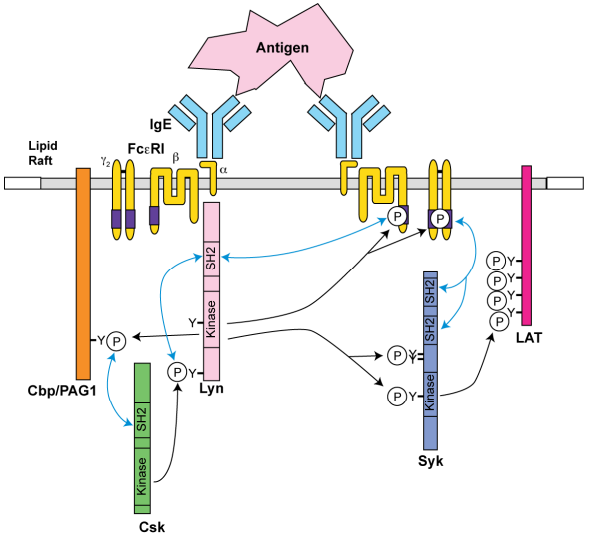
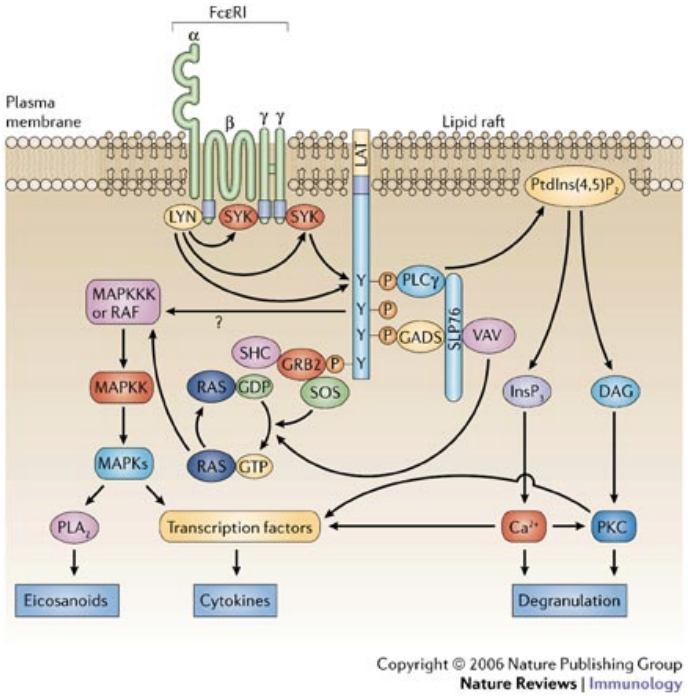
Introduction to rule-based modeling with BioNetGen and RuleBender

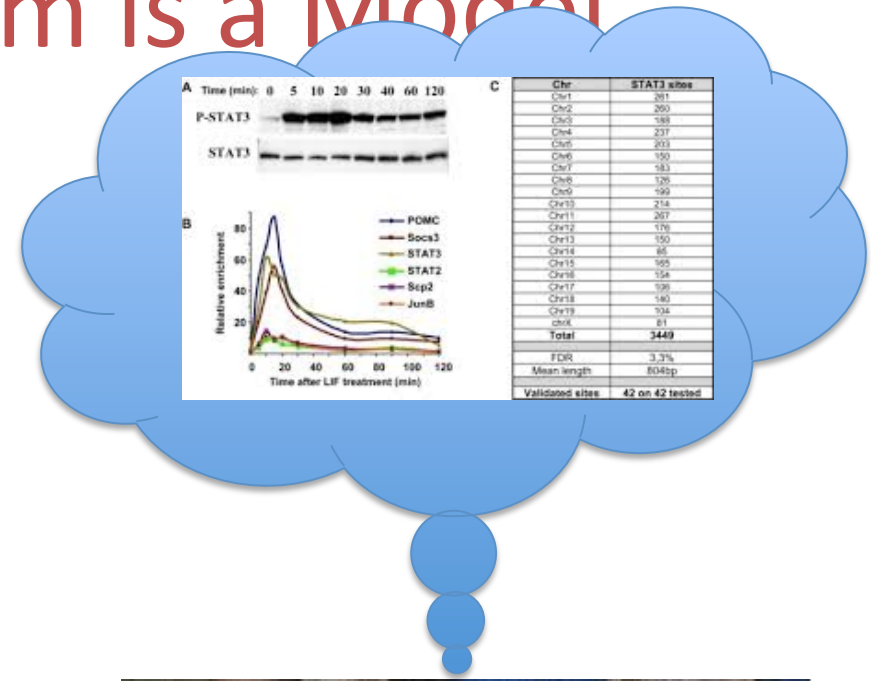
Jim Faeder
Department of Computational and Systems Biology
University of Pittsburgh School of Medicine



For additional information and references see
http://bionetgen.org/index.php/Tutorials

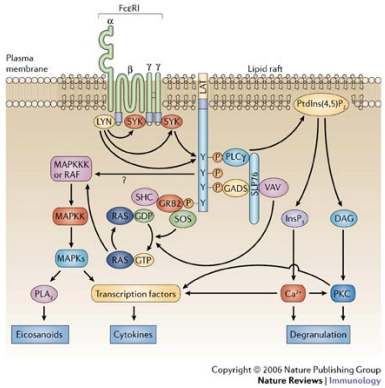
A Diagram is a Model





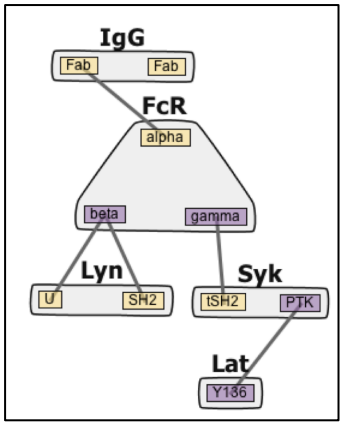
Bubba Thomas

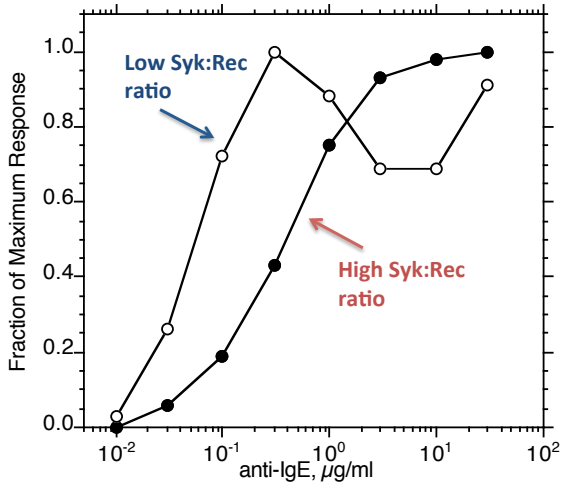
Computational Models



JJ

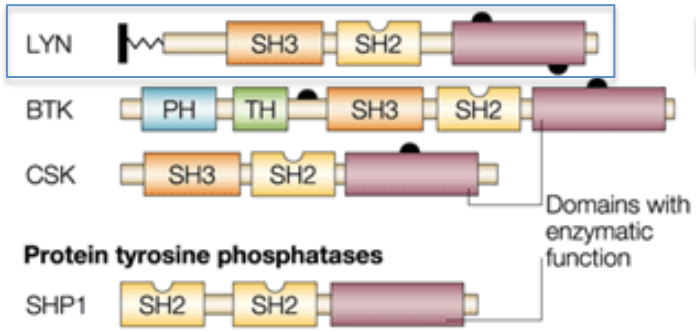
BioNetGen model

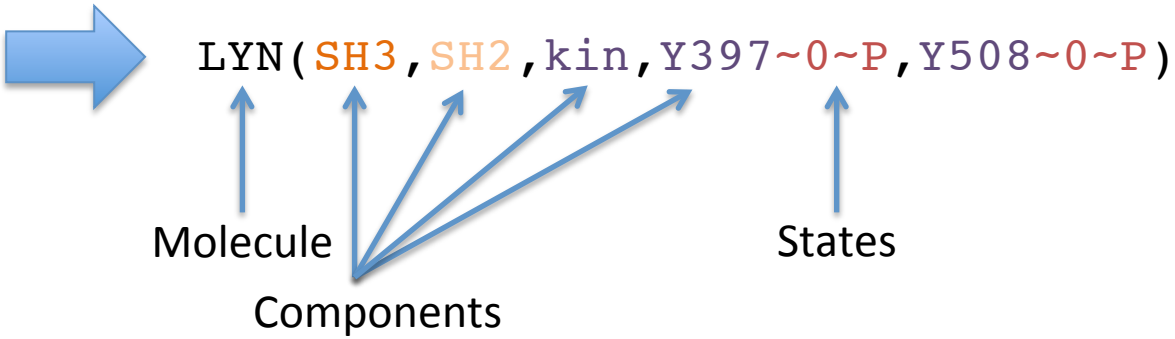


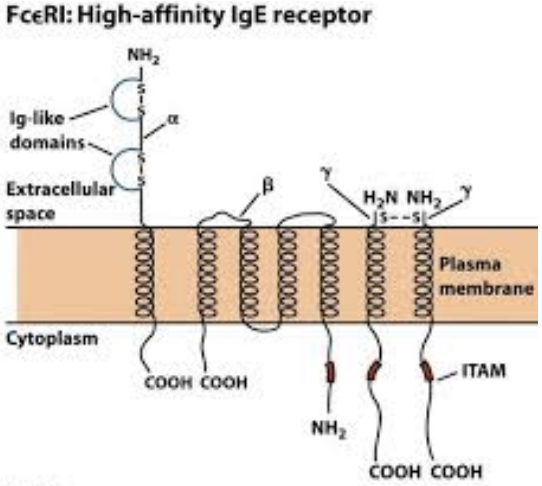


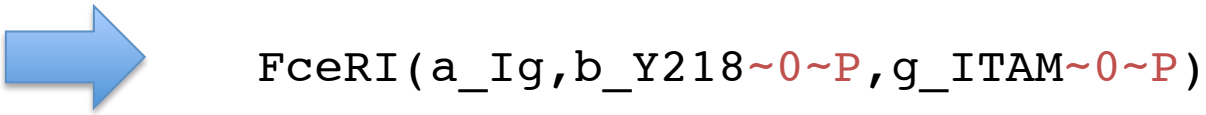
What is Rule-based Modeling (RBM)?

Molecules are modeled as *structured objects*



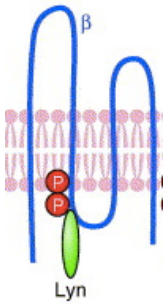






What is Rule-based Modeling (RBM)?

Rules define the interactions of molecules



"Lyn SH2 domain binds to phosphorylated Tyr 218 on the β subunit of FcεRI "

Reactants **Products** **Rate Law**

```
LYN(SH2) + Rec(bY_218~P) <-> LYN(SH2!1).FceRI(bY_218~P!1) kpL, kmL
```

bond

Center – elements modified by the action of the rule

Context – elements required for reaction to occur but not modified

"Don't write don't care" – elements not mentioned may be in any state

➔ *One rule can generate reactions involving many different species*

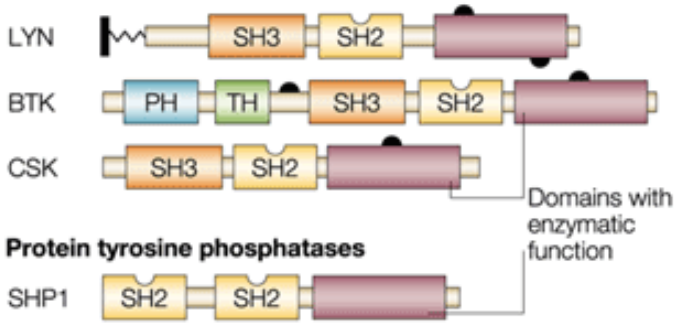
Reaction rate determined by **Mass Action kinetics**

rate forward = `kpL*[Lyn(SH2)]*[Rec(bY_218~P)]`

rate per reactant set (instance rate)

Why was RBM developed?

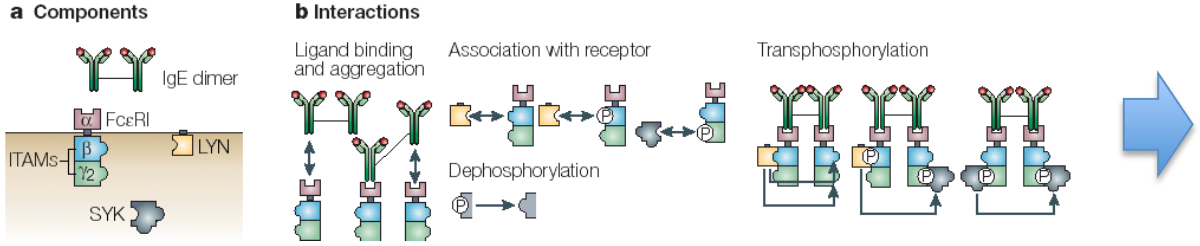
- Proteins are multi-functional

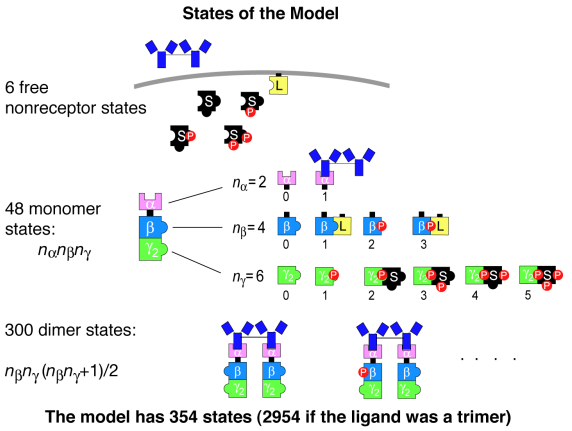


multiple sites of binding

multiple sites of posttranslational modification

- Representing their known interactions requires handling of *combinatorial complexity*



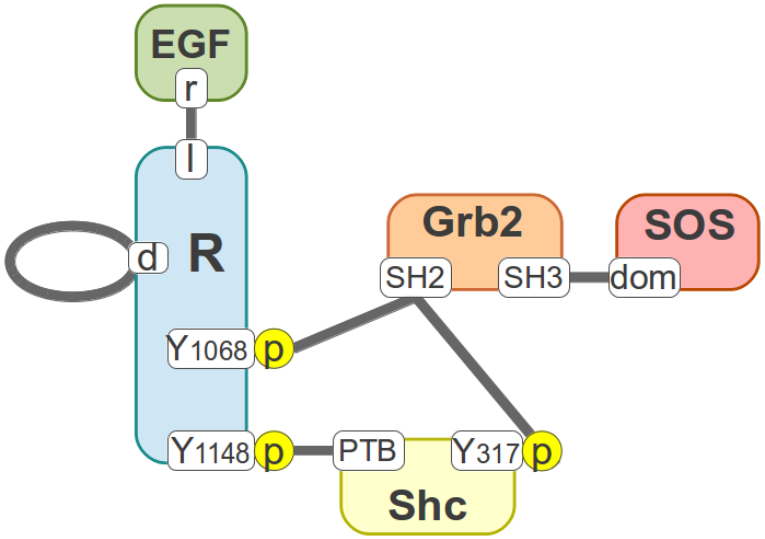


Small number of rules

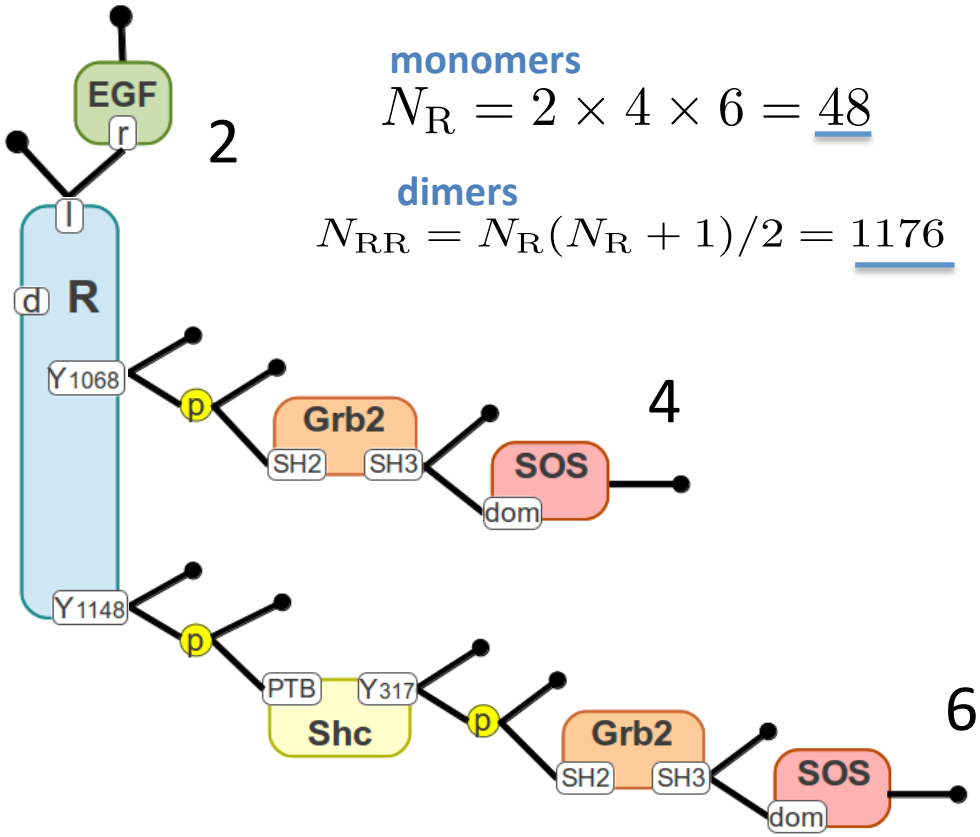
Small number of components and interactions ➔huge number of possible species and reactions

Combinatorial Complexity in Biochemical Interactions

Contact Map for molecules involved in EGFR signaling



Enumeration of receptor-containing species



monomers

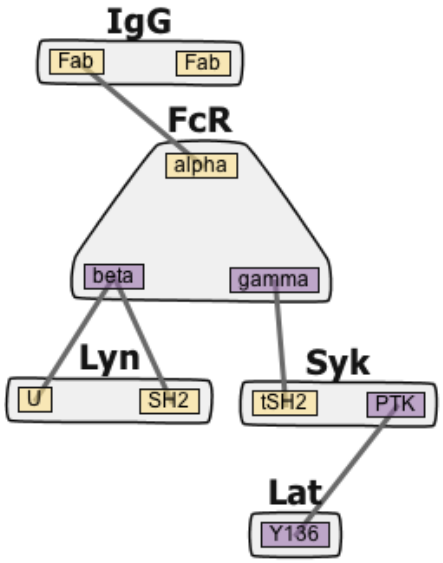
$$N_{\mathrm{R}} = 2 \times 4 \times 6 = 48$$

dimers

$$N_{\mathrm{RR}} = N_{\mathrm{R}}(N_{\mathrm{R}} + 1)/2 = 1176$$

Why use RBM?

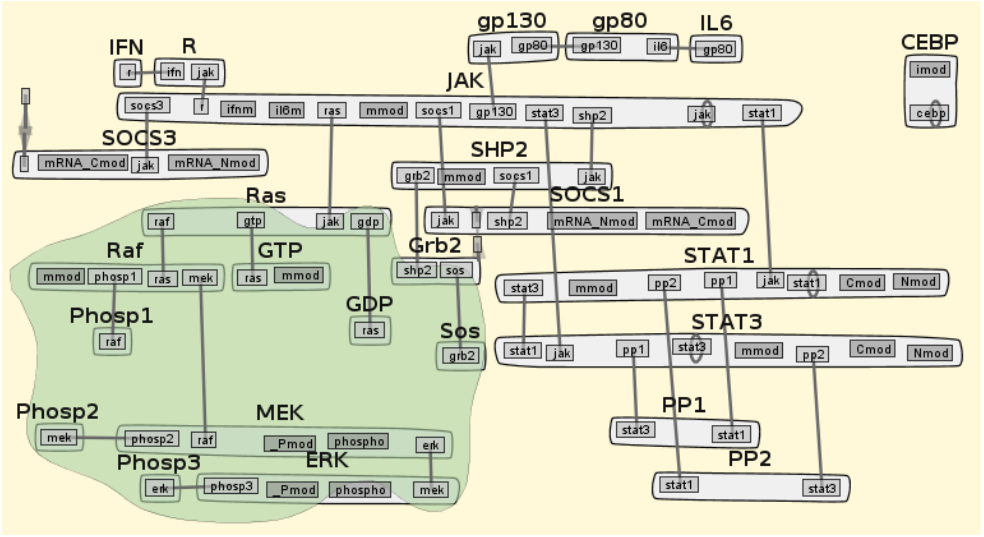
- *Concise and precise* representation of biochemical knowledge
  - rules are simple (less context) when interactions are modular
- *Flexible* with respect to simulation method
  - Deterministic / stochastic
  - Well-mixed / compartmental / spatial
- Structures and rules are *reusable*
  - Rule libraries
- Compact visual representation
  - Contact map and beyond



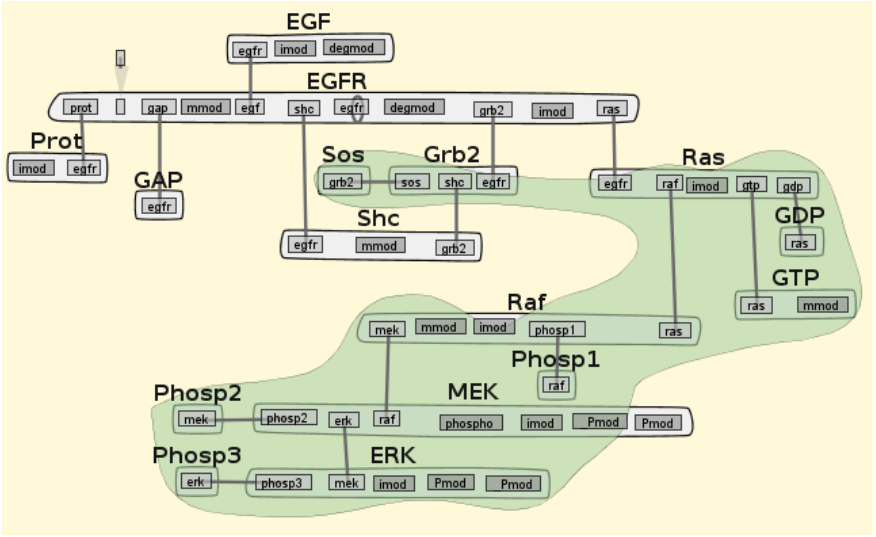
Contact map

Model Comparison

Generated by MOSBIE, a model exploration system within RuleBender



BioModels 543



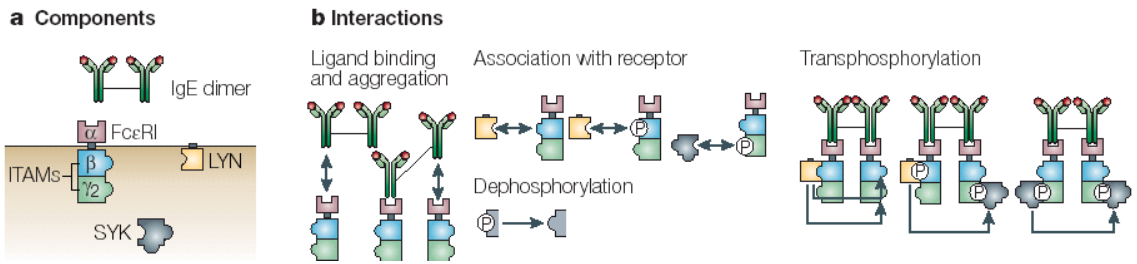
BioModels 19

Some Rule-Based Modeling Tools

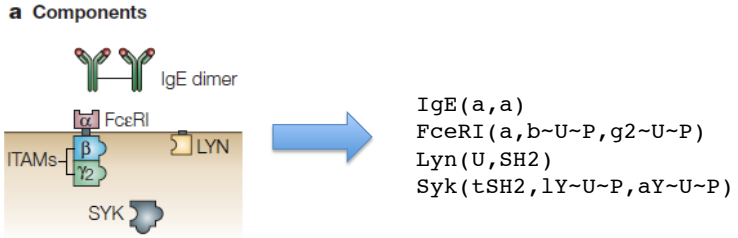
- BioNetGen & relatives
  - RuleBender
  - NFsim
  - BioNetGen@VirtualCell
- Kappa & relatives
- Simmune
- pySB
- SBML Level 3 Multi

Rule-Based Modeling protocol

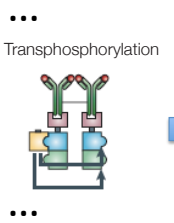
1. Identify components and interactions



2. Translate into objects (molecules) and rules

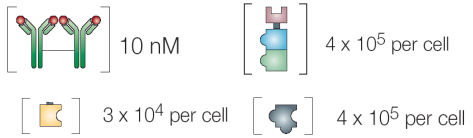



```
IgE(a,a)
FceRI(a,b~U~P,g2~U~P)
Lyn(U,SH2)
Syk(tSH2,lY~U~P,aY~U~P)
```




```
Lyn(U!1).FceRI(b!1).FceRI(b~U)-> \
Lyn(U!1).FceRI(b!1).FceRI(b~P)
```

3. Determine **concentrations** and **rate constants**



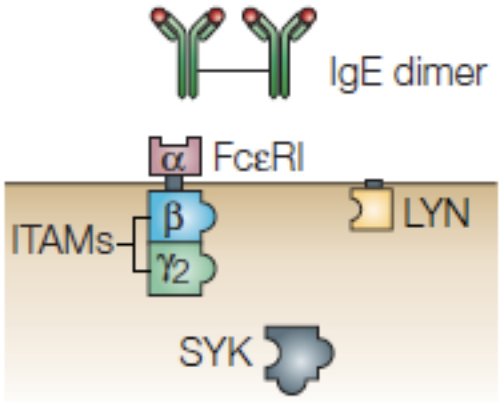
4. Simulate and analyze the model

ODE's SSA PDE's BD

SPECIFYING A RULE-BASED MODEL

Defining Molecules

Molecules are the basic objects in a BNG model



BioNetGen Language

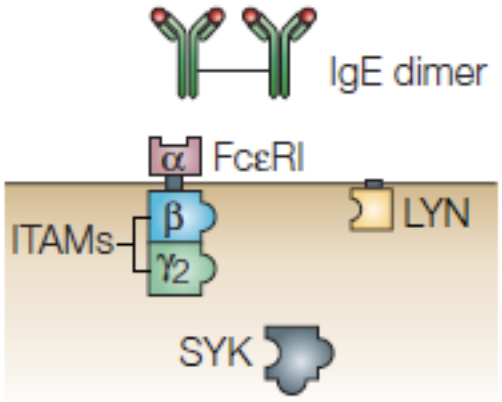
```
IgE(a,a)
FceRI(a,b~U~P,g2~U~P)
Lyn(U,SH2)
Syk(tSH2,lY~U~P,aY~U~P)
```

Components represent molecule elements

- Domains
- Motifs
- Properties

Defining Molecules

Molecules are the basic objects in a BNG model



BioNetGen Language

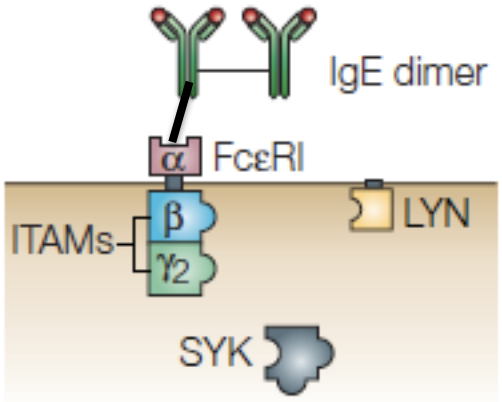
```
IgE(a,a)
FceRI(a,b~U~P,g2~U~P)
Lyn(U,SH2)
Syk(tSH2,lY~U~P,aY~U~P)
```

Components may have different states representing

- posttranslational modifications
- conformational state
- …

Binding

Molecules bind other molecules through components



BioNetGen Language

```
IgE(a,a!1).FceRI(a!1,b~U,g2~U)
```

Bonds are formed by linking two components. The '.' indicates a set of molecules forming a complex.

```
FceRI(a,b~U!1,g2~U).Lyn(U!1)
```

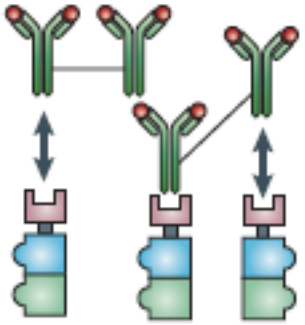
Components may have both states and bonds.

```
Lyn(SH2!1,Cterm~P!1)
```

Bonds may occur within a molecule.

Defining Interaction Rules

Ligand binding and aggregation

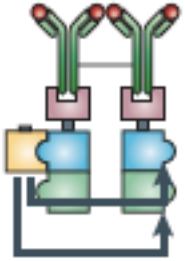


BioNetGen Language

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1)
…
```

binding and dissociation

Transphosphorylation

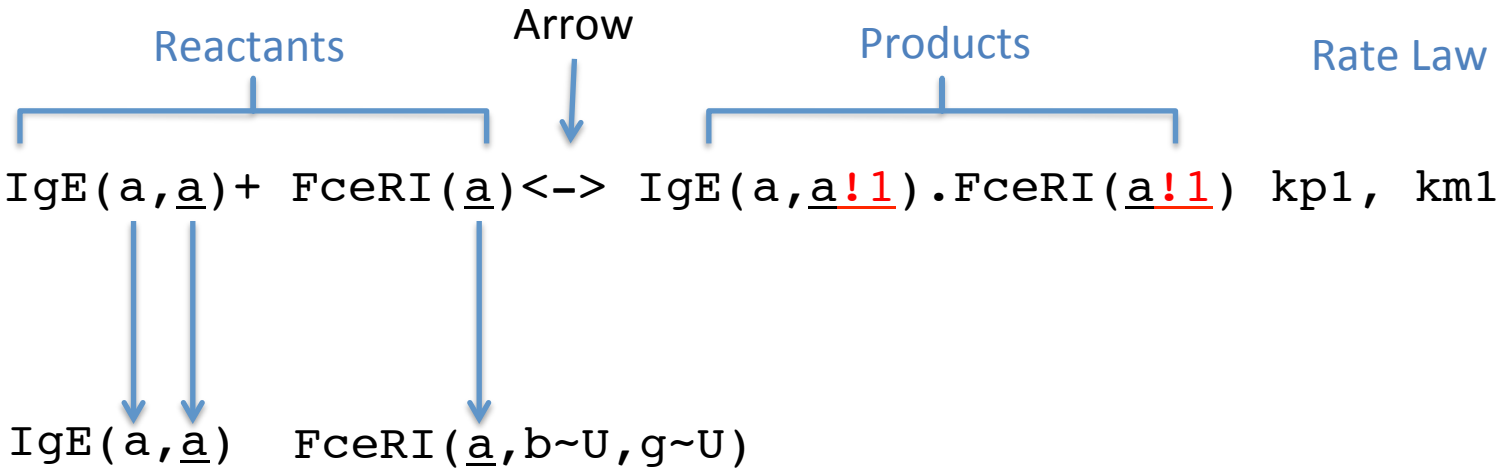


```
Lyn(U!1).FceRI(b!1).FceRI(b~U)-> \
Lyn(U!1).FceRI(b!1).FceRI(b~P)
```

component state change

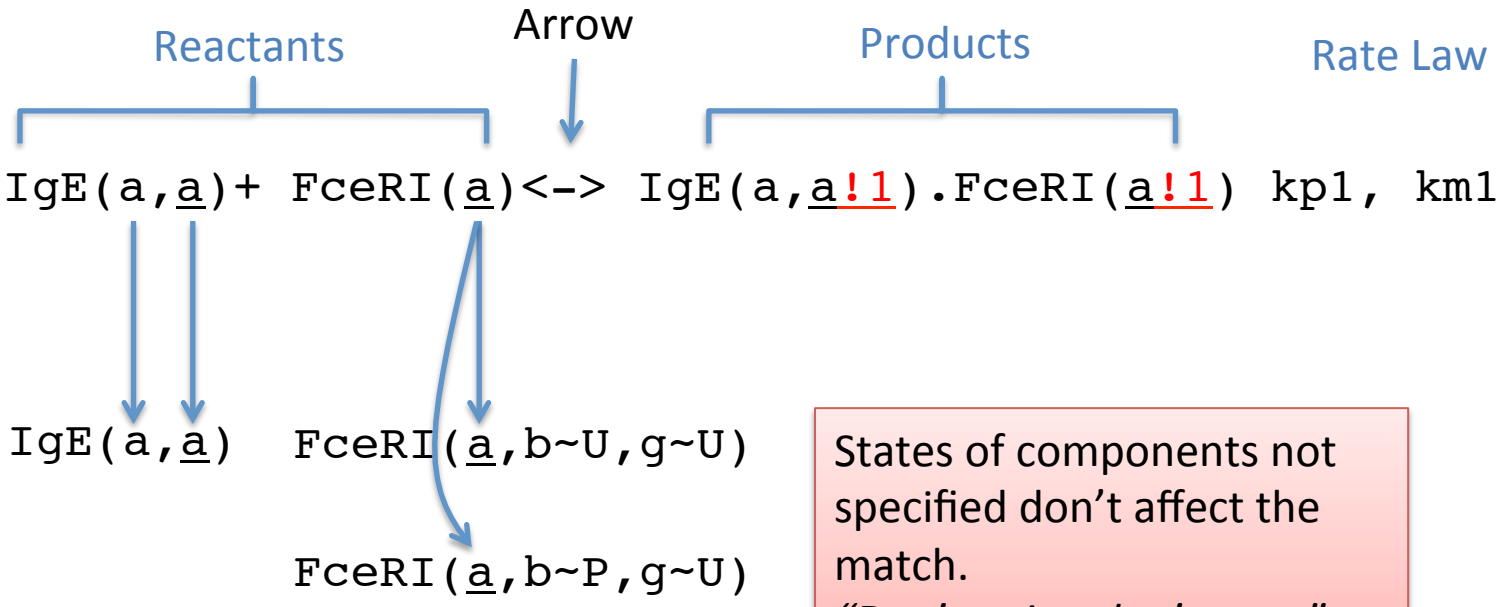
Parts of a rule

Parts of a rule



Reactant patterns select properties of each reactant molecule.

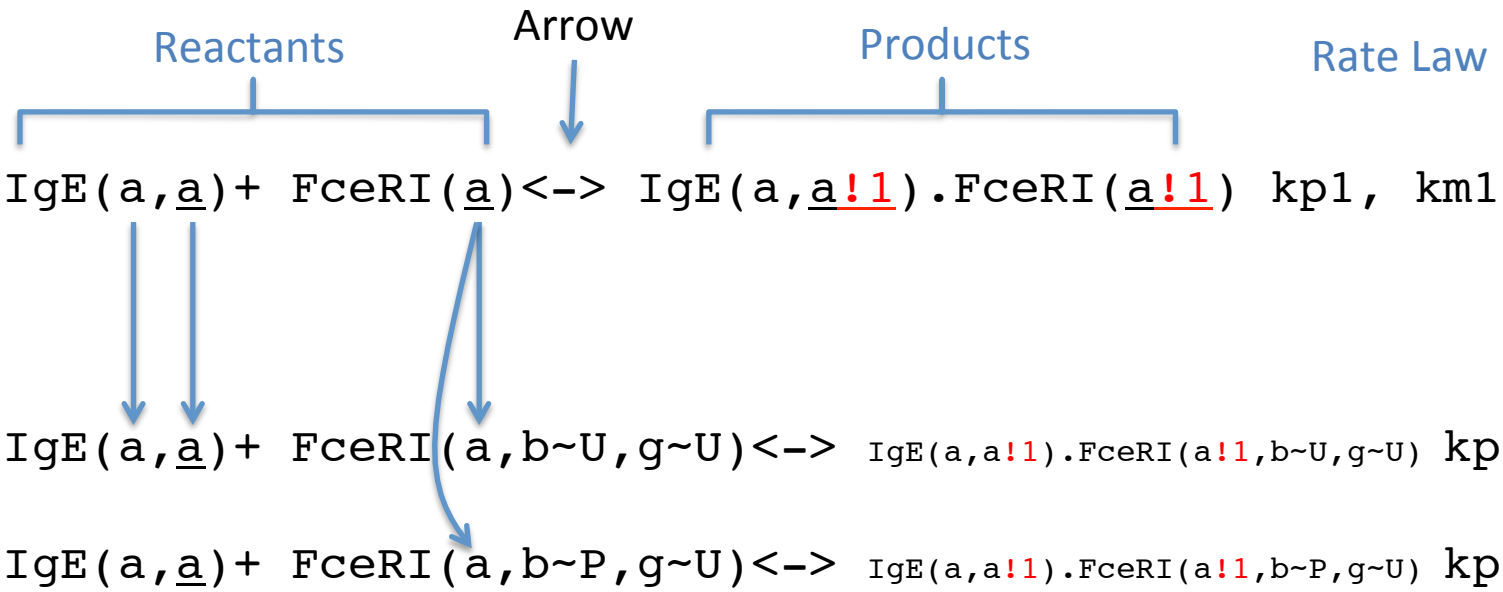
Parts of a rule



States of components not specified don't affect the match.
"Don't write, don't care."

Reactant patterns select properties of each reactant molecule.

Parts of a rule



Reactant patterns select properties of each reactant molecule.

Because patterns can match many different species, each rule can generate many reactions.

Center and context

The **center** of a rule is the part that the rule changes.

bond is formed

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

The **context** is the part that is necessary for the rule to happen but is unchanged.

Center and context

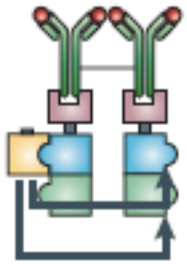
The **center** of a rule is the part that the rule changes.

bond is formed

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

The **context** is the part that is necessary for the rule to happen but is unchanged.

Transphosphorylation

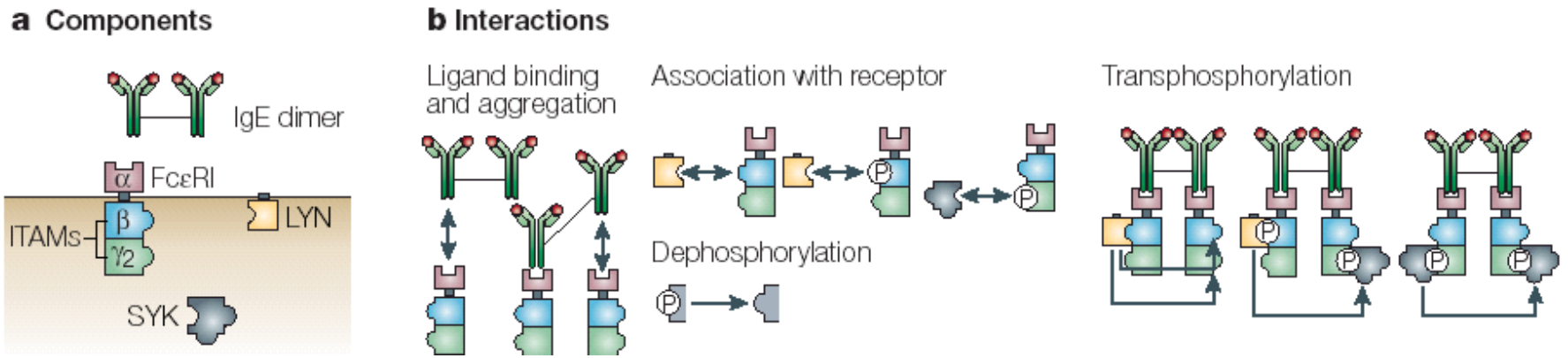


Context can represent complex biochemistry.

component state is changed

```
Lyn(U!1).FceRI(b!1).FceRI(b~U)-> \
Lyn(U!1).FceRI(b!1).FceRI(b~P)
```

Composition of a Rule-Based Model



Molecules

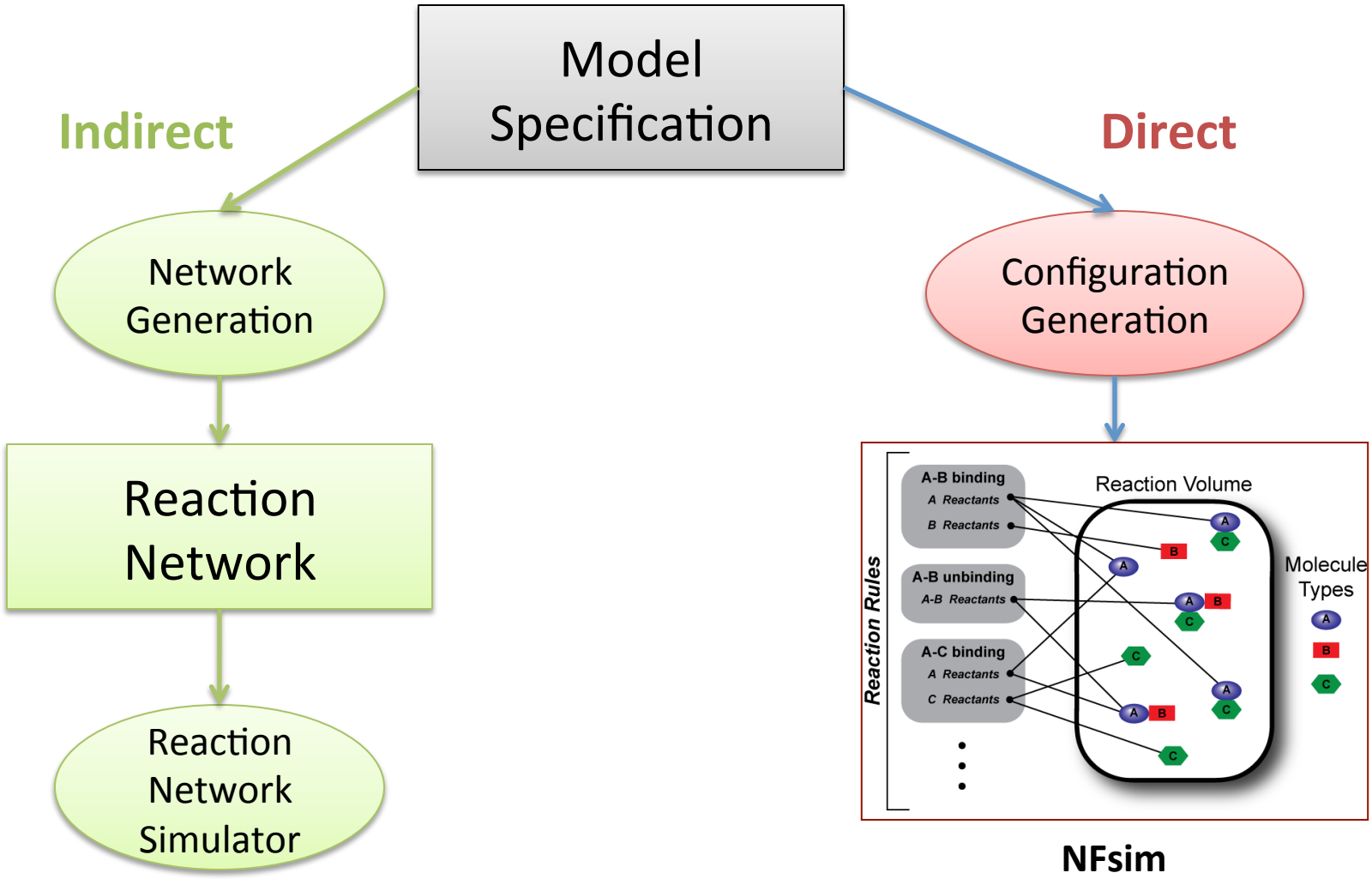
```
begin molecules
Lig(l,l)
Lyn(U,SH2)
Syk(tSH2,l~U~P,a~U~P)
Rec(a,b~U~P,g~U~P)
end molecules
```

Reaction Rules

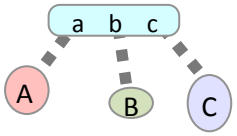
BioNetGen language

```
begin reaction_rules
# Ligand-receptor binding
1 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1
 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1

# Receptor-aggregation
2 Rec(a) + Lig(l,l!1) <-> Rec(a!2).Lig(l!2,l!1) kp2,km2

# Constitutive Lyn-receptor binding
3 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL
…
```

Methods for simulating RBMs

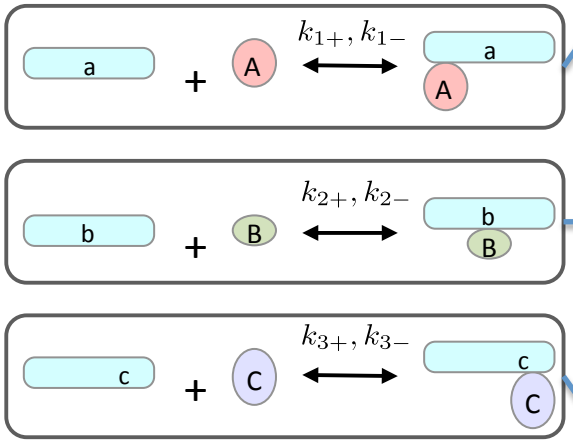


Indirect Methods – Network Generation



contact map

reaction rules



reactions

S	+	A	$\overset{k_{1+},k_{1-}}{\rightleftharpoons}$	SA
SB	+	A	$\overset{k_{1+},k_{1-}}{\rightleftharpoons}$	SAB
SC	+	A	$\overset{k_{1+},k_{1-}}{\rightleftharpoons}$	SAC
SBC	+	A	$\overset{k_{1+},k_{1-}}{\rightleftharpoons}$	SABC
S	+	B	$\overset{k_{2+},k_{2-}}{\rightleftharpoons}$	SB
SA	+	B	$\overset{k_{2+},k_{2-}}{\rightleftharpoons}$	SAB
SC	+	B	$\overset{k_{2+},k_{2-}}{\rightleftharpoons}$	SBC
SAC	+	B	$\overset{k_{2+},k_{2-}}{\rightleftharpoons}$	SABC
S	+	C	$\overset{k_{3+},k_{3-}}{\rightleftharpoons}$	SC
SA	+	C	$\overset{k_{3+},k_{3-}}{\rightleftharpoons}$	SAC
SB	+	C	$\overset{k_{3+},k_{3-}}{\rightleftharpoons}$	SBC
SAB	+	C	$\overset{k_{3+},k_{3-}}{\rightleftharpoons}$	SABC

S, A, B, C,
SA, SB, SC,
SAB, SAC, SBC,
SABC

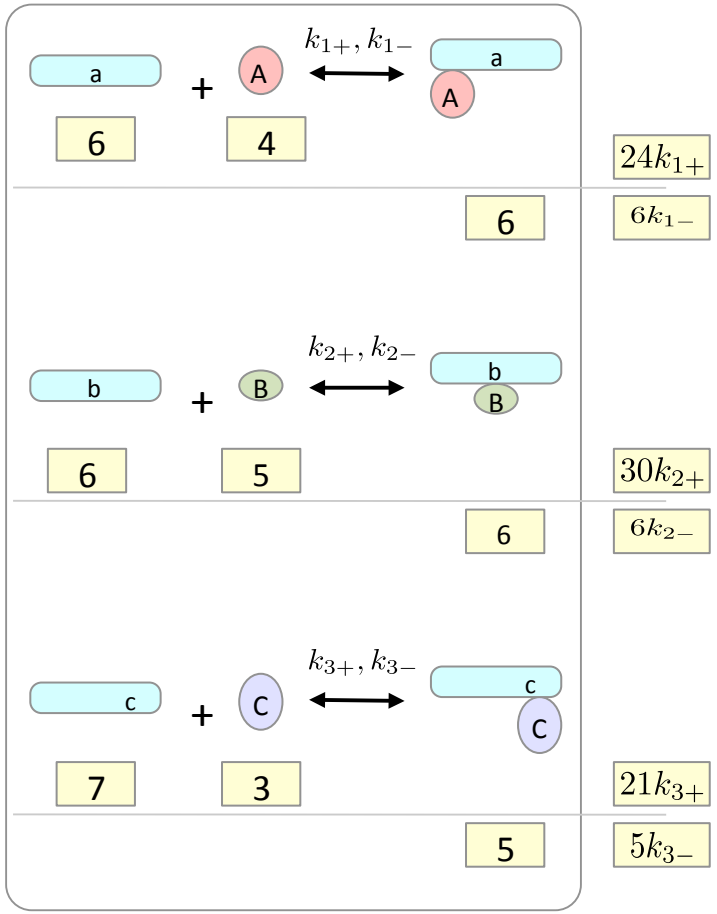
species

4 molecule types
3 rules

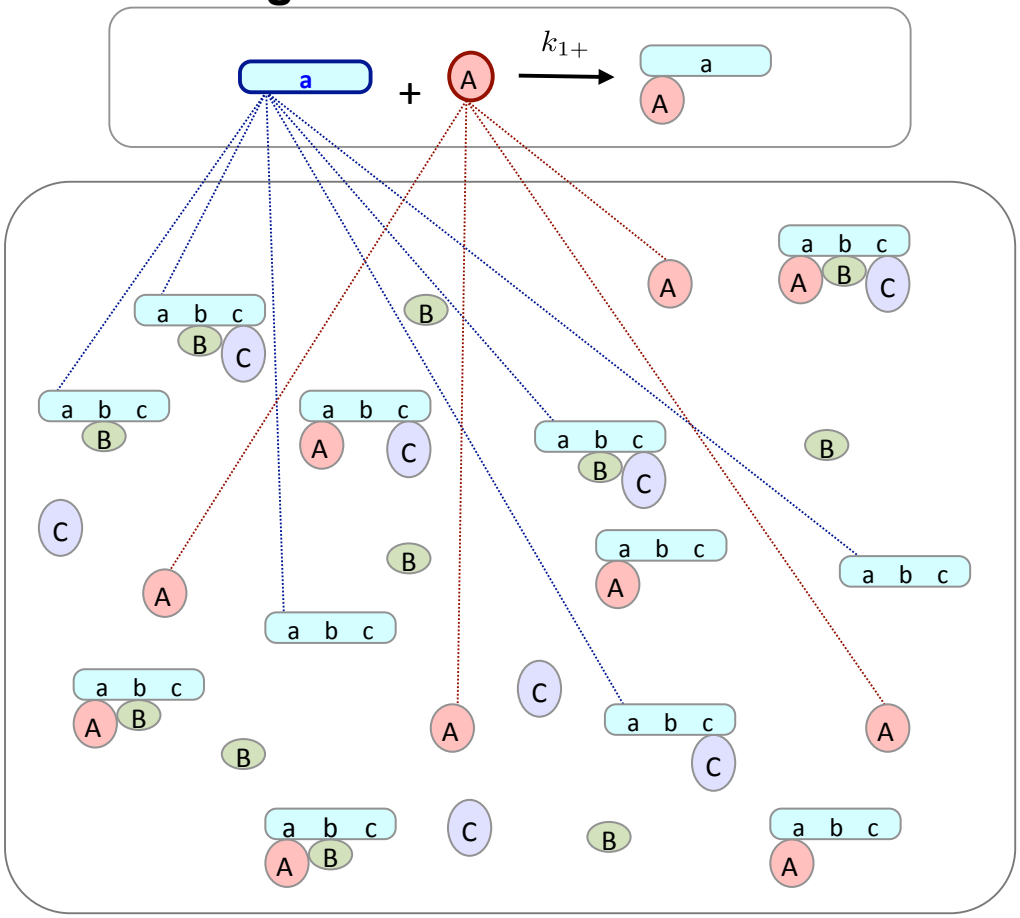
11 species
12 reactions

Direct Methods (NFsim)

reaction rules



event generation



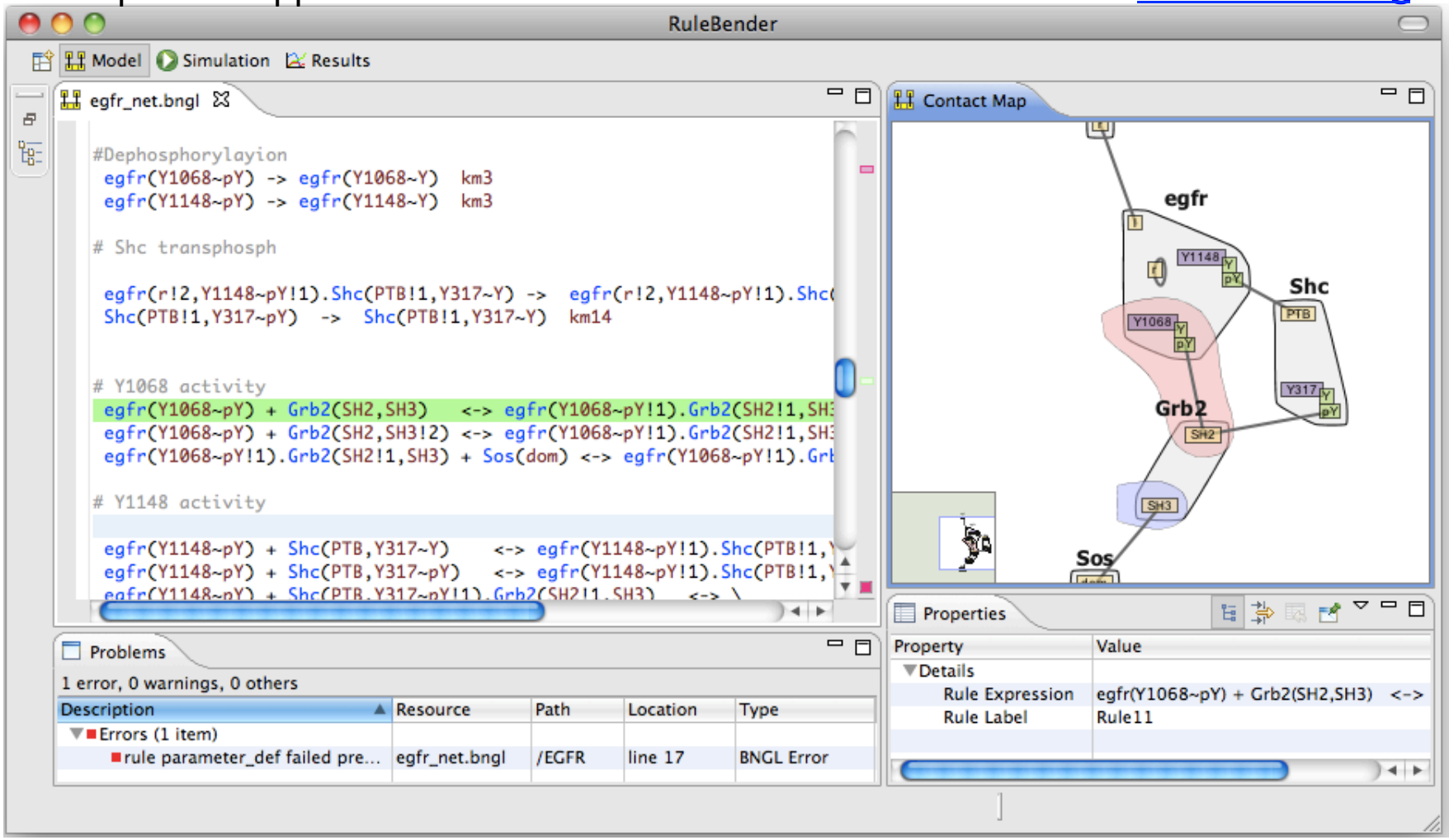
system configuration

total propensity $24k_{1+} + 6k_{1-} + 30k_{2+} + 6k_{2-} + 21k_{3+} + 5k_{3-}$

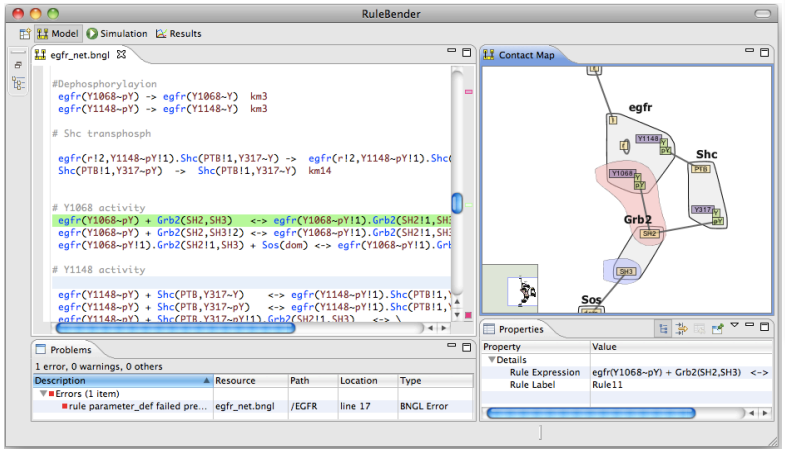
RuleBender

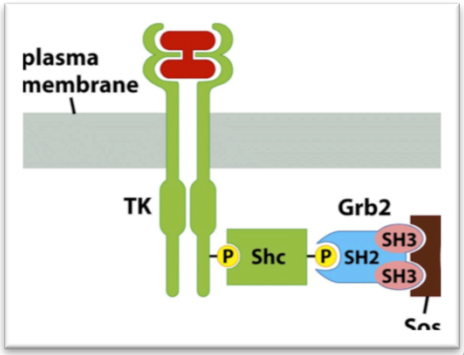
An eclipse RCP application

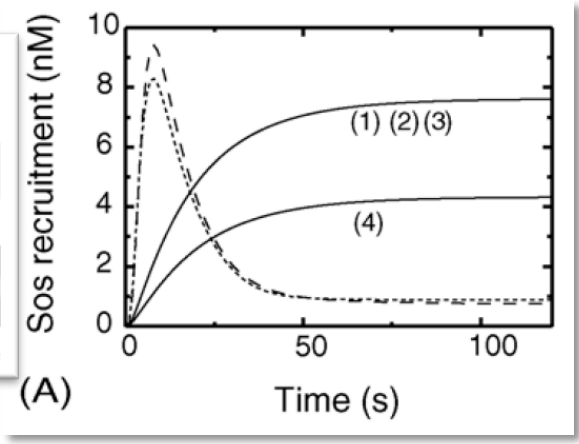
rulebender.org

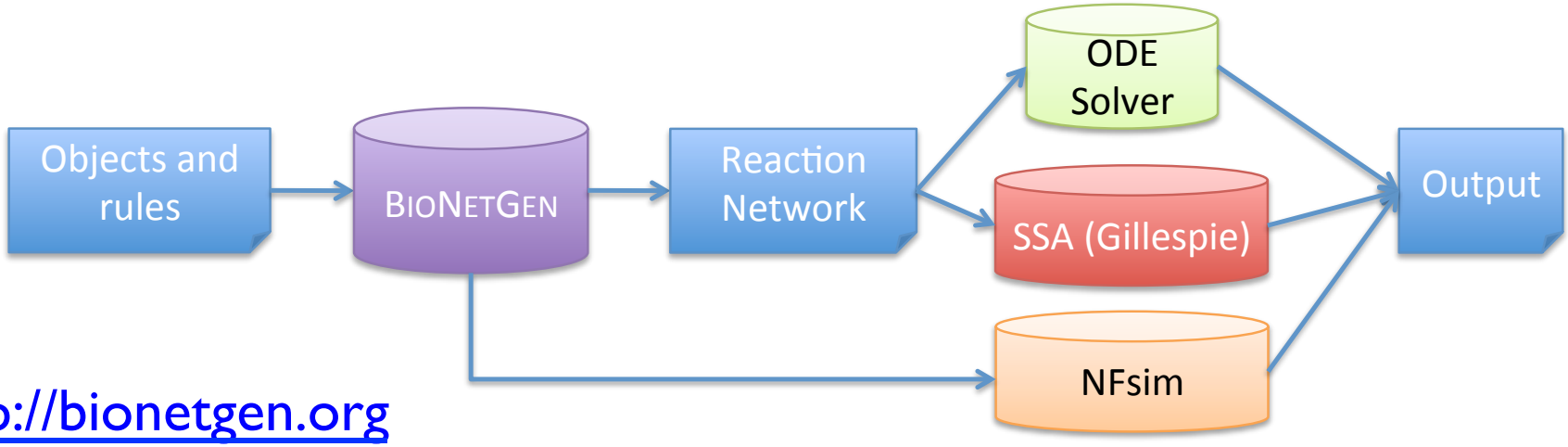


Basic RBM workflow with BioNetGen



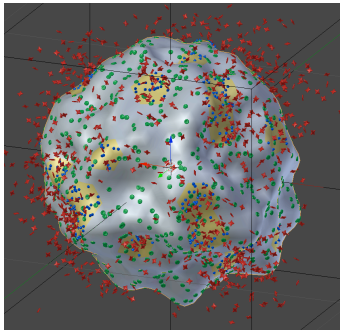






http://bionetgen.org
http://rulebender.org
http://nfsim.org

Links to other simulation tools



SBML

Matlab

Virtual Cell

PDEs

BNGL

BioNetGen

Reaction Network

CVODE

SSA (Gillespie)

PLA (τ-leap)

Output

MCell

NFsim

Brownian dynamics

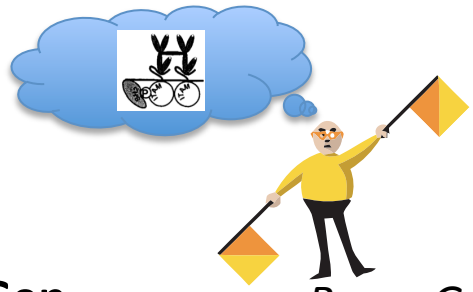
RuleMonkey

rejection-free network free

pySB

SRsim

Brownian dynamics + mol. structure



Credits

BioNetGen

Jim Faeder (Pitt)
Leonard Harris
Justin Hogg
John Sekar
Jose-Juan Tapia
Ilya Korsunksy
Robert Sheehan
Dipak Barua
Arshi Arora
Natasa Miskov-Zivanov
Robert Clark

Byron Goldstein (LANL)

Michael Blinov (UCHC)

William Hlavacek (LANL)
Lily Chylek
Jin Yang
Bin Hu
Matthew Fricke

Richard Posner (NAU)
Josh Colvin

RuleBender
Liz Marai (Pitt , now UIC)
Adam Smith
Wen Xu
John Wenskovitch
Yao Sun
Robert Engelke

NFsim
Thierry Emonet (Yale)
Michael Sneddon

Funding

- NIH
  P41 GM103712, T32 EB009403, UL1 RR024153, R01 GM076570, GM085273, AI35997 and CA109552
- NSF
  Expeditions in Computing, CCF-0829788
- University of Pittsburgh School of Medicine, Arizona Biomedical Research Commission, Los Alamos National Laboratory, Department of Energy