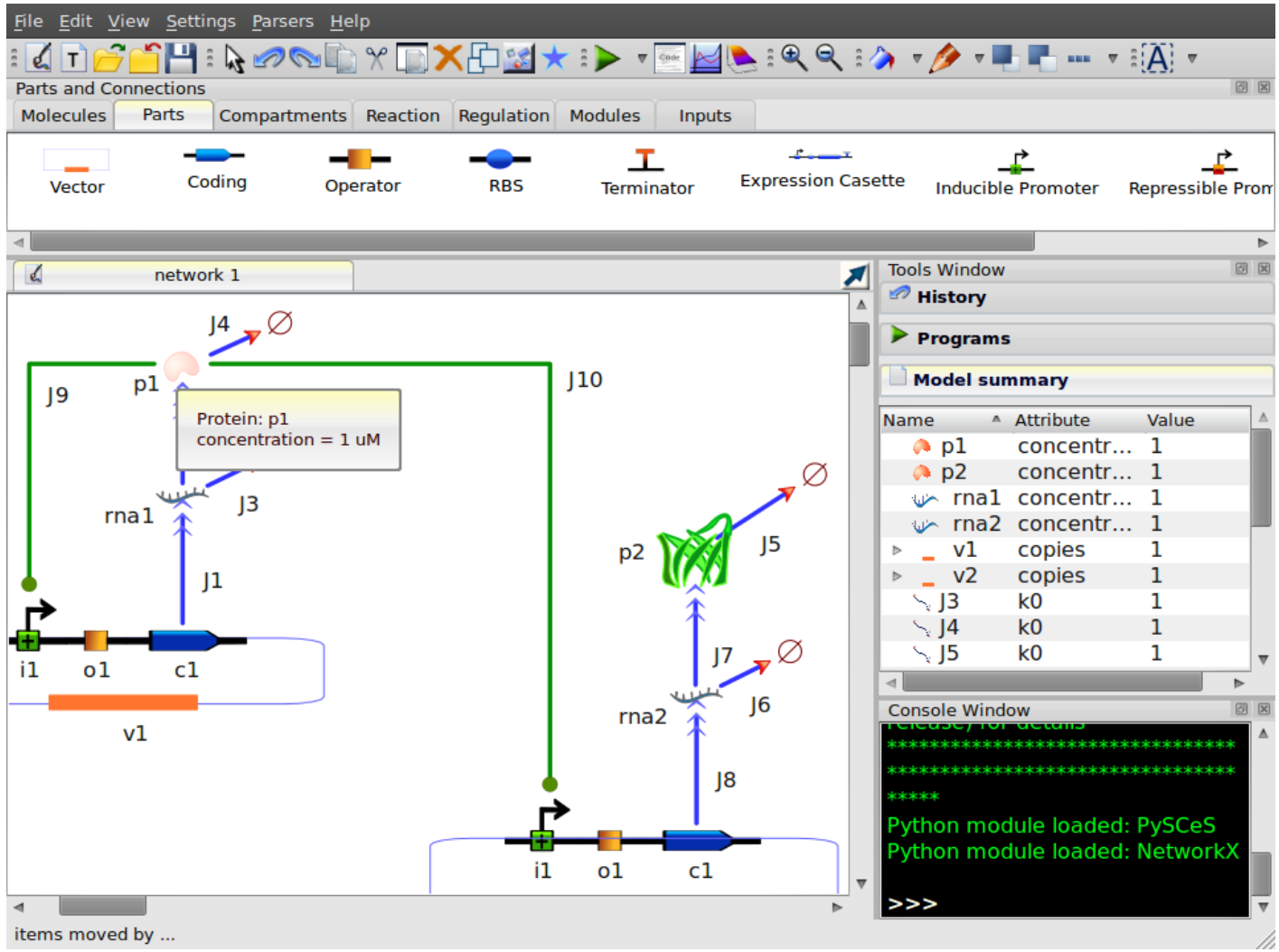
What is TinkerCell?

By Deepak Chandran

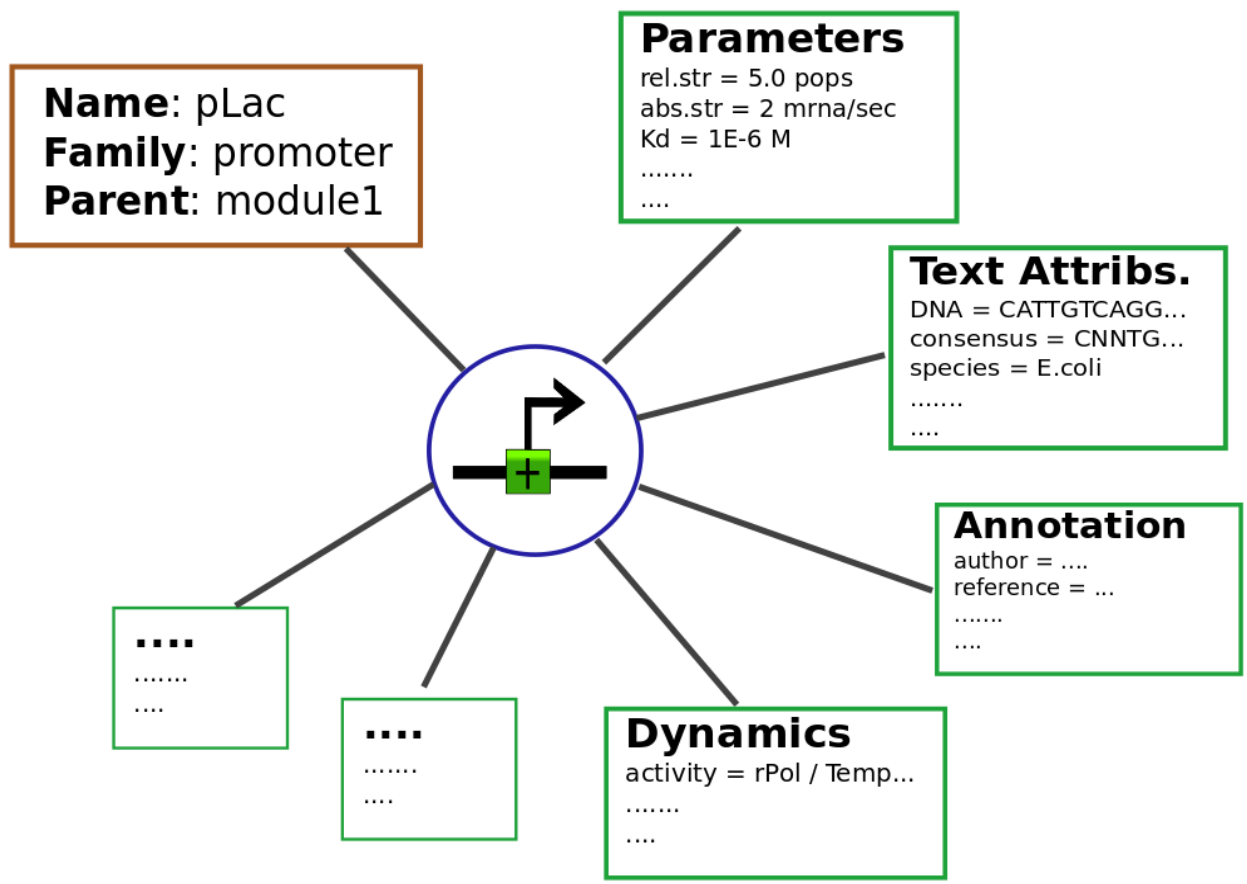
The purpose of TinkerCell is to serve as a Computer-Aided Design application for engineering biological cells.



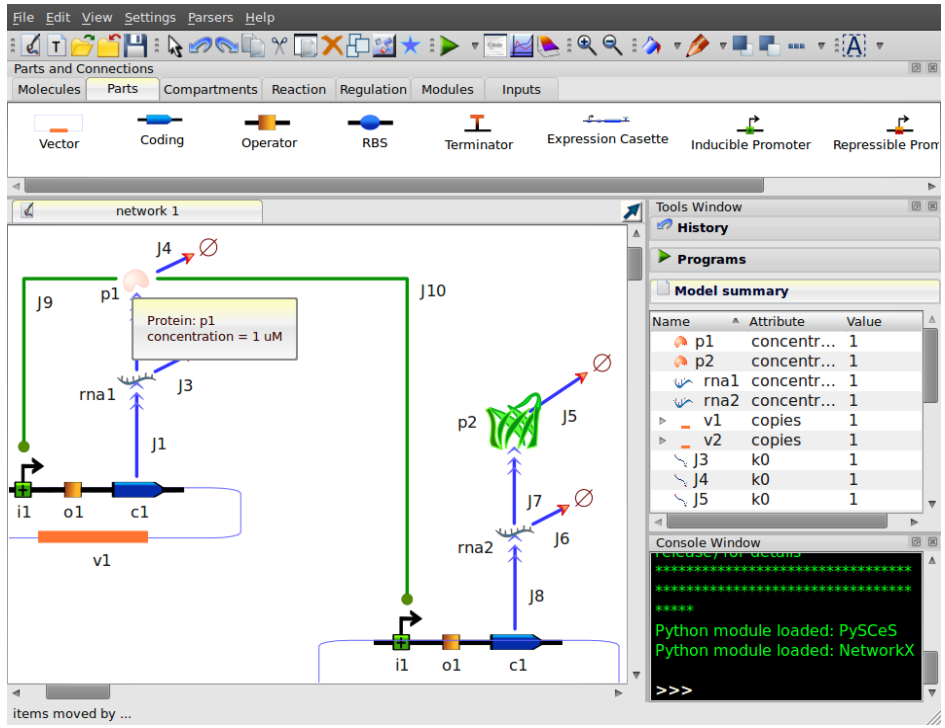
TinkerCell is a program for drawing highly descriptive network diagrams



Each component in the TinkerCell diagram contains information associated with that item



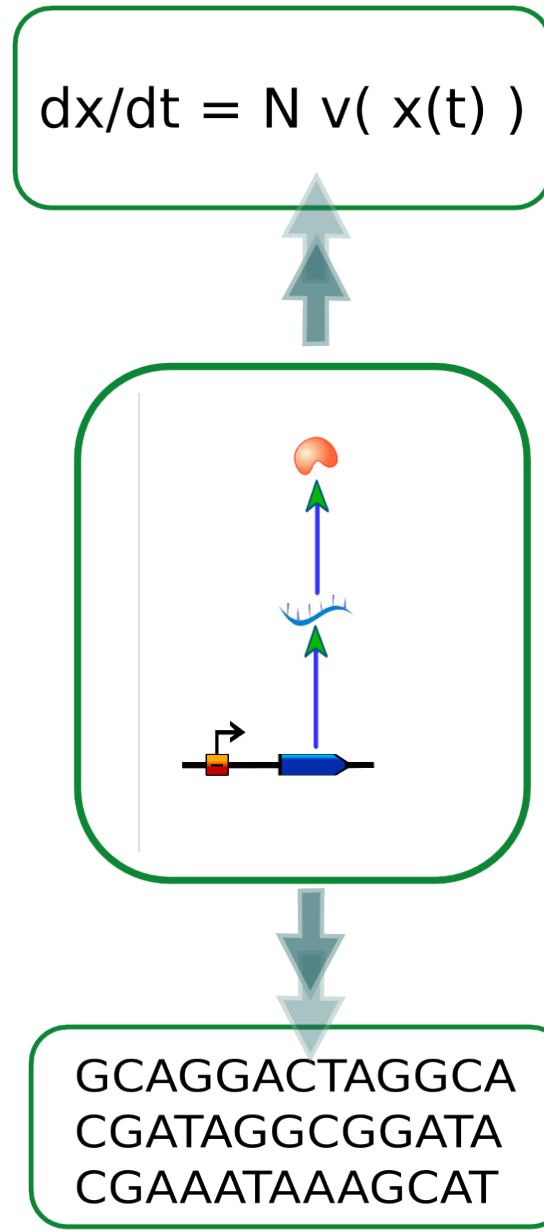
TinkerCell "plugins" convert information in the diagrams to different outputs, such as mathematical models



mathematical model

biological parts

?



Mathematical models

plugins

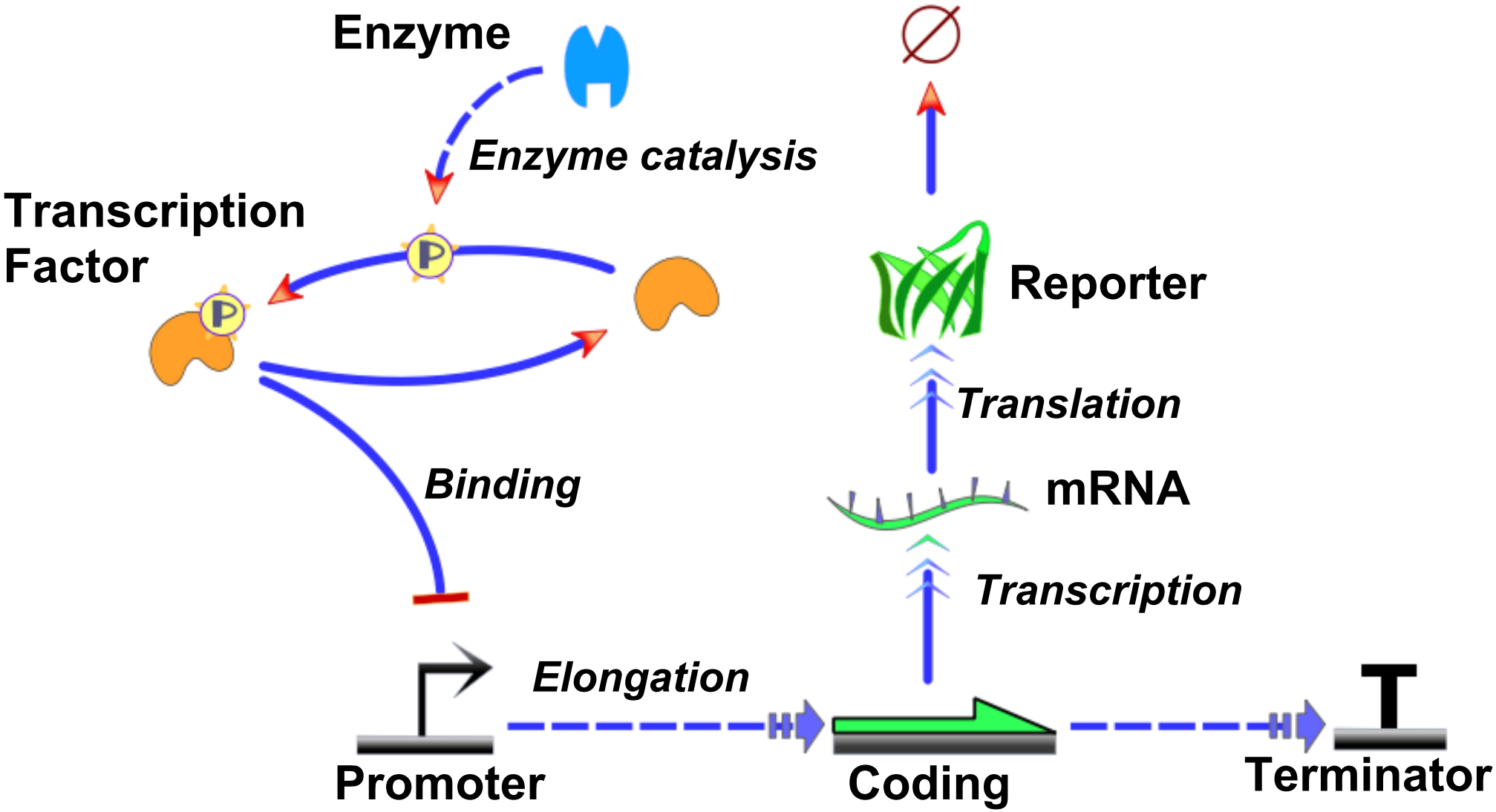
Biological concepts

TinkerCell diagrams

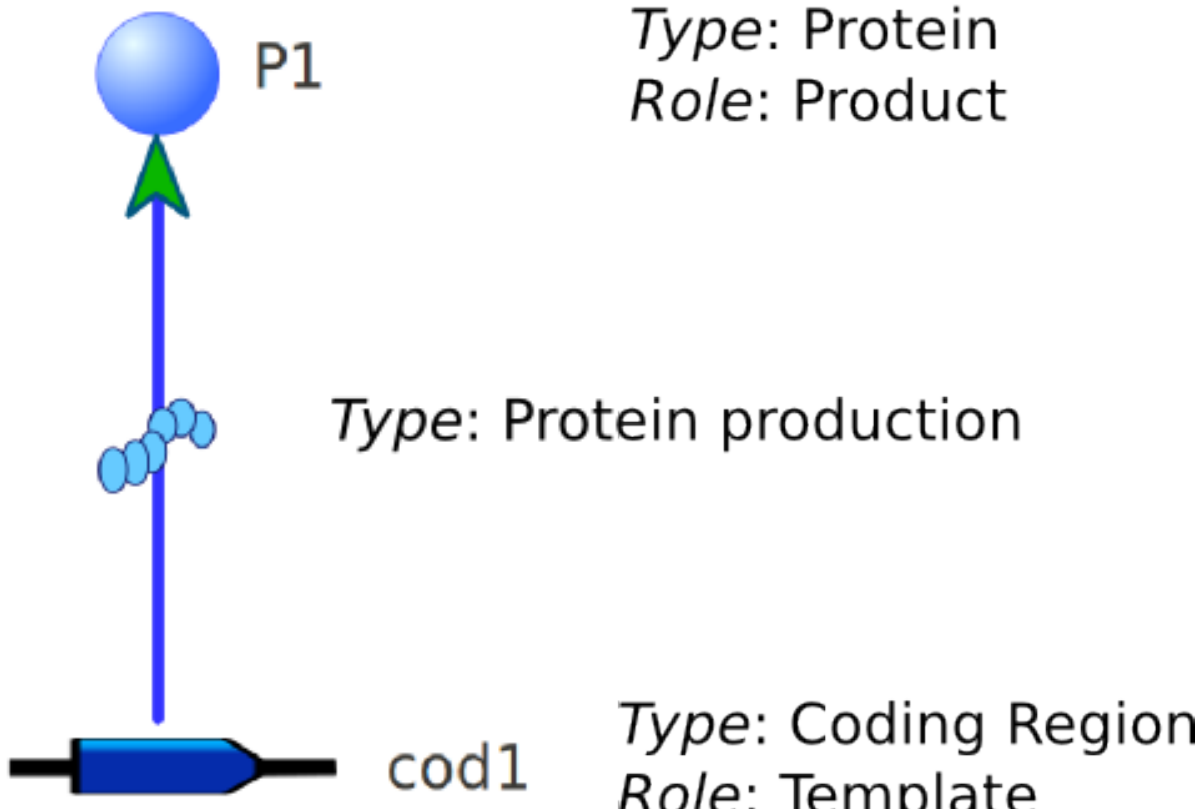
plugins

DNA code

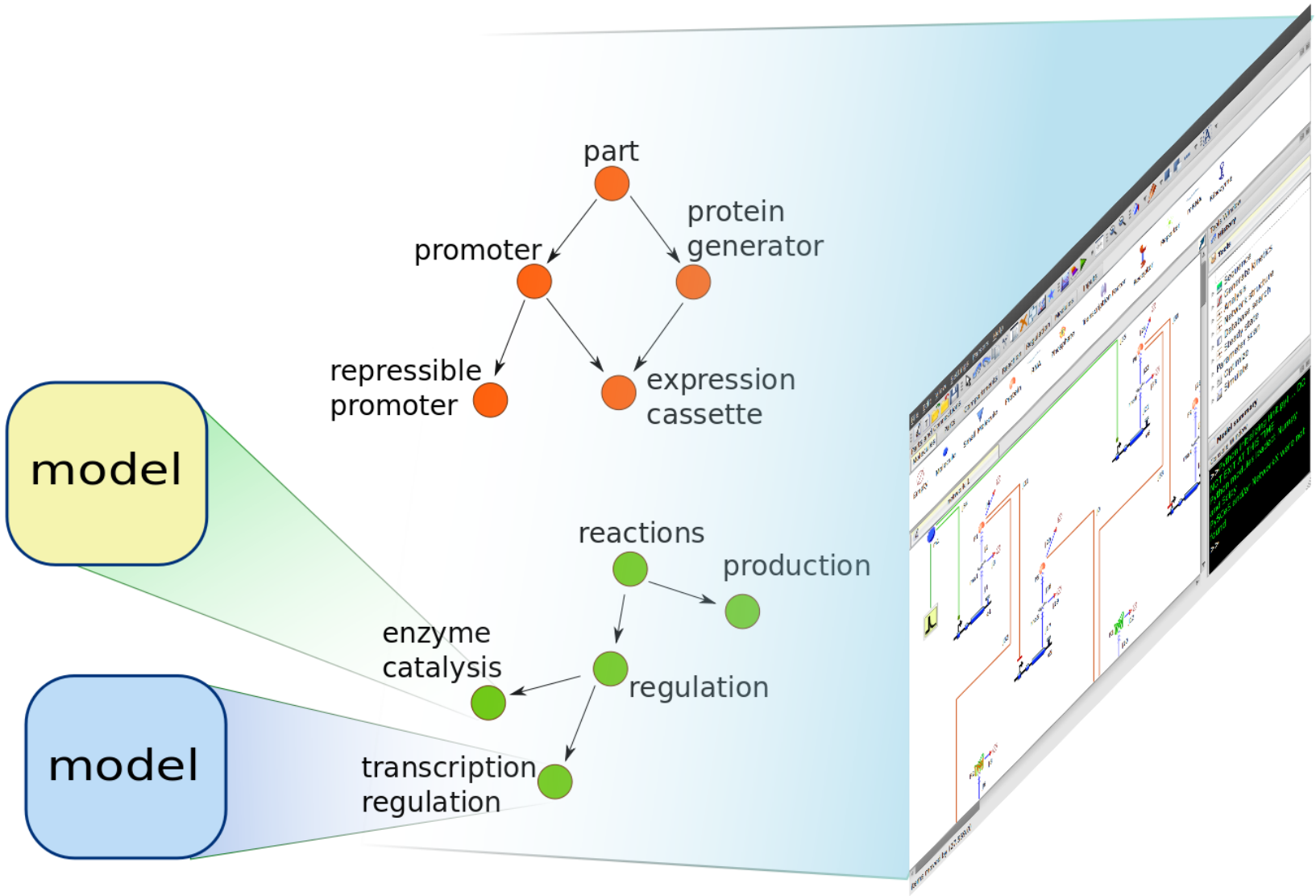
Therefore the whole diagram is semantically annotated, which allows plug-ins to interpret the diagram correctly.



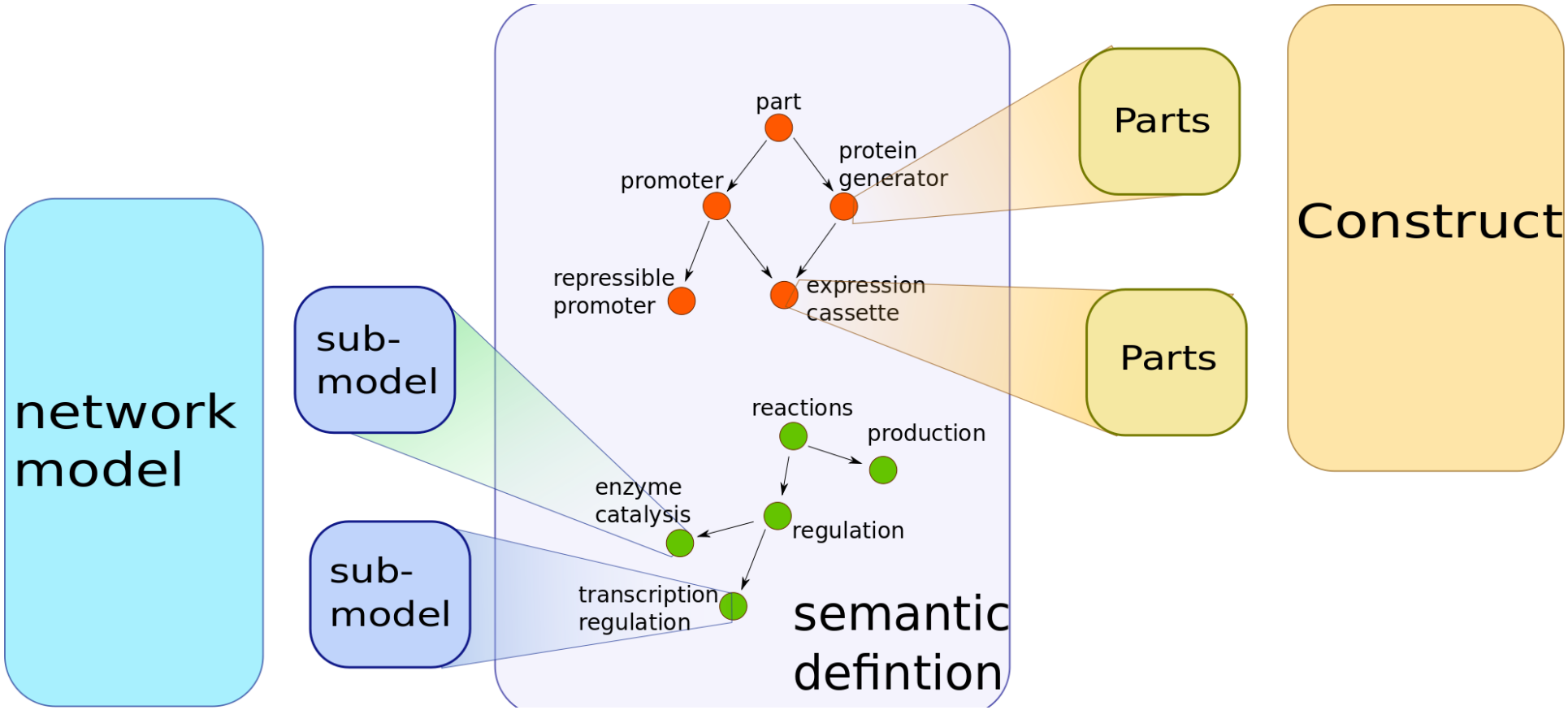
Each connection is *semantically* annotated



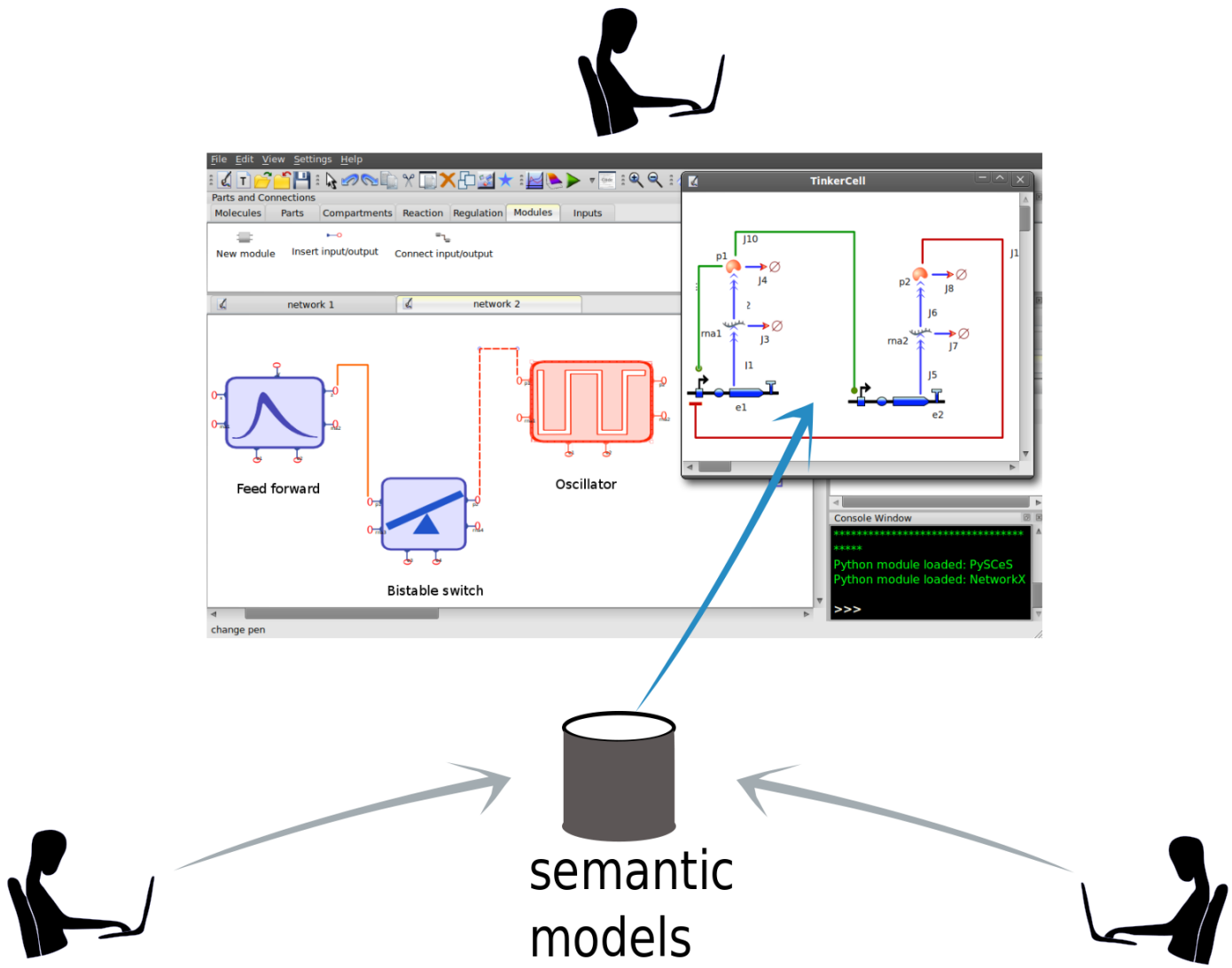
A TinkerCell model can be composed of sub-models, or modules, which are automatically connected via the semantic descriptions



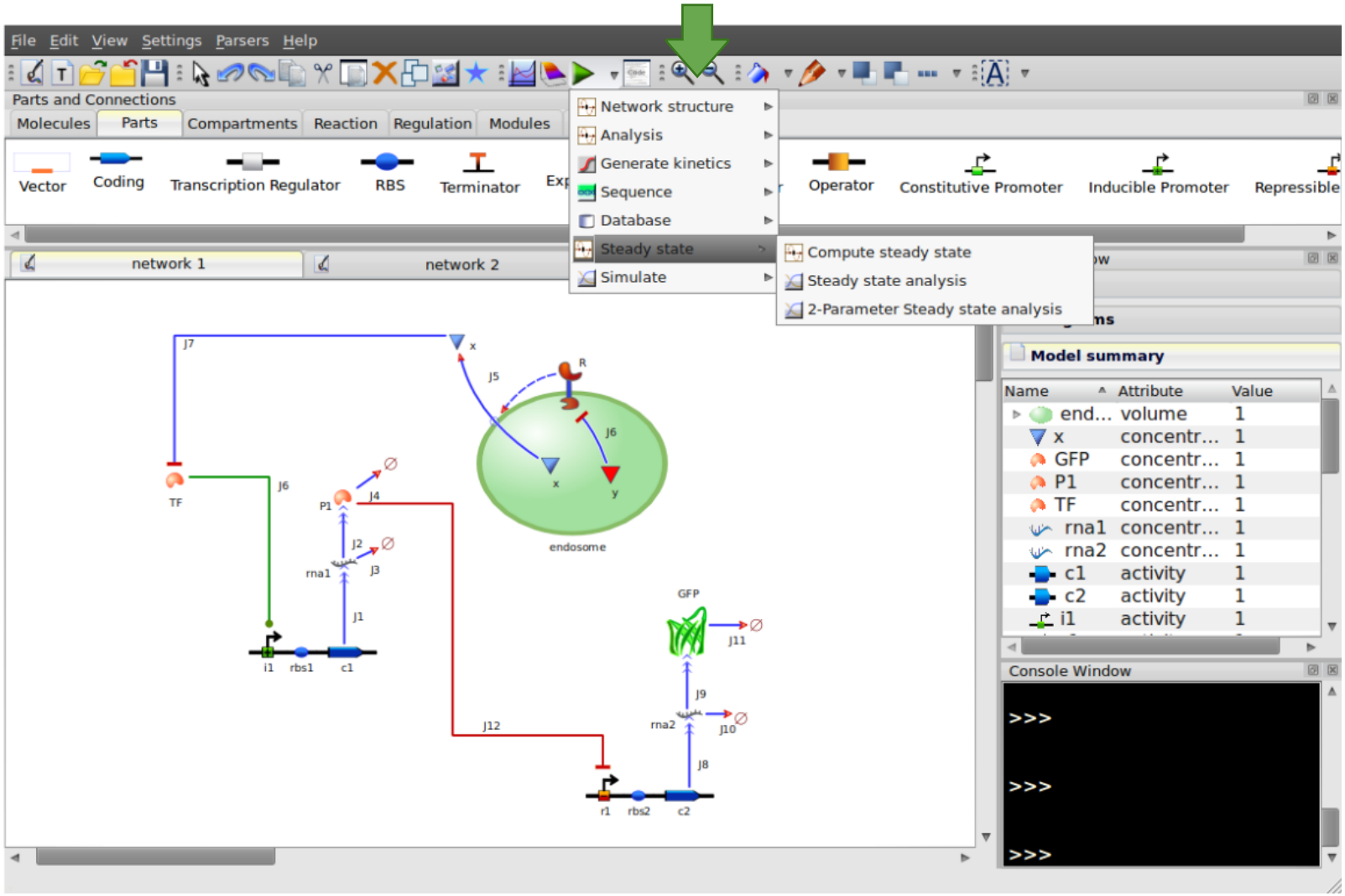
Because TinkerCell models capture the conceptual diagram, it can potentially act as a bridge between the mathematical models and the physical constructs



Semantically annotated modules allow users to construct and share modules, which can be used for collaborative design



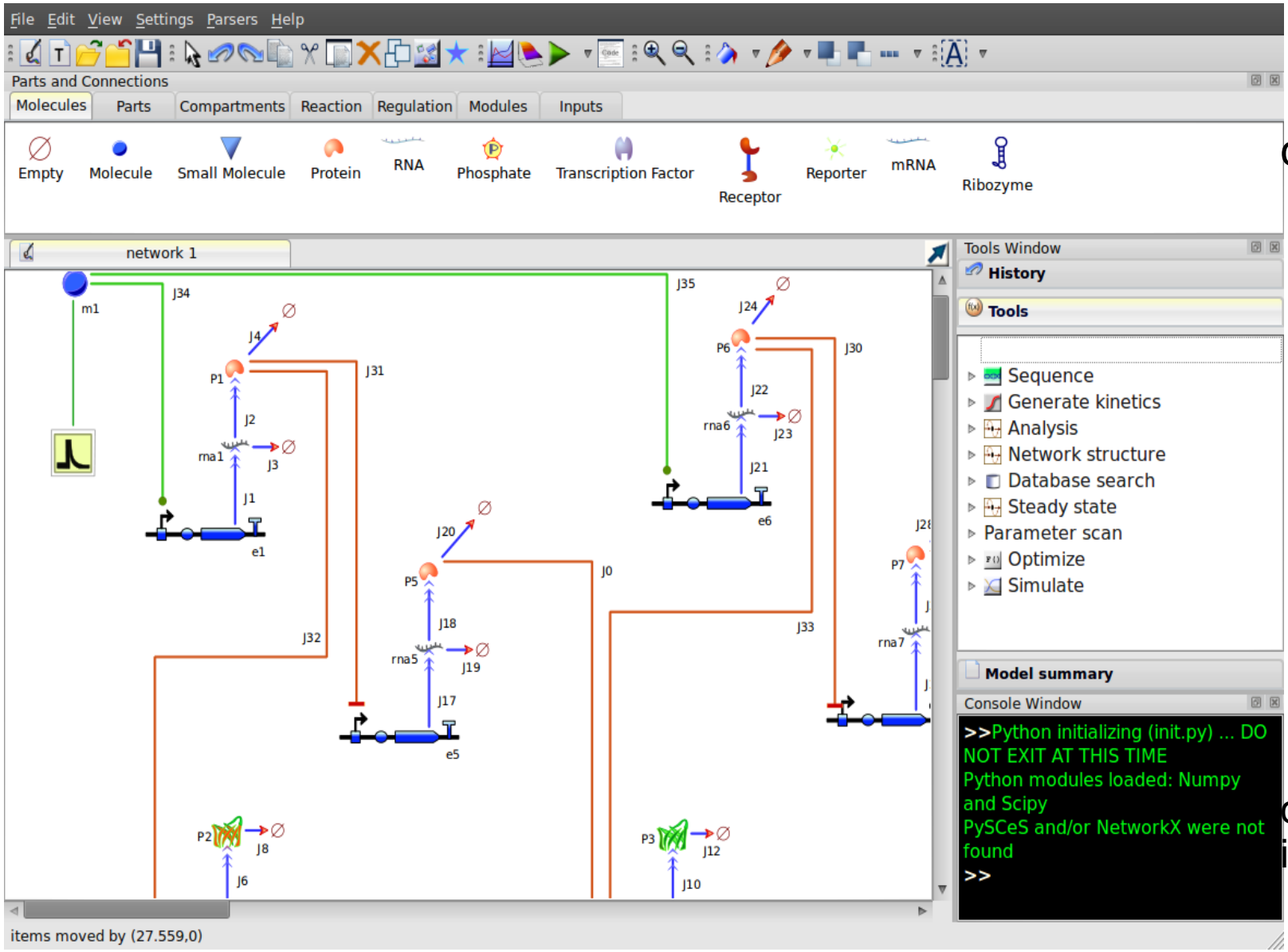
TinkerCell contain several analysis programs, such as simulation or flux-balance analysis. Users can add new programs (current support for C, Octave, and Python languages)



The Basic Layout

using default plugins

toolbar for coloring, annotating, scripting, aligning, etc.



components

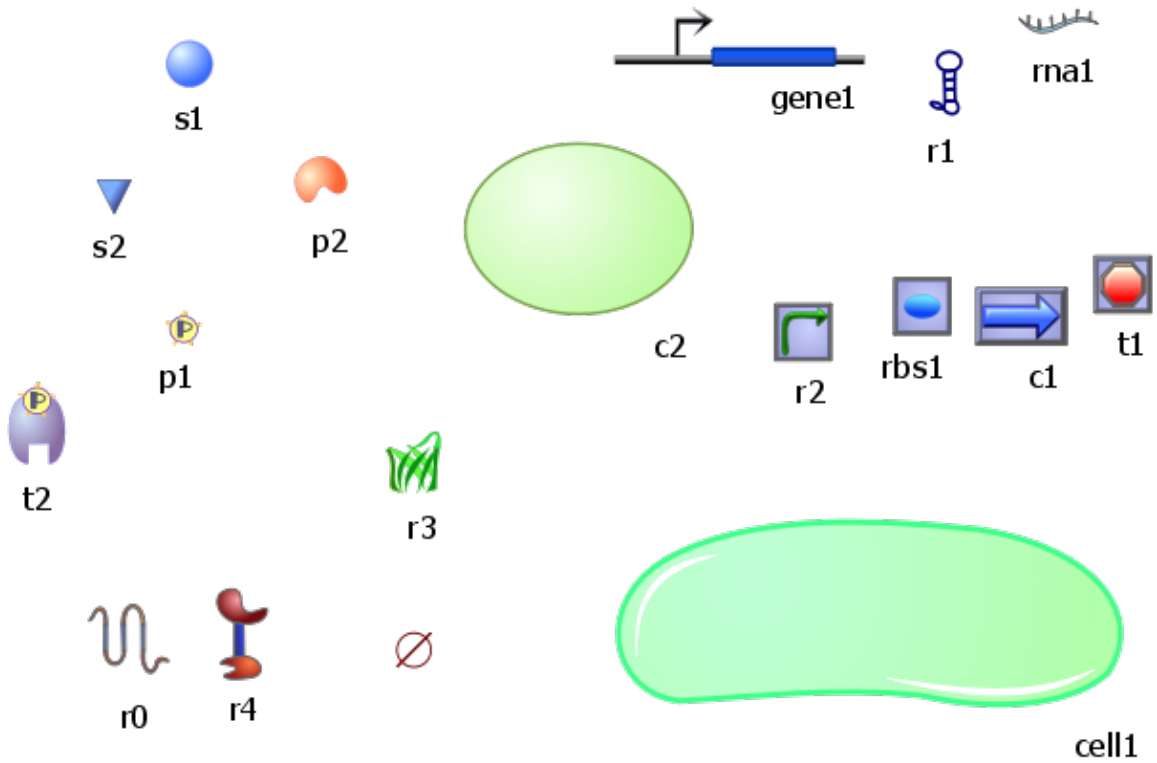
history

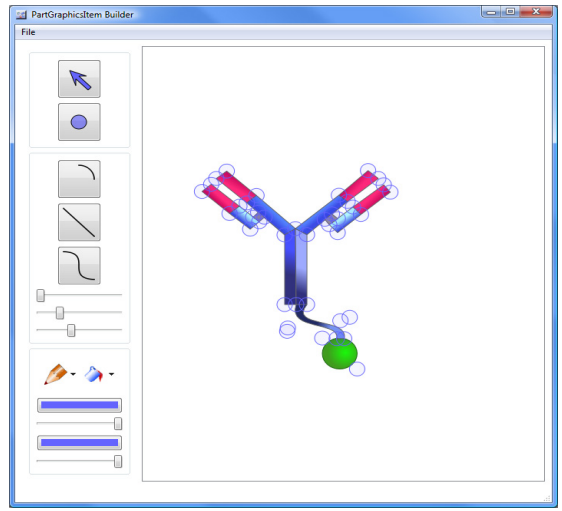
functions

parameters

command-line input

TinkerCell uses a flexible visual representation



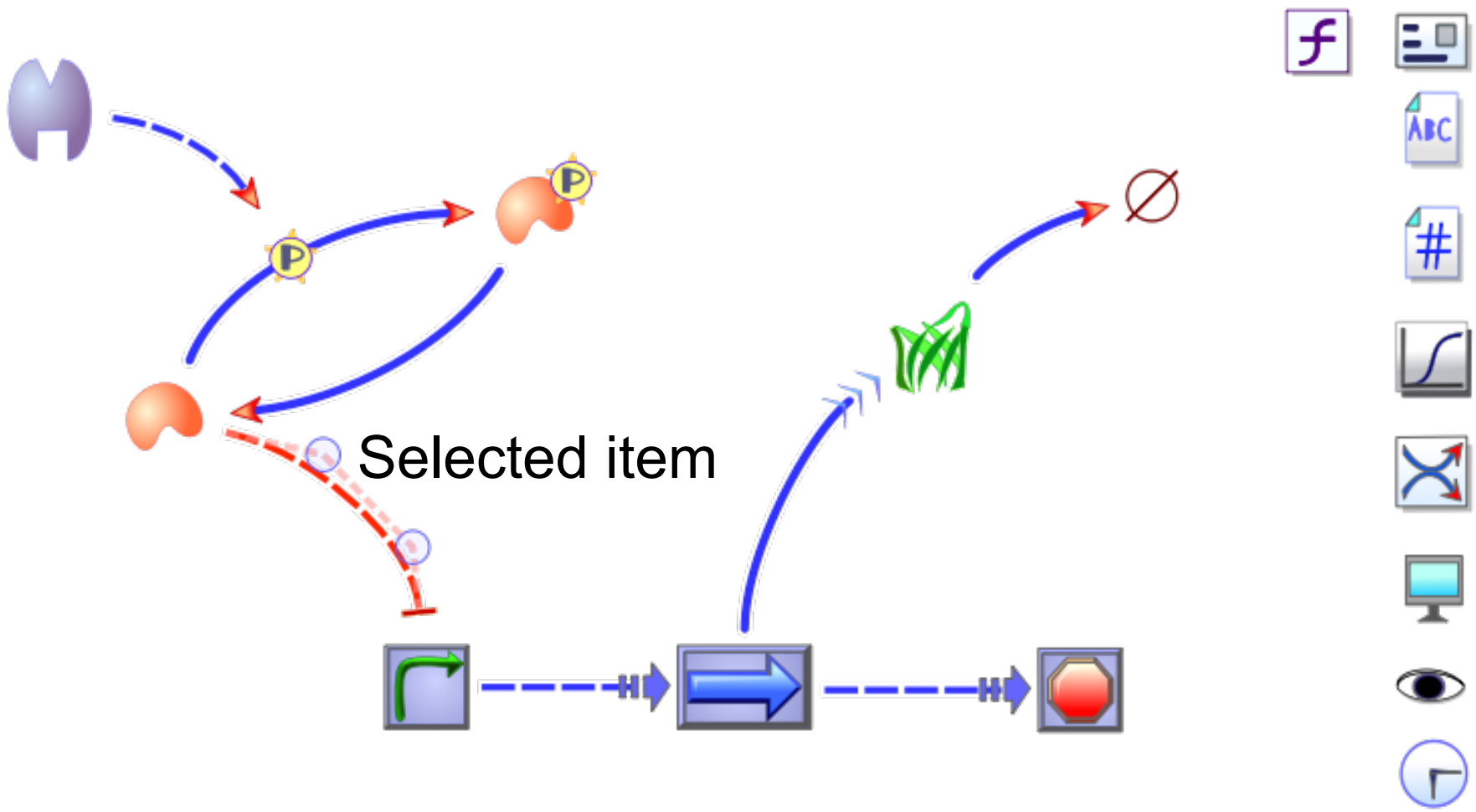


A custom polygon drawing program is used to generate the parts in SBGN render extension format.

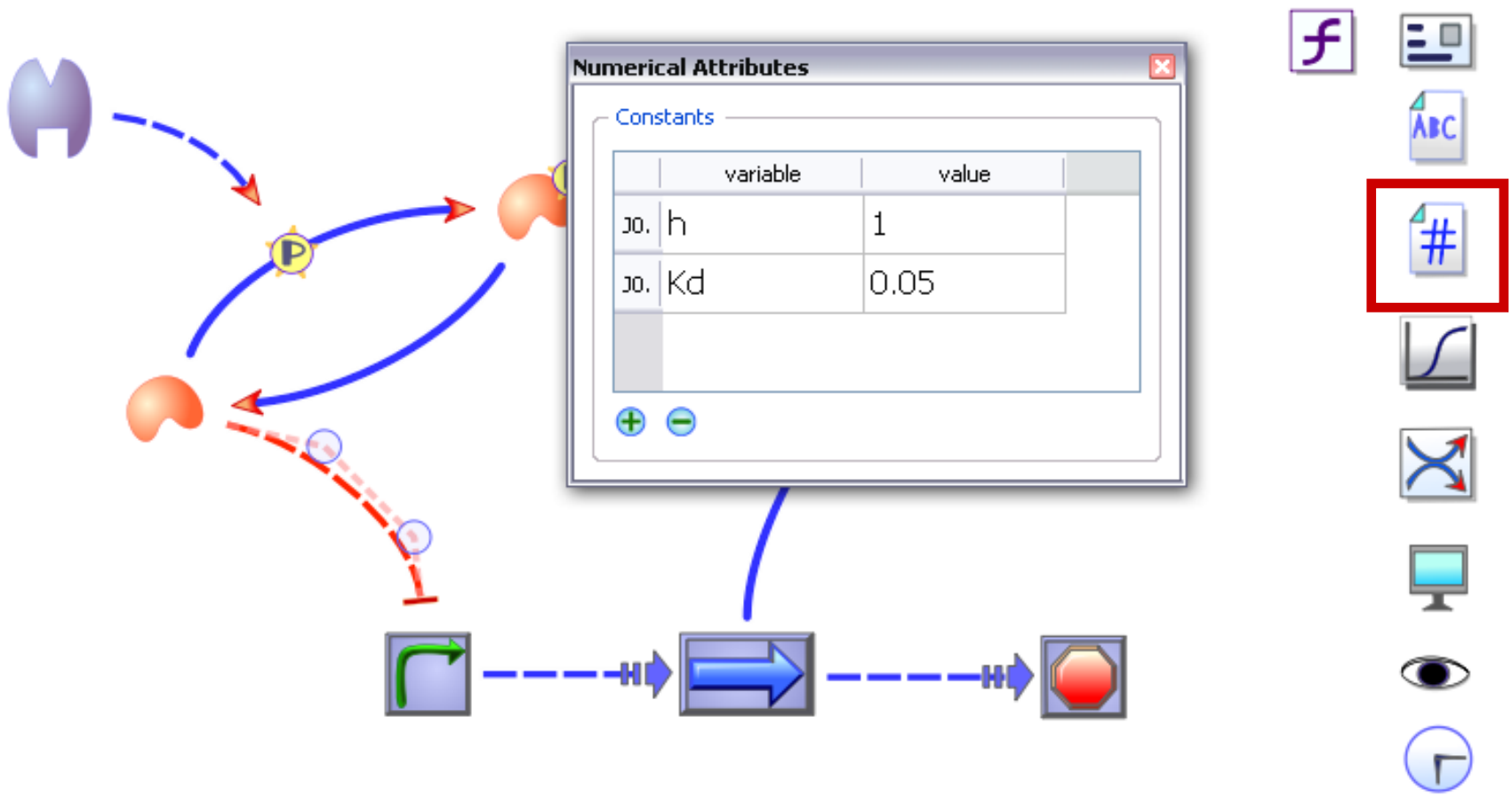
...and users can add more

TinkerCell tries to display all the options and information visually, such as the set of "tools" shown to the right

Tools associated with selected item(s)



Parameters and other attributes are **local**, i.e. a reaction rate is an attribute of the reaction, not the model. This allows better information exchange with databases



Parameters are therefore associated with an item, e.g. J0. Kd is reaction J0's dissociation constant.

Many C and Python packages are included with TinkerCell

Sundials

- Time-course simulation
- Steady state plot

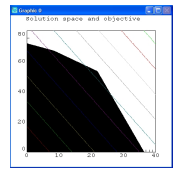
PySCeS
Python Simulator for Cellular Systems

Custom programs

- Gillespie algorithm
- Hill equation derivation
- 2^N automatic binding events
- Loops in Jacobian

PySCeS

- structural analysis
- sensitivity analysis
- bifurcation analysis
- parameter scan
- simulation

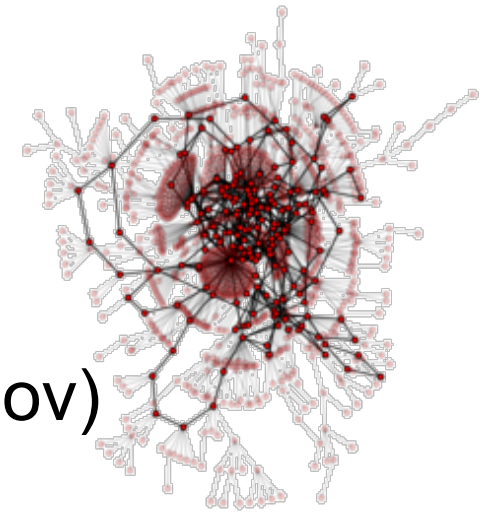


lp_solve

- flux balance analysis

SciPy

- optimization
- matrix operations
- statistics
- numerical methods
- frequency analysis

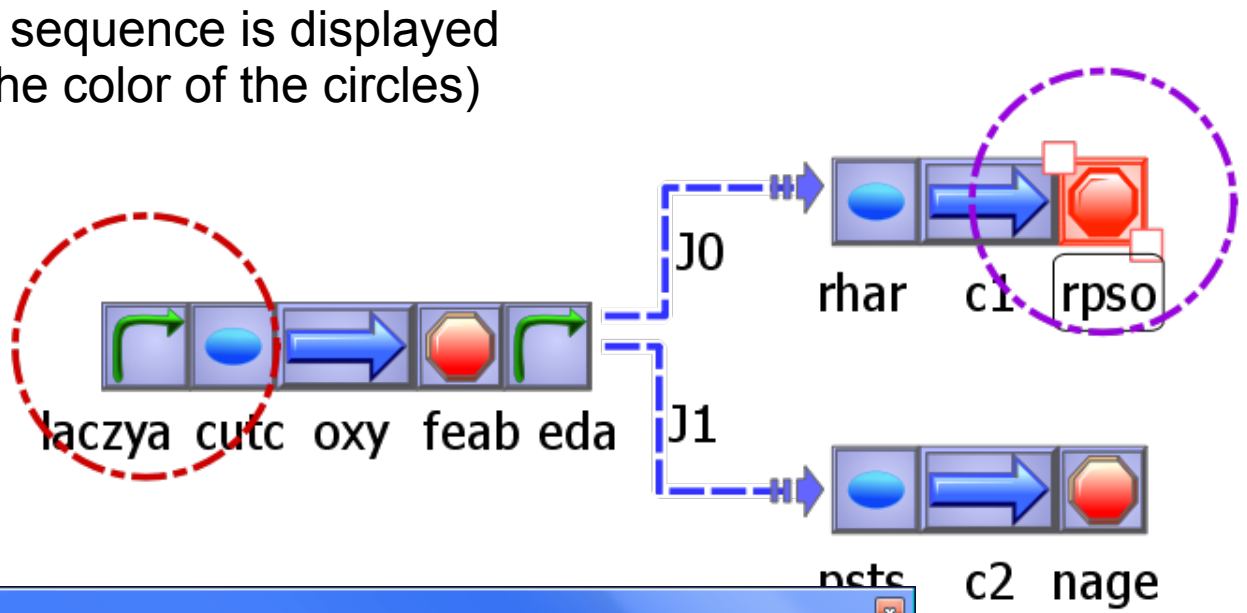


NetworkX (lanl.gov)

- graph analysis
- graph layout

TinkerCell has several small tools, such as sequence viewer

the circles items are the first and last items whose sequence is displayed
(note the color of the circles)



From a programmer's perspective, TinkerCell is highly extensible. It is designed in layers, where each layer is extensible.