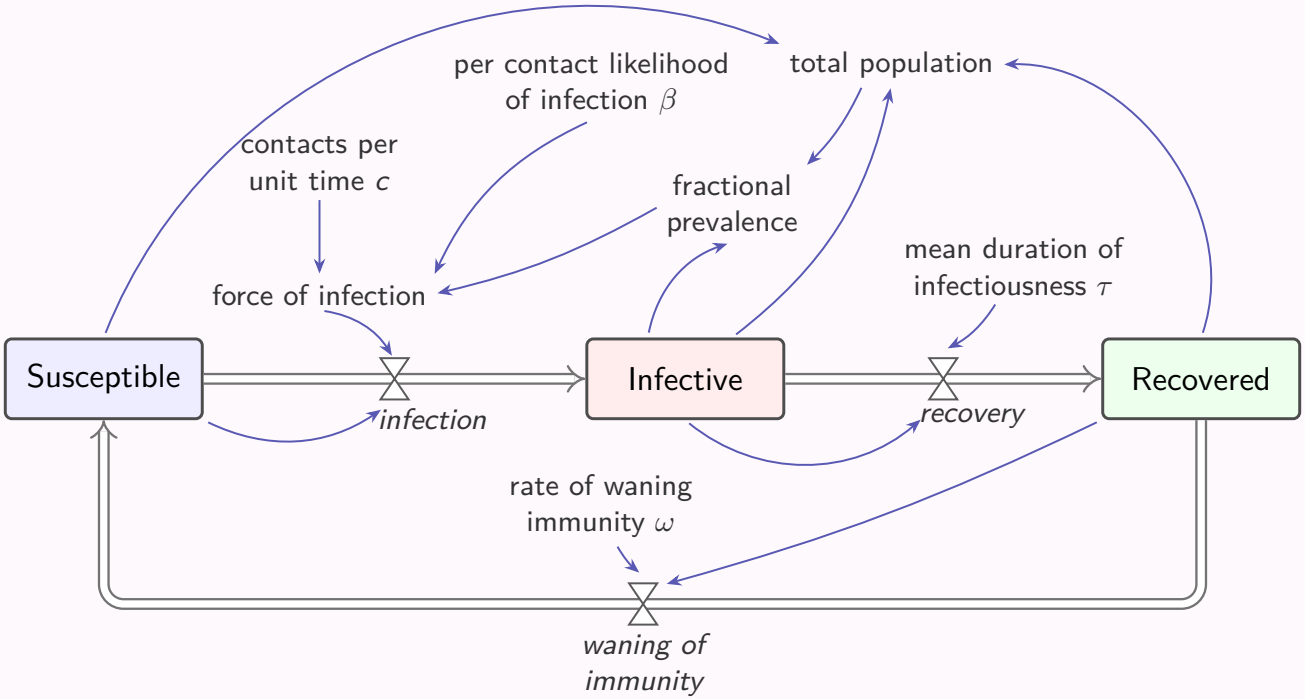
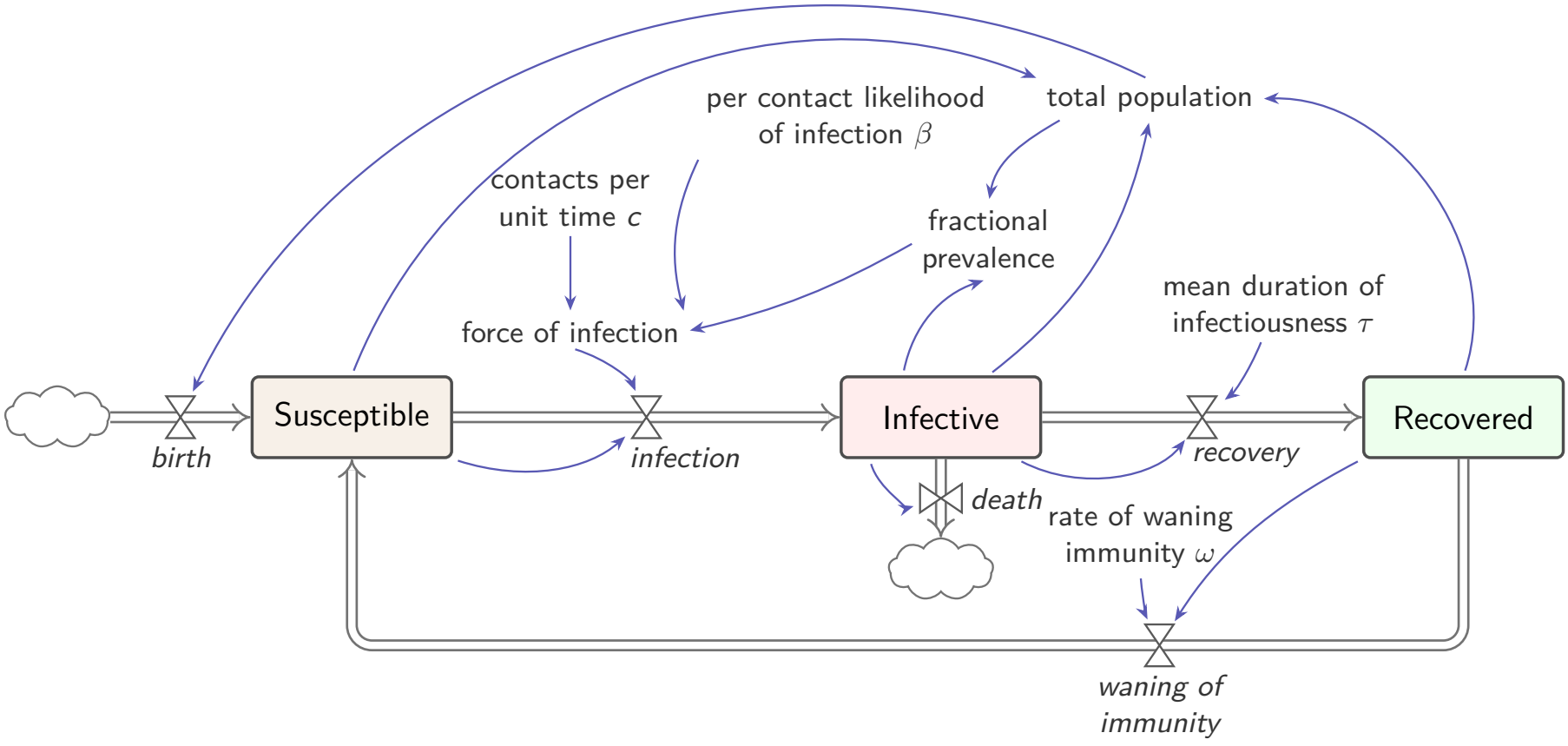
Applied Category Theory and The Mathematics of Networks – III



John Baez, University of Edinburgh

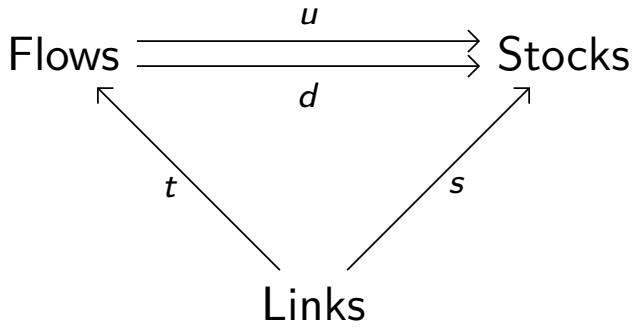
GeoCOW 2026

In general a stock-flow diagram can have many elements including "auxiliary variables" and "clouds":

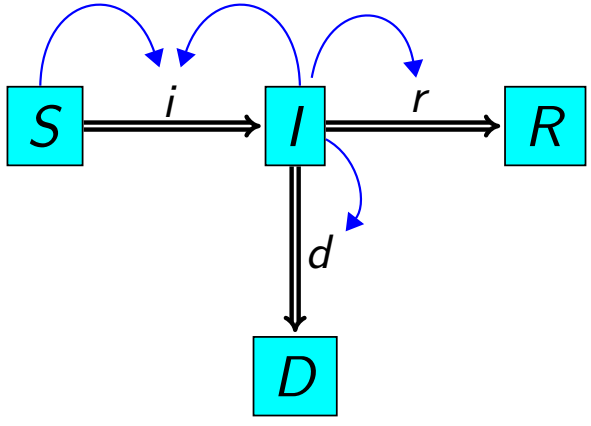


To keep this presentation quick, let's leave out those two elements.

Thus, for now, a **stock-flow diagram** consists of finite sets and functions:



together with, for each $f \in \text{Flows}$, a **flow function** $\phi_f : \mathbb{R}^{L(f)} \to \mathbb{R}$ where $L(f)$ is the set of all $\ell \in \text{Links}$ with $t(\ell) = f$.

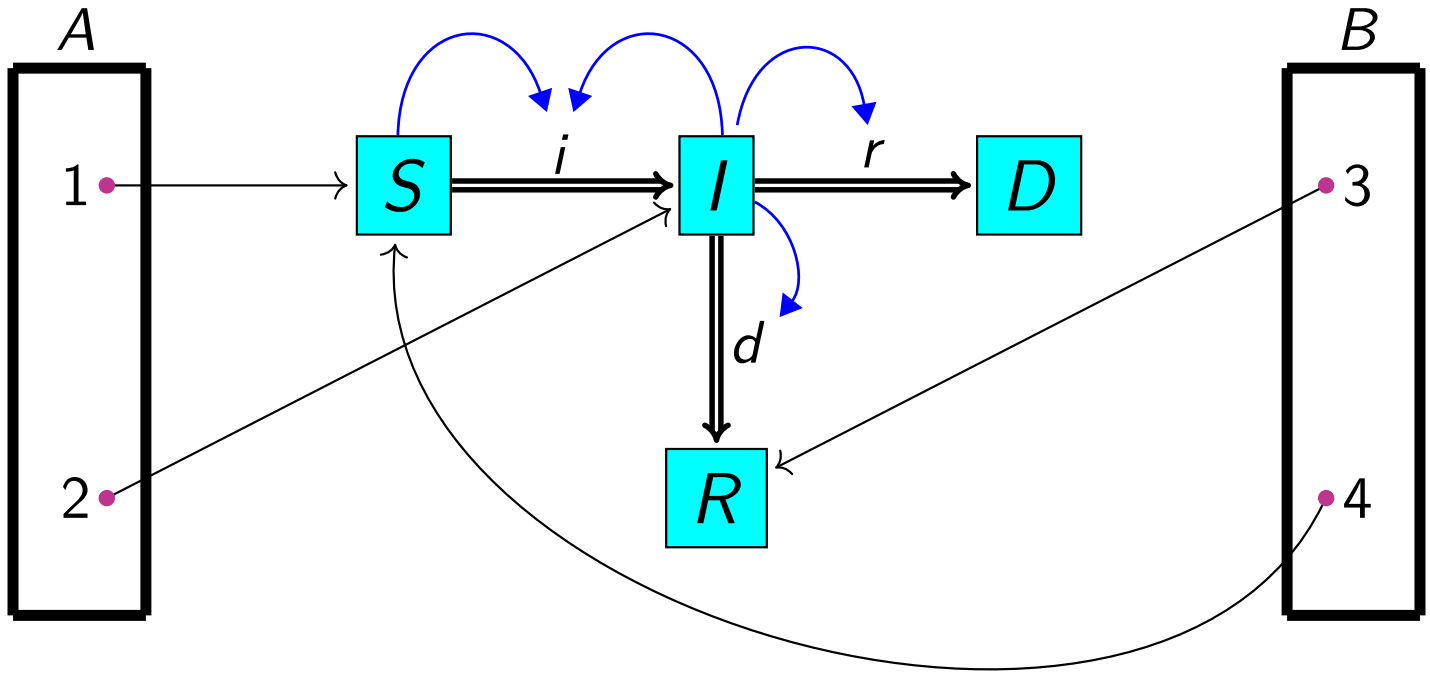


$\phi_i : \mathbb{R}^2 \to \mathbb{R}$ gives $\phi_i(S, I)$

$\phi_r : \mathbb{R} \to \mathbb{R}$ gives $\phi_r(I)$

$\phi_d : \mathbb{R} \to \mathbb{R}$ gives $\phi_d(I)$

An **open stock-flow diagram** is a stock-flow diagram equipped with maps $i\colon A \to \text{Stocks}, o\colon B \to \text{Stocks}$ for some finite sets A, B.

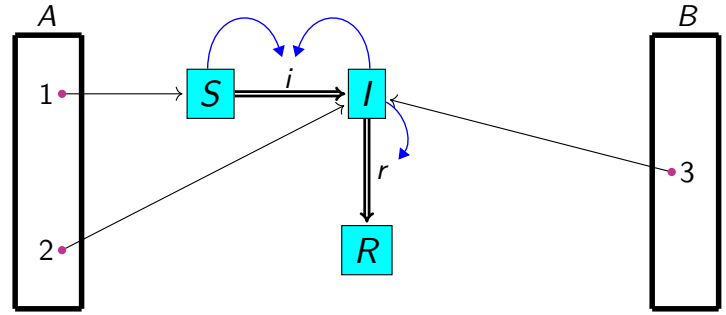


We call this an open stock-flow diagram **from A to B** and write it as $A \xrightarrow{F} B$.

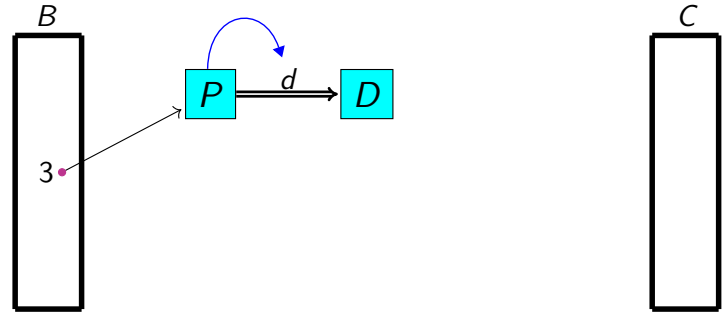
We can **compose** open stock-flow diagrams $A \xrightarrow{F} B$ and $B \xrightarrow{G} C$ by "gluing them together along B".

We get an open stock-flow diagram called $A \xrightarrow{GF} C$.

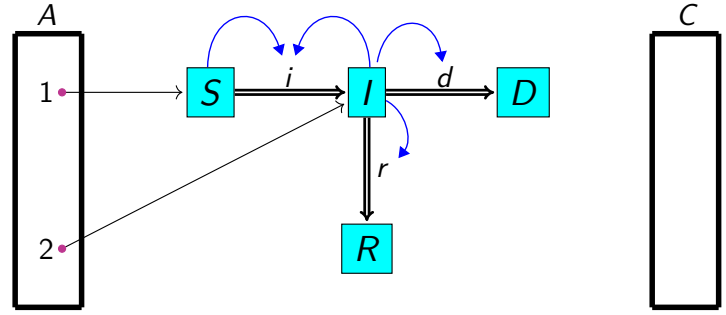
$A \xrightarrow{F} B$



$B \xrightarrow{G} C$



$A \xrightarrow{GF} C$



Thus, we get a category **Open**(**StockFlow**) with:

- finite sets as objects,
- open stock-flow diagrams as morphisms.

In fact this is a monoidal category, where:

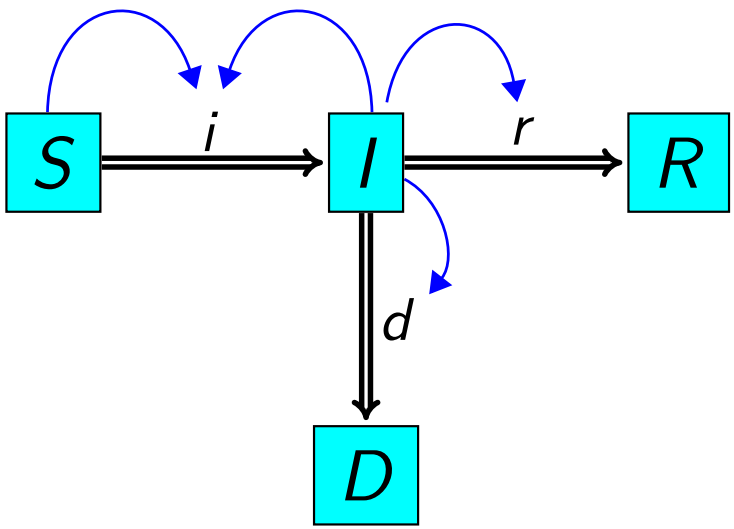
- the tensor product of finite sets is their disjoint union;
- to tensor two open stock-flow diagrams we "set them side by side".

Next we can construct a monoidal category **Open**(**Dynam**) of "open dynamical systems", and a monoidal functor

$$\Phi : \mathbf{Open}(\mathbf{StockFlow}) \to \mathbf{Open}(\mathbf{Dynam})$$

This turns any open stock-flow diagram into an open dynamical system.

We have already seen how this works without the "openness":



ϕ_i

ϕ_r

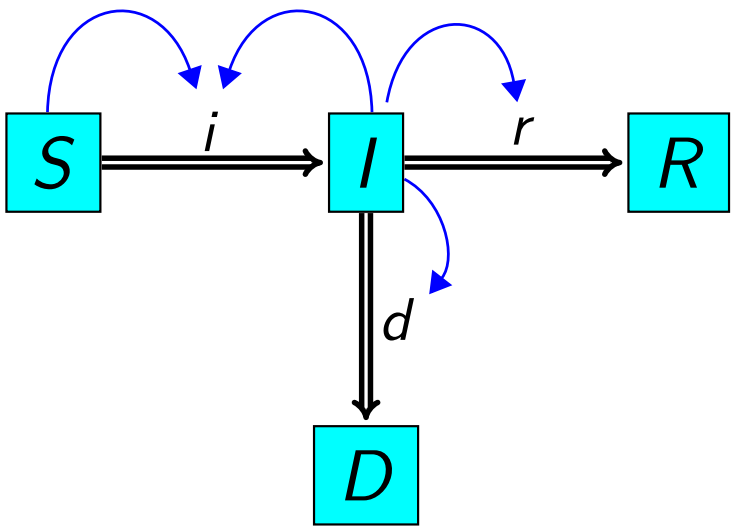
ϕ_d

Next we can construct a monoidal category **Open**(**Dynam**) of "open dynamical systems", and a monoidal functor

$$\Phi\colon \mathbf{Open}(\mathbf{StockFlow}) \to \mathbf{Open}(\mathbf{Dynam})$$

This turns any open stock-flow diagram into an open dynamical system.

We have already seen how this works without the "openness":



$$\frac{dS}{dt} = -\phi_i(S, I)$$

$$\frac{dI}{dt} = \phi_i(S, I) - \phi_r(I) - \phi_d(I)$$

$$\frac{dR}{dt} = \phi_r(I)$$

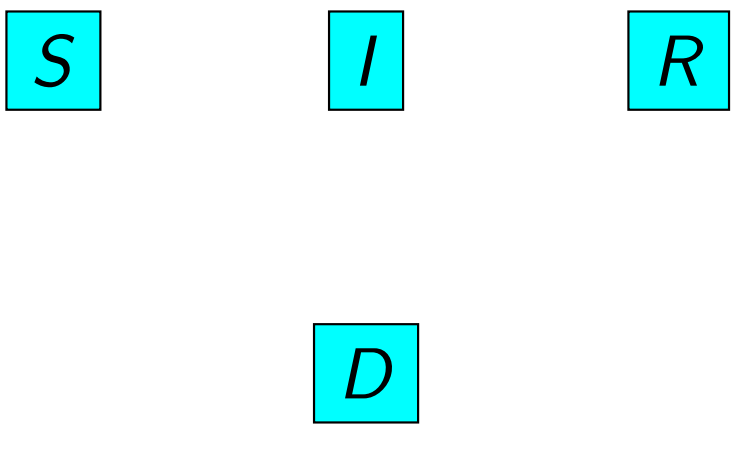
$$\frac{dD}{dt} = \phi_d(I)$$

Next we can construct a monoidal category **Open**(**Dynam**) of "open dynamical systems", and a monoidal functor

$$\Phi\colon \mathbf{Open}(\mathbf{StockFlow}) \to \mathbf{Open}(\mathbf{Dynam})$$

This turns any open stock-flow diagram into an open dynamical system.

We have already seen how this works without the "openness":



$$\frac{dS}{dt} = -\phi_i(S, I)$$

$$\frac{dI}{dt} = \phi_i(S, I) - \phi_r(I) - \phi_d(I)$$

$$\frac{dR}{dt} = \phi_r(I)$$

$$\frac{dD}{dt} = \phi_d(I)$$

A **dynamical system** on some finite set of variables X is a vector field v on $\mathbb{R}^X$. This lets us write down a system of first-order ordinary differential equations.

For example, if $X = \{S, I, D, R\}$ and v is the 4-component vector field (v_S, v_I, v_D, v_R) on $\mathbb{R}^X \cong \mathbb{R}^4$, we get

$$\begin{aligned}
\frac{d}{dt}S(t) &= v_S(S(t), I(t), D(t), R(t)) \\
\frac{d}{dt}I(t) &= v_I(S(t), I(t), D(t), R(t)) \\
\frac{d}{dt}D(t) &= v_D(S(t), I(t), D(t), R(t)) \\
\frac{d}{dt}R(t) &= v_R(S(t), I(t), D(t), R(t))
\end{aligned}$$

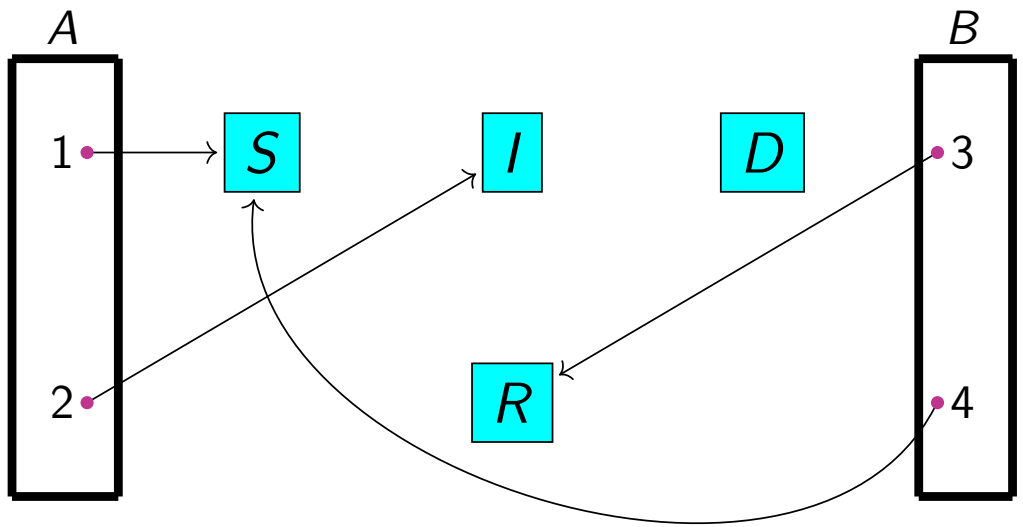
A **dynamical system** on some finite set of variables X is a vector field v on $\mathbb{R}^X$. This lets us write down a system of first-order ordinary differential equations.

For example, if $X = \{S, I, D, R\}$ and v is the 4-component vector field (v_S, v_I, v_D, v_R) on $\mathbb{R}^X \cong \mathbb{R}^4$, we get

$$\begin{aligned} \frac{dS}{dt} &= v_S \\ \frac{dI}{dt} &= v_I \\ \frac{dD}{dt} &= v_D \\ \frac{dR}{dt} &= v_R \end{aligned}$$

An **open dynamical system** $A \xrightarrow{v} B$ is a dynamical system v on some finite set X equipped with maps $i\colon A \to X$, $o\colon B \to X$ for some finite sets A, B.

For example:



$$\frac{dS}{dt} = v_S$$

$$\frac{dI}{dt} = v_I$$

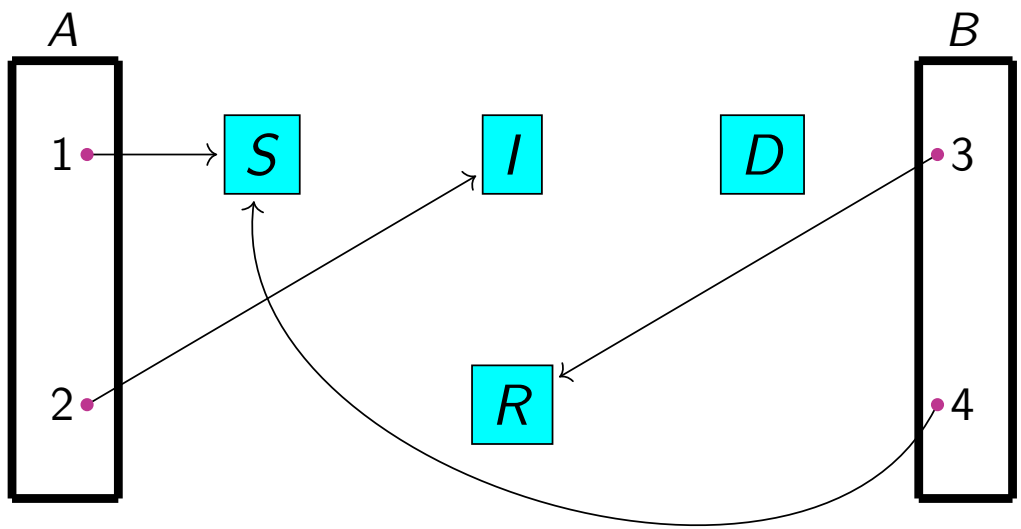
$$\frac{dR}{dt} = v_R$$

$$\frac{dD}{dt} = v_D$$

Here $X = \{S, I, D, R\}$.

An **open dynamical system** $A \xrightarrow{v} B$ is a dynamical system v on some finite set X equipped with maps $i\colon A \to X$, $o\colon B \to X$ for some finite sets A, B.

For example:



$$\frac{dS}{dt} = v_S + i_1 - o_4$$

$$\frac{dI}{dt} = v_I + i_2$$

$$\frac{dR}{dt} = v_R - o_3$$

$$\frac{dD}{dt} = v_D$$

Here $X = \{S, I, D, R\}$.

Just as we constructed **Open**(**StockFlow**), we can construct a monoidal category **Open**(**Dynam**) with:

- finite sets as objects,
- open dynamical systems as morphisms.

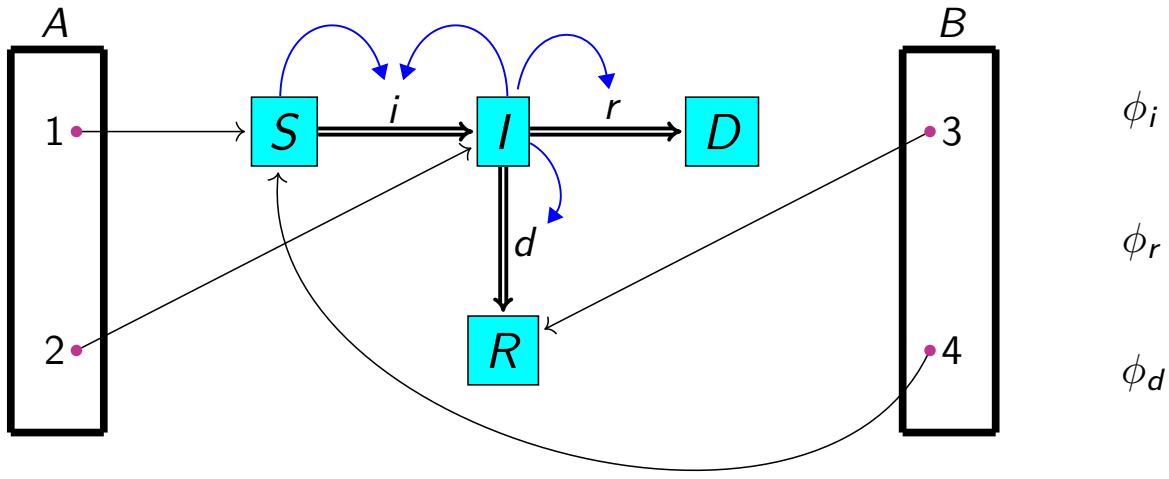
Just as we constructed **Open**(**StockFlow**), we can construct a monoidal category **Open**(**Dynam**) with:

- finite sets as objects,
- open dynamical systems as morphisms.

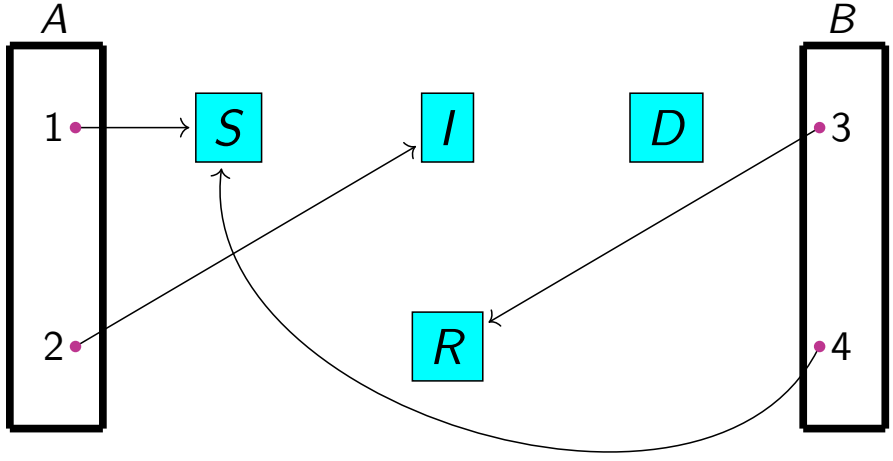
The process we've already seen for converting stock-flow diagrams into dynamical systems extends to a monoidal functor

$$\Phi \colon \mathbf{Open}(\mathbf{StockFlow}) \to \mathbf{Open}(\mathbf{Dynam})$$

For example, Φ maps this open stock-flow diagram:



to this open dynamical system:



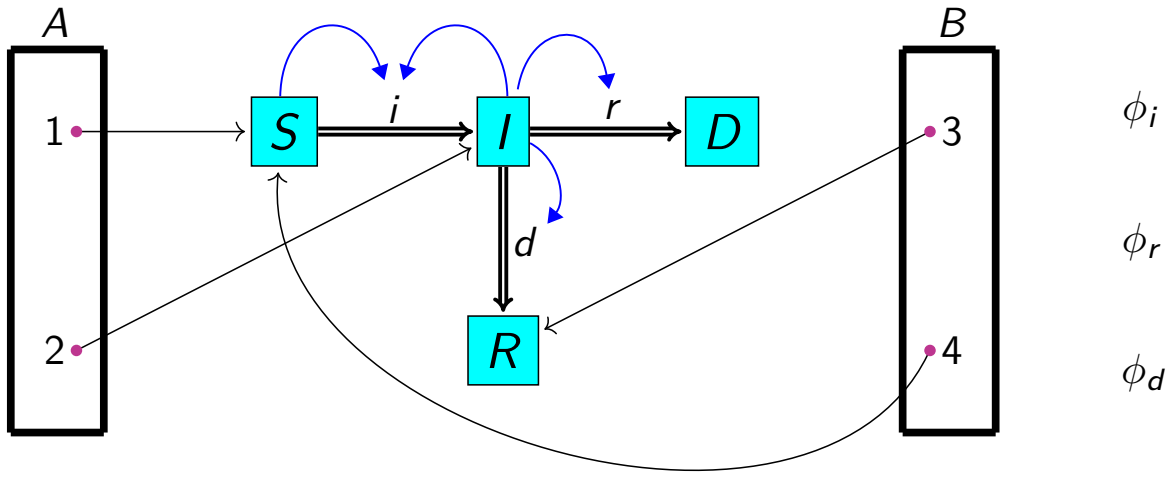
$$\frac{dS}{dt} = v_S$$

$$\frac{dI}{dt} = v_I$$

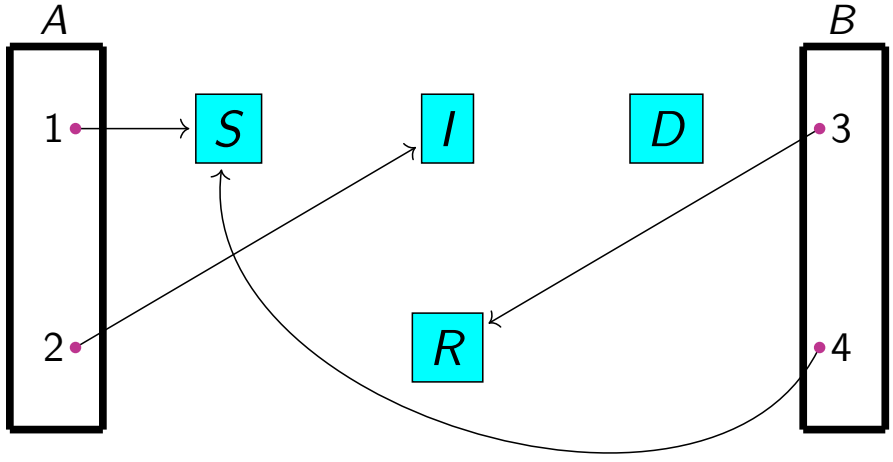
$$\frac{dR}{dt} = v_R$$

$$\frac{dD}{dt} = v_D$$

For example, Φ maps this open stock-flow diagram:



to this open dynamical system:



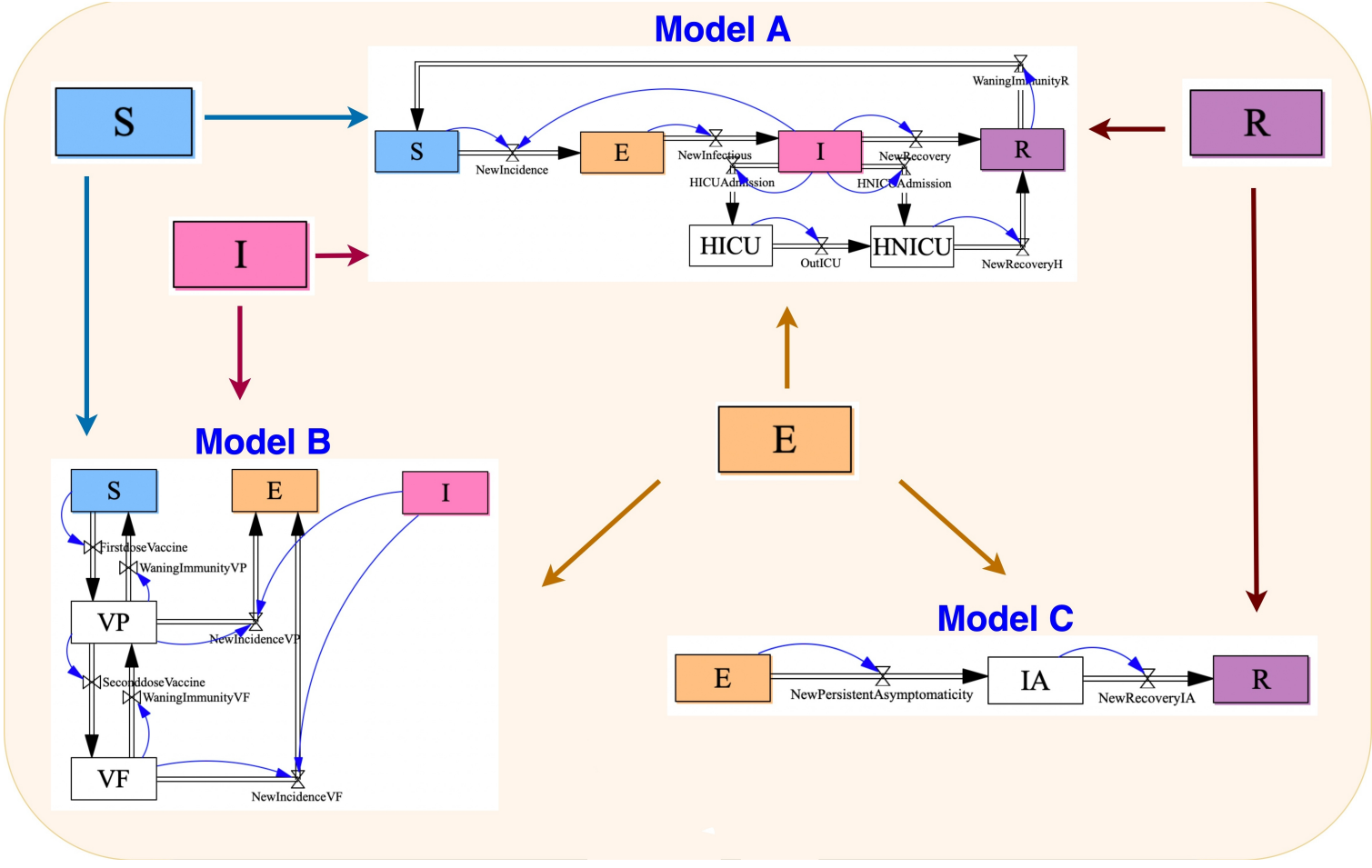
$$\frac{dS}{dt} = v_S + i_1 - o_4$$

$$\frac{dI}{dt} = v_I + i_2$$

$$\frac{dR}{dt} = v_R - o_3$$

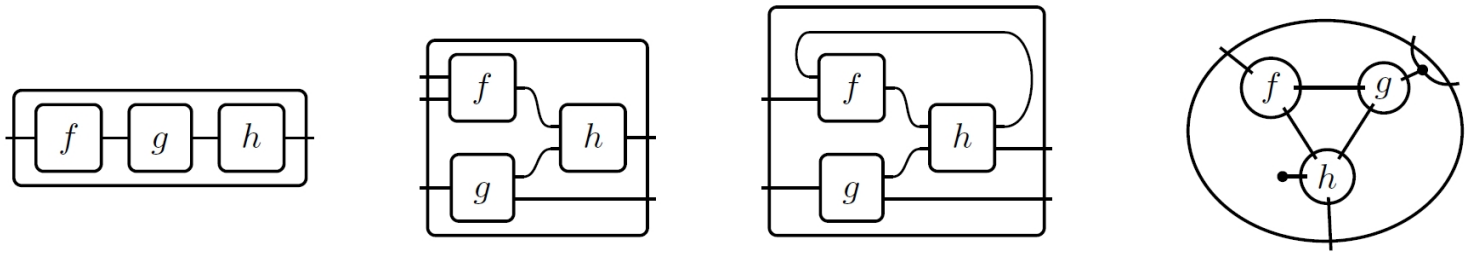
$$\frac{dD}{dt} = v_D$$

Besides composing just two stock-flow diagrams with just two boundaries each, our software can also compose stock-flow diagrams in more complex ways, like this:



To formalize this, we use the fact that many categories with open networks as morphisms are "hypergraph categories".

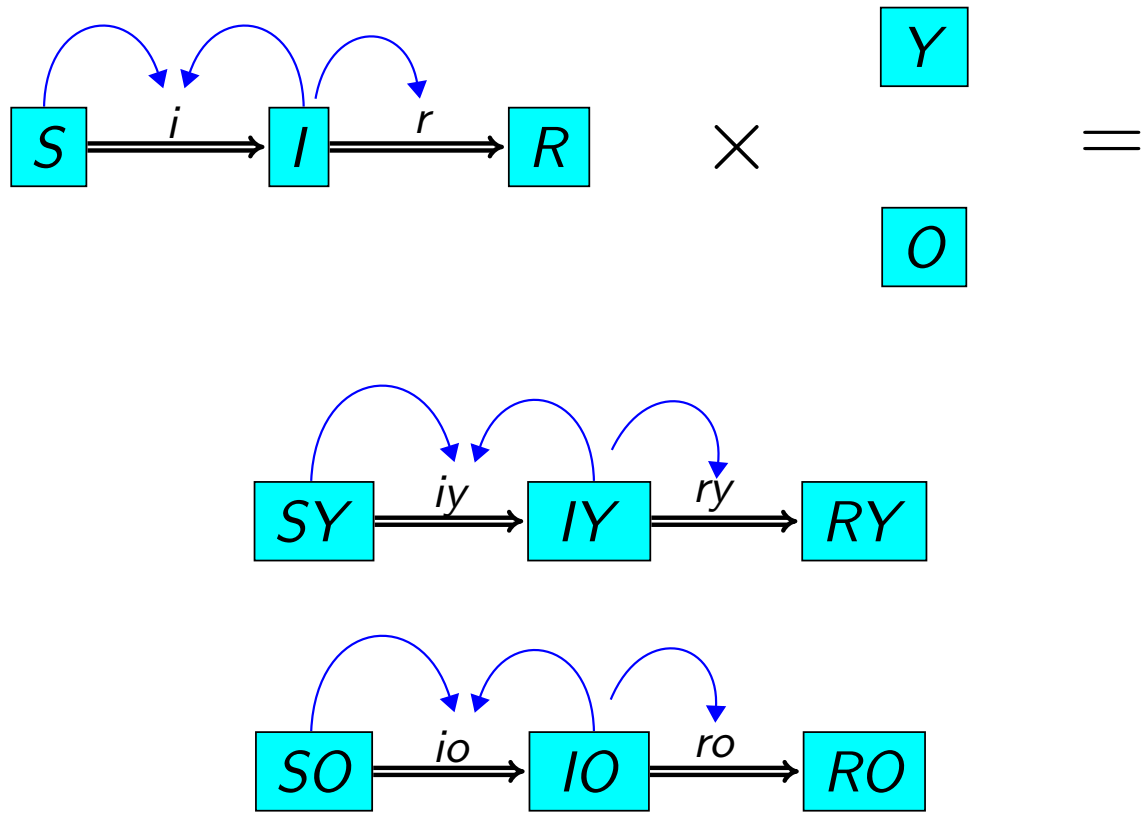
In a hypergraph category, we can compose operations in patterns described by the "operad of undirected wiring diagrams":



category | monoidal category | traced monoidal category | hypergraph category

- Brendan Fong and David Spivak, Hypergraph categories.
- JB and Kenny Courser, Structured cospans, Thm. 3.12.

Our software can also "stratify" stock-flow diagrams — refine them by breaking stocks into more stocks. For this we use "pullbacks", e.g. products:



We created our software with Evan Patterson and Sophie Libkind at the Topos Institute, who are experts on categories for computing:




- JB, Xiaoyan Li, Sophie Libkind, Nathaniel D. Osgood and Evan Patterson, Compositional modeling with stock and flow diagrams.

We used AlgebraicJulia: a framework for high-performance scientific computing using categories. This was developed by James Fairbanks, Evan, Sophie, and many others.



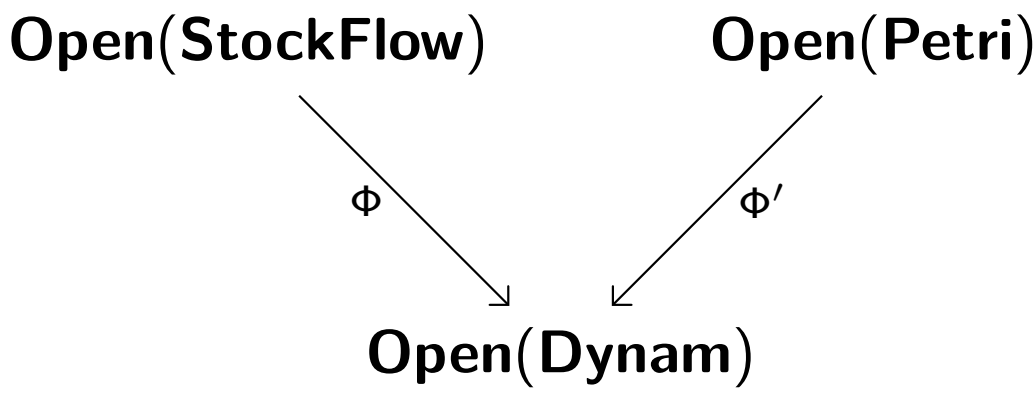
Composing open Petri nets using undirected wiring diagrams had already been implemented in AlgebraicJulia:

- Sophie Libkind, Andrew Baas, Evan Patterson and James Fairbanks, Operadic modeling of dynamical systems: mathematics and computation.

Using AlgebraicJulia we created a software package called **StockFlow**, now available on GitHub. This lets you:

- build stock-flow diagrams
- view them
- make them "open"
- compose them using the operad of undirected wiring diagrams
- apply various functors to interpret stock-flow diagrams, e.g. as systems of differential equations but also other things
- numerically solve the resulting differential equations
- stratify stock-flow diagrams using pullbacks.

Functorial semantics also lets you build models out of smaller pieces described using *different choices of syntax* — as long as they all map to some common semantics (or syntax):



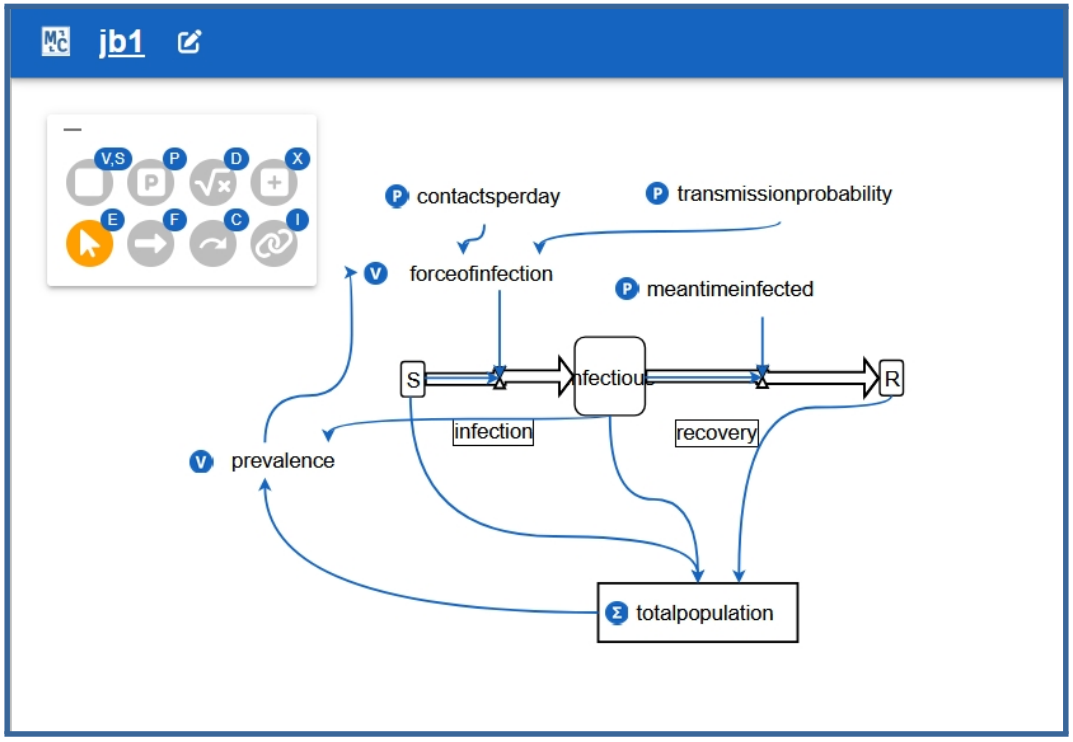
Indeed, Φ' has also been implemented in AlgebraicJulia:

- Andrew Baas, James Fairbanks, Micah Halter, Sophie Libkind and Evan Patterson, An algebraic framework for structured epidemic modeling.

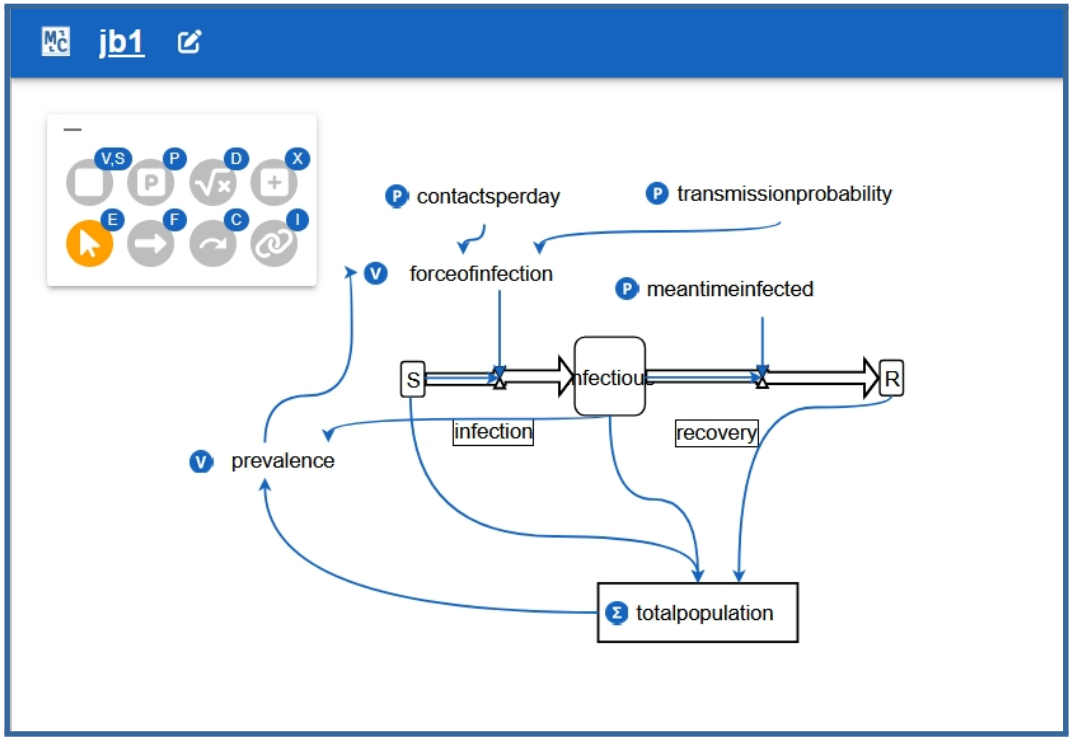
With Eric Redekopp we have made a graphical user interface for StockFlow, called **ModelCollab** — also available on GitHub.



- John C. Baez, Xiaoyan Li, Sophie Libkind, Nathaniel D. Osgood, Long Pham and Eric Redekopp, A categorical framework for modeling with stock and flow diagrams.



ModelCollab runs in your browser, so teams can collaborate to build stock-flow diagrams.



ModelCollab runs in your browser, so teams can collaborate to build stock-flow diagrams. And the great thing about ModelCollab is that you *don't need to know anything about category theory or AlgebraicJulia to use it!*

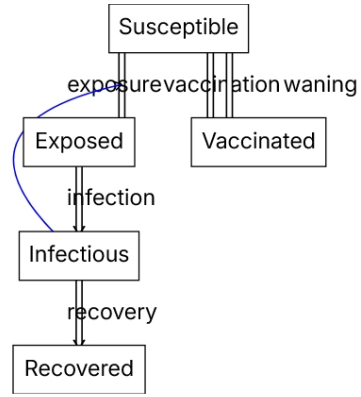
The fancy math should be black-boxed. Applied category theory can truly succeed only when it becomes inconspicuous.

More recently, Topos has moved from AlgebraicJulia to CatColab, a more advanced system based on the math of double categories. Still in development, this system is currently illustrated with examples involving stock-flow diagrams, Petri nets and other kinds of networks.

SEIRV Stock and flow

This stock-flow diagram exhibits the standard SIR model from epidemiology augmented with Exposed and Vaccinated populations: there are populations of susceptible, exposed, infectious, vaccinated, and recovered subjects, and processes flow among these stocks. A link from the infectious population to the infection process reflects that the infection rate increases as the density of infectious subjects does.

Visualization



Mass-action dynamics equations

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Susceptible} &= -r_{\mathrm{exposure}} \cdot \mathrm{Infectious} \cdot \mathrm{Susceptible} - r_{\mathrm{vaccination}} \cdot \mathrm{Susceptible} + r_{\mathrm{waning}} \cdot \mathrm{Vaccinated} \\
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Exposed} &= -r_{\mathrm{infection}} \cdot \mathrm{Exposed} + r_{\mathrm{exposure}} \cdot \mathrm{Infectious} \cdot \mathrm{Susceptible} \\
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Infectious} &= r_{\mathrm{infection}} \cdot \mathrm{Exposed} - r_{\mathrm{recovery}} \cdot \mathrm{Infectious} \\
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Recovered} &= r_{\mathrm{recovery}} \cdot \mathrm{Infectious} \\
\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Vaccinated} &= r_{\mathrm{vaccination}} \cdot \mathrm{Susceptible} - r_{\mathrm{waning}} \cdot \mathrm{Vaccinated}
\end{aligned}
$$