

Hierarchical Modeling for Phylogenetic Inference using RevBayes

Tracy A. Heath

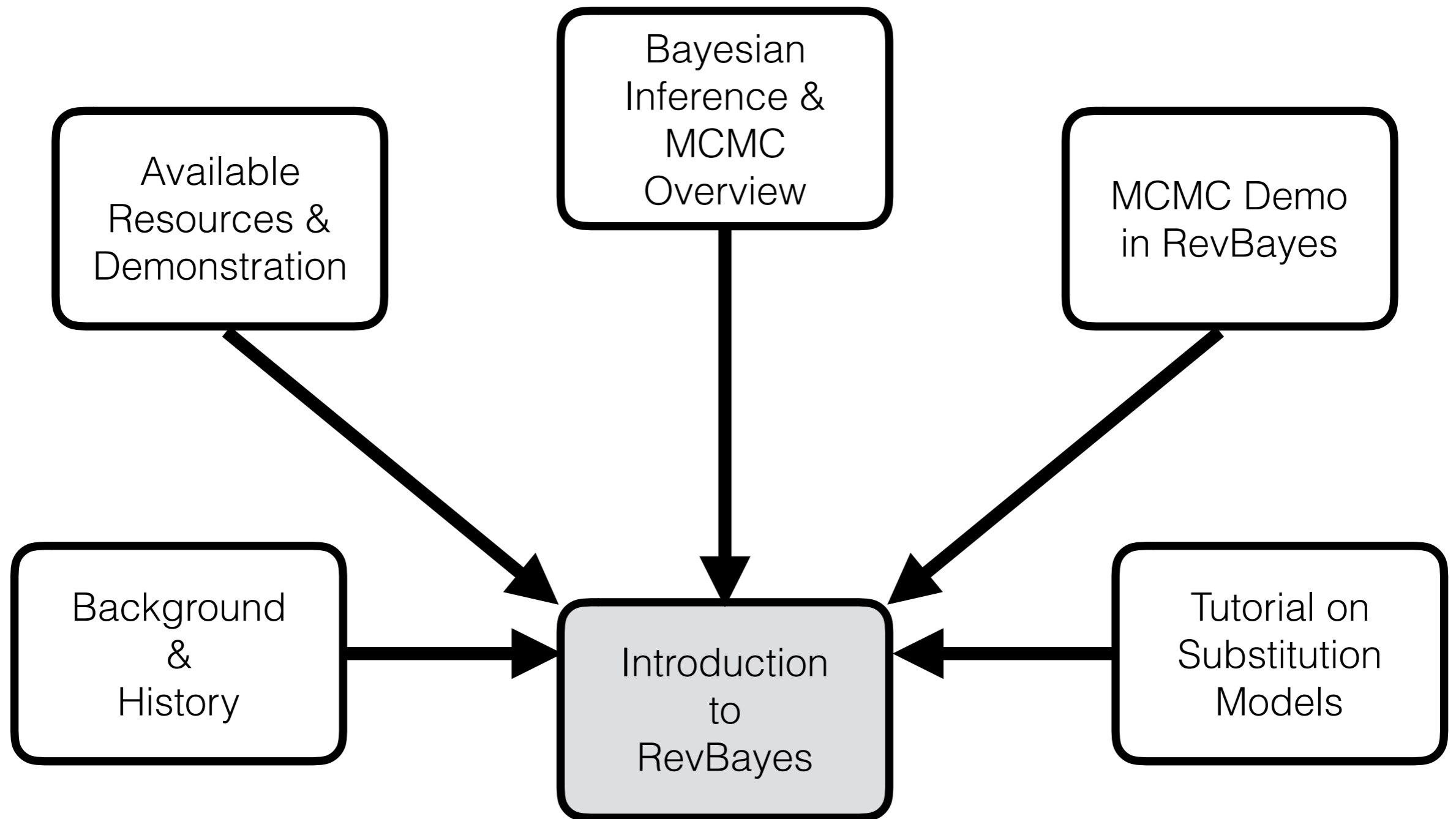
Ecology, Evolution, & Organismal Biology

Iowa State University

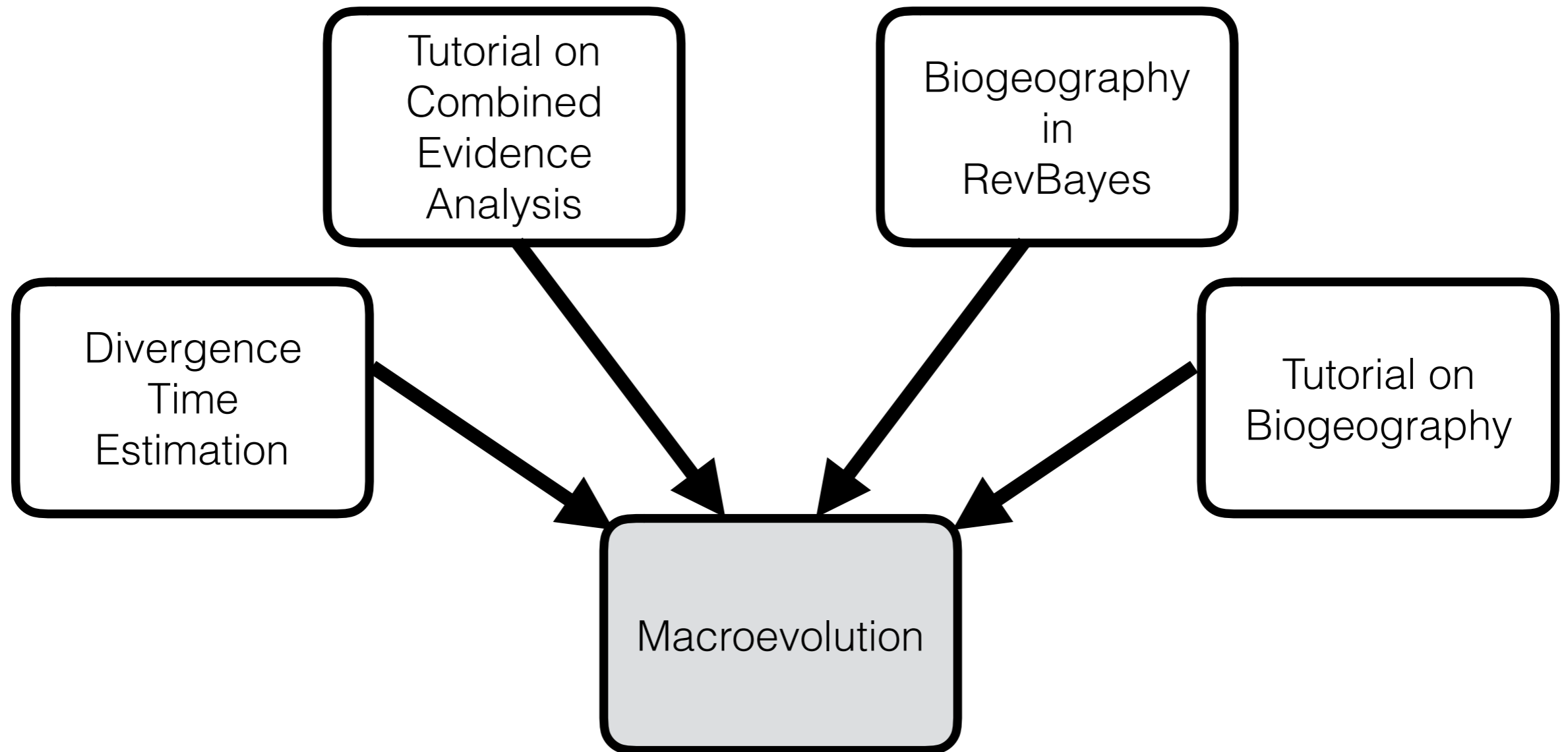
 [@trayc7](https://twitter.com/trayc7)

Bayesian Phylogenetics & Macroevolution in RevBayes Workshop
Canberra, Australia
March 26-28, 2019

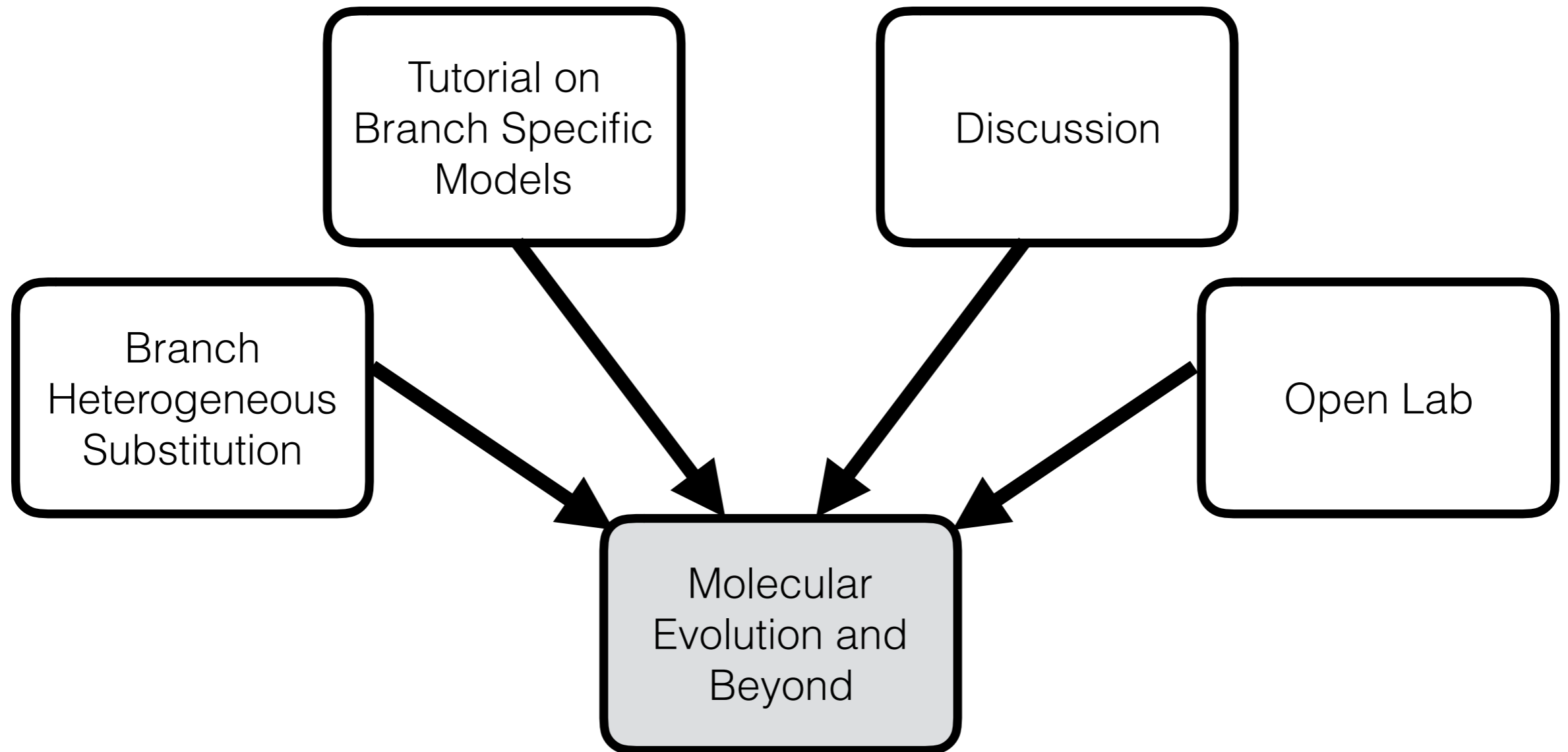
Workshop Day 1



Workshop Day 2



Workshop Day 3



RevBayes



Fully integrative Bayesian inference of phylogenetic parameters using **probabilistic graphical models** and an **interpreted language**



<http://revbayes.com>



<https://github.com/revbayes>

Höhna et al. 2016. RevBayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language. *Systematic Biology*. (doi: 10.1093/sysbio/syw021)

RevBayes: Projects



The Code

- 220,00+ lines of code
- 2,900+ source files
- 10,000+ commits

Documentation

- <http://revbayes.com>
- 30+ extensive user tutorials
- Tutorials for new developers
- Workshop materials

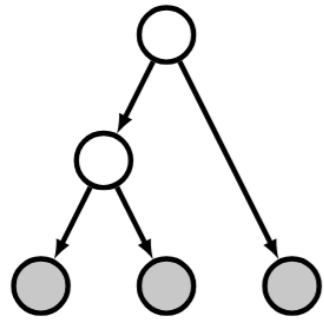
Visualization

- RevGadgets (R package)

Interfaces

- Interpreted language
- RevStudio (editor/console)
- Jupyter Kernel (notebook)
- Interactive GUI*
- Script Generator GUI*

<http://revbayes.com>

[Software](#)[Tutorials](#)[Workshops](#)[Jobs](#)[Developer](#)

RevBayes

Bayesian phylogenetic inference using probabilistic graphical models and an interpreted language

About

RevBayes provides an interactive environment for statistical computation in phylogenetics. It is primarily intended for modeling, simulation, and Bayesian inference in evolutionary biology, particularly phylogenetics. However, the environment is quite general and can be useful for many complex modeling tasks.

RevBayes uses its own language, Rev, which is a probabilistic programming language like [JAGS](#), [STAN](#), [Edward](#), [PyMC3](#), and related software. However, phylogenetic models require inference machinery and distributions that are unavailable in these other tools.

The Rev language is similar to the language used in R. Like the R language, Rev is designed to support interactive analysis. It supports both functional and procedural programming models, and makes a clear distinction between the two. Rev is also more strongly typed than R.

Core Development Team

RevBayes was designed and developed by [Sebastian Höhna](#), [Fredrik Ronquist](#) and [John P. Huelsenbeck](#). The core development team additionally includes [Michael J. Landis](#), [Bastien Boussau](#), [Tracy A. Heath](#), [Nicolas Lartillot](#), [Walker Pett](#), and [William A. Freyman](#).

[GitHub](#) | [License](#) | [Citation](#) | [Users Forum](#)

RevBayes: Tutorials



Introductions

Getting Started with RevBayes
Markov chain Monte Carlo
Molecular phylogenetics

Substitution models

Data partitioning

Divergence time estimation

node calibrations
tip dating
fossilized birth-death processes

Diversification rate estimation

episodic models
branch-heterogeneous rate variation
state-dependent rate variation

Species tree estimation

concatenated analyses
multispecies coalescent

Biogeography

biogeographic dating
DEC & epoch models

Chromosome evolution

Continuous trait evolution

Discrete morphological evolution

Methods

MCMC Strategies
model Selection
model adequacy

<http://revbayes.com/tutorials>

Why RevBayes?

- There is a clear need for more flexible statical software for phylogenetic analysis
- Flexibility is needed for both users and developers to enable analysis under new complex models



Challenges of Statistical Programming

MrBayes Default Model Settings



Default model settings:

Parameter	Options	Current Setting
Tratio _{pr}	Beta/Fixed	Beta(1.0,1.0)
Revmat _{pr}	Dirichlet/Fixed	Dirichlet(1.0,1.0,1.0,1.0,1.0,1.0)
Aamodel _{pr}	Fixed/Mixed	Fixed(Poisson)
Aarevmat _{pr}	Dirichlet/Fixed	Dirichlet(1.0,1.0,...)
Omegapr	Dirichlet/Fixed	Dirichlet(1.0,1.0)
Ny98omega1 _{pr}	Beta/Fixed	Beta(1.0,1.0)
Ny98omega3 _{pr}	Uniform/Exponential/Fixed	Exponential(1.0)
M3omegapr	Exponential/Fixed	Exponential
Codoncatfreqs	Dirichlet/Fixed	Dirichlet(1.0,1.0,1.0)
Statefreq _{pr}	Dirichlet/Fixed	Dirichlet(1.0,1.0,1.0,1.0)
Shape _{pr}	Uniform/Exponential/Fixed	Exponential(1.0)
Ratecorr _{pr}	Uniform/Fixed	Uniform(-1.0,1.0)
Pinvar _{pr}	Uniform/Fixed	Uniform(0.0,1.0)
Covswitch _{pr}	Uniform/Exponential/Fixed	Uniform(0.0,100.0)
Symdirihyper _{pr}	Uniform/Exponential/Fixed	Fixed(Infinity)
Topology _{pr}	Uniform/Constraints/Fixed/ Speciestree	Uniform
Brlen _{pr}	Unconstrained/Clock/Fixed	Unconstrained:GammaDir(1.0,0.100,1.0,1.0)

Modular Bayesian Phylogenetic Software

Several software packages in phylogenetics are moving toward a more modular framework

- reuse code
- easier to extend existing models and implement new ones
- provides a unified framework for analyses under complex models

RevBayes

BaliPhy

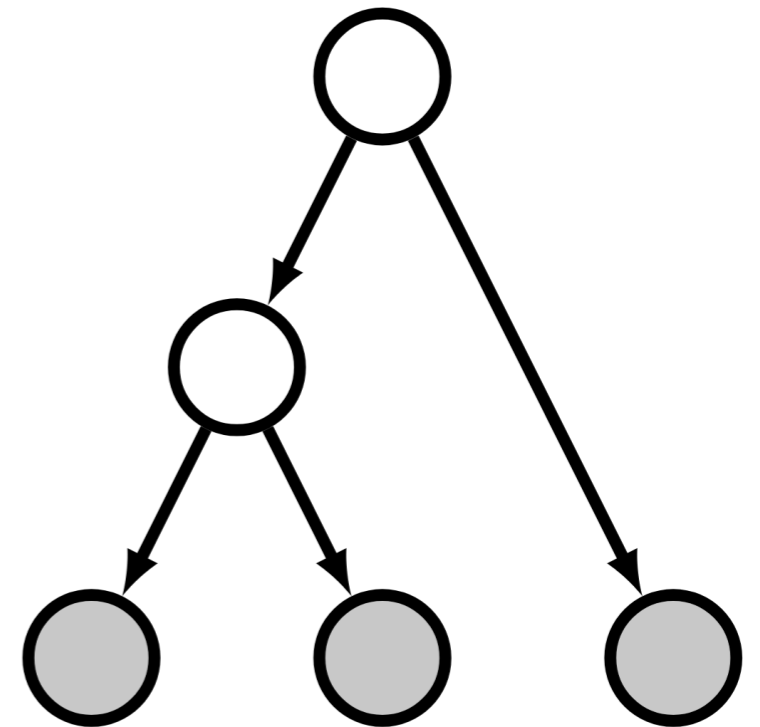
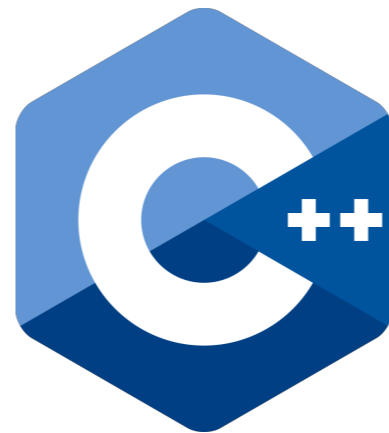
BEAST2



RevBayes Design



- C++ backend
- flexible model specification
- graphical models
- easy and intuitive to use Rev language interface



```
x <- 1  
y ~ dnExp(x)  
z := sqrt(y)
```

Rev Language



```
n_branches <- 2 * n_taxa - 2

for(i in 1:n_branches){
  branch_rates[i] ~ dnExp(1/branch_rates_mean)
  moves[mvi++] = mvScale(branch_rates[i], lambda=1.0, weight=1.0)
}

speciation_rate ~ dnExponential(10)
extinction_rate ~ dnExponential(10)

moves[mvi++] = mvScale(speciation_rate, lambda=0.01, weight=1)
moves[mvi++] = mvScale(extinction_rate, lambda=0.1, weight=1)
```

RevBayes Jupyter Kernel

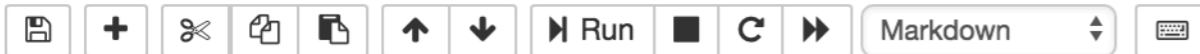


jupyter revbayes_mcmc_demo (autosaved)

Logout

File Edit View Insert Cell Kernel Help

RevBayes



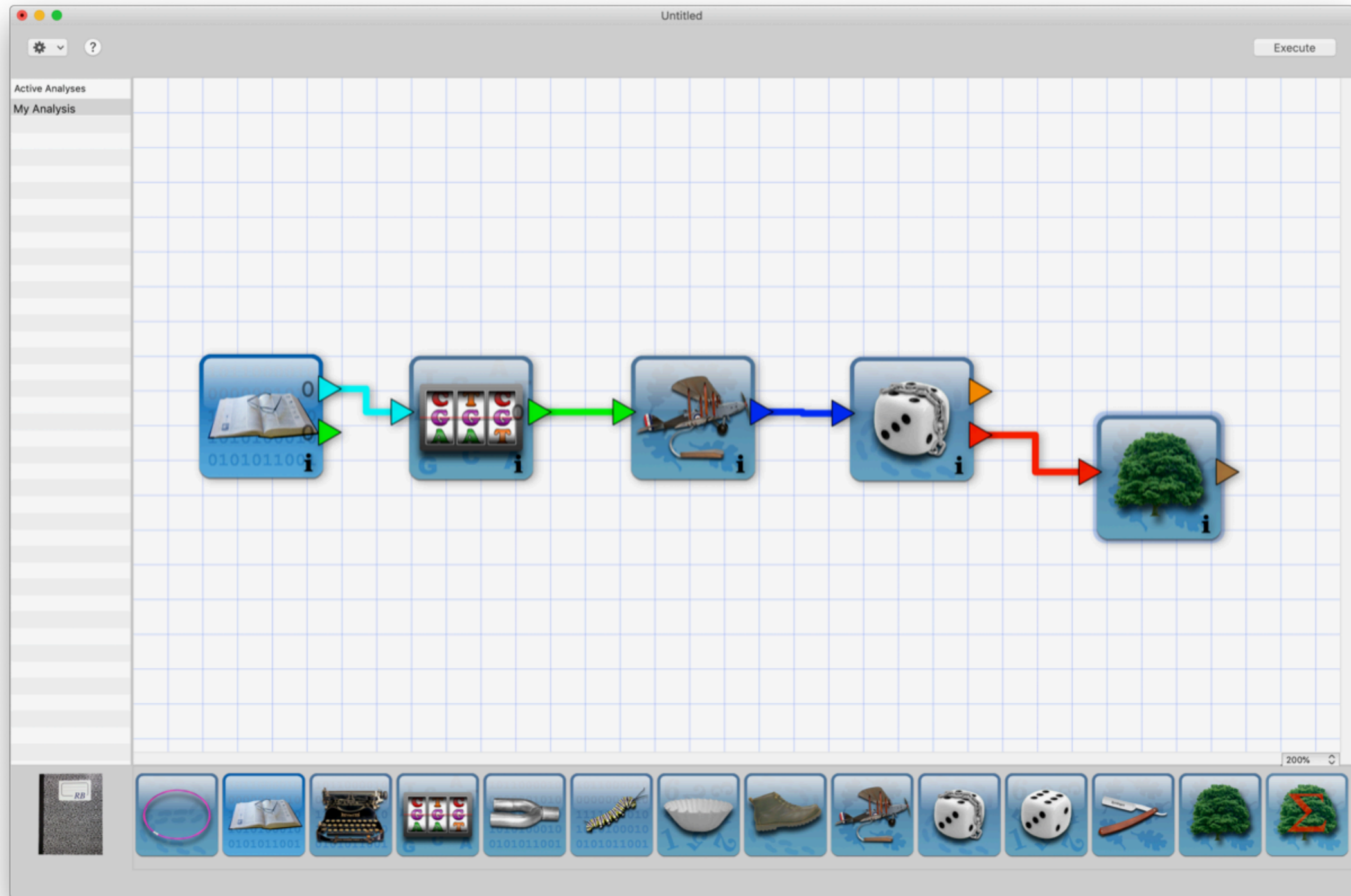
RevBayes with jupyter

This notebook demonstrates how to run a simple RevBayes analysis using jupyter. Clicking a cell will allow you to modify its contents. Note that some cells contain RevBayes code and others contain Markdown code. Pressing 'Shift+Enter' will execute or render the code within a cell. The prompt on the left hand side reading e.g. 'In [1]:' indicates the sequence of executed cells (where '[2]' is executed *after* '[1]').

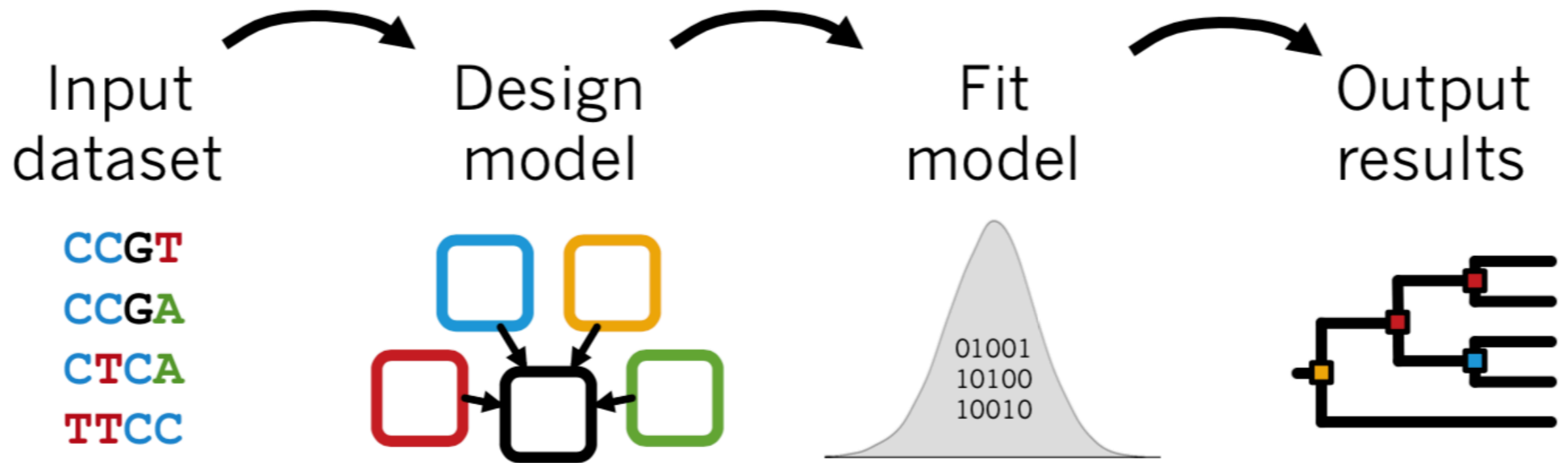
First, we'll create filepath and filename variables for IO.

```
In [ ]: # IO
dat_fp = "../example/data/"
dat_fn = dat_fp + "primates_cytb.nex"
out_fp = "../example/output/"
out_fn = out_fp + "primates"
print("path to data: " + dat_fn)
print("path to output: " + out_fn)
```

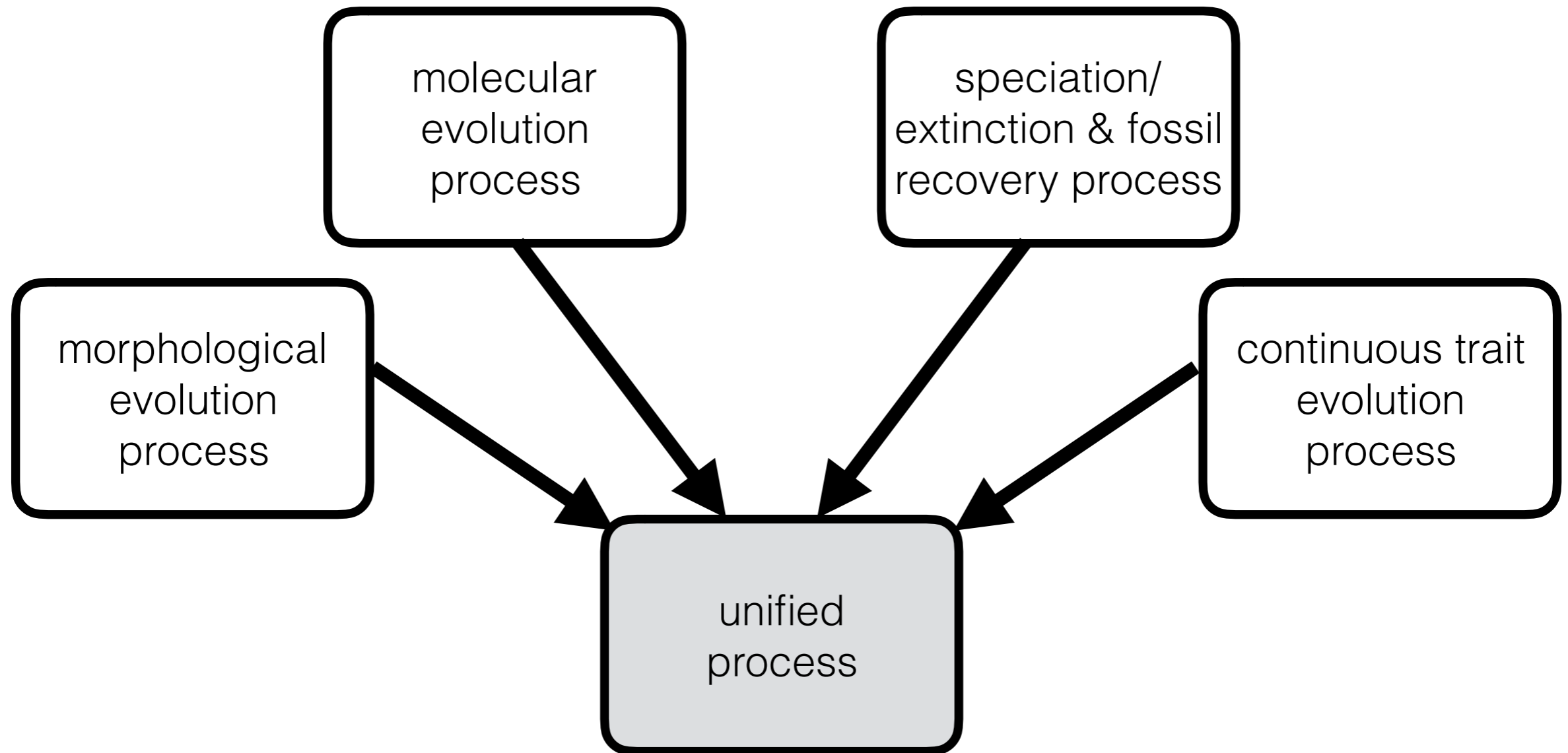

RevBayes GUI



RevBayes Inference

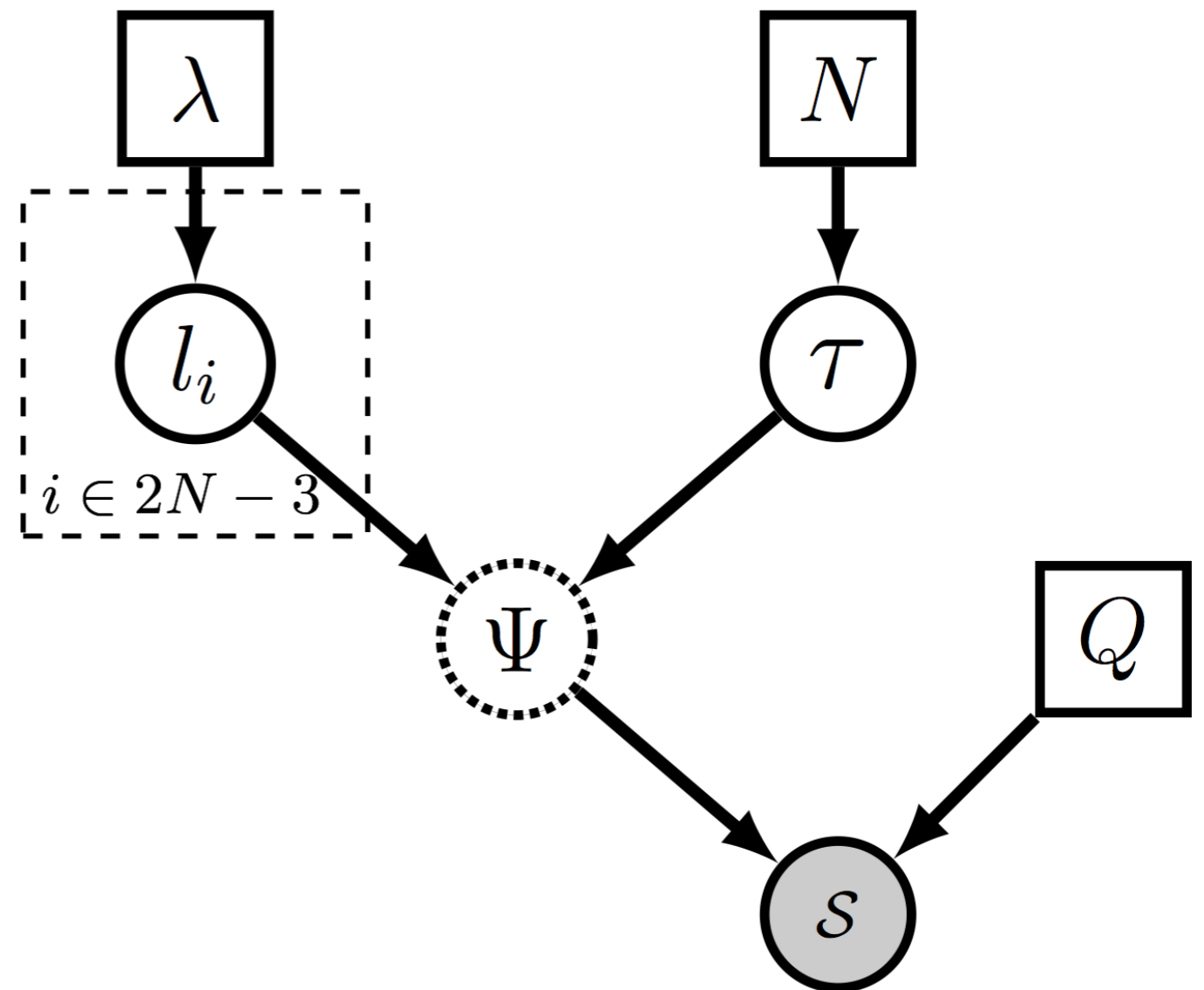


RevBayes Model Design



Probabilistic Graphical Models

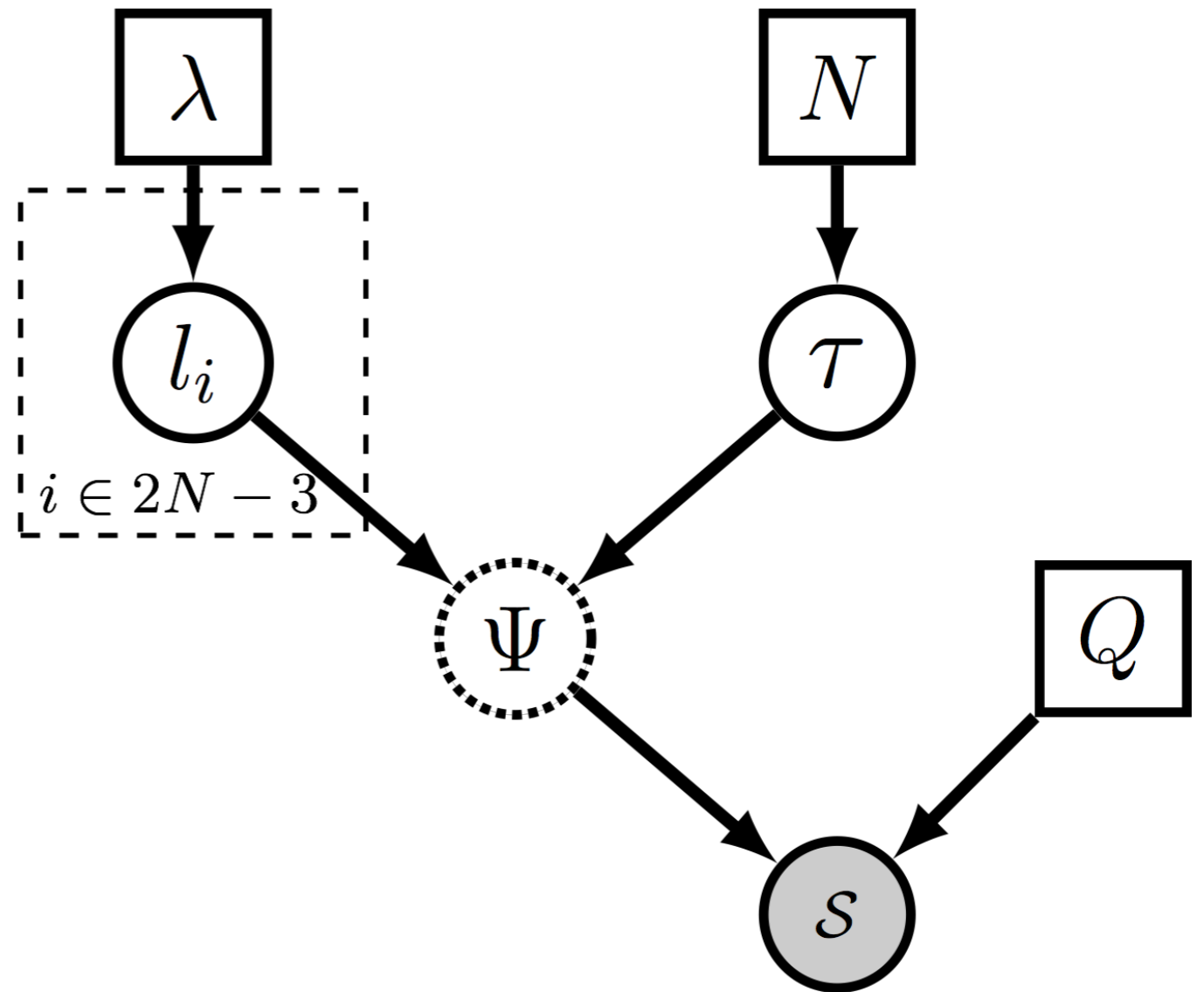
A representation of a model as a **directed acyclic graph** that exposes all of the assumptions and conditional dependence structure



Höhna et al. 2014. RevBayes: Probabilistic graphical model representation in phylogenetics. *Systematic Biology*. (doi: 10.1093/sysbio/syu039)

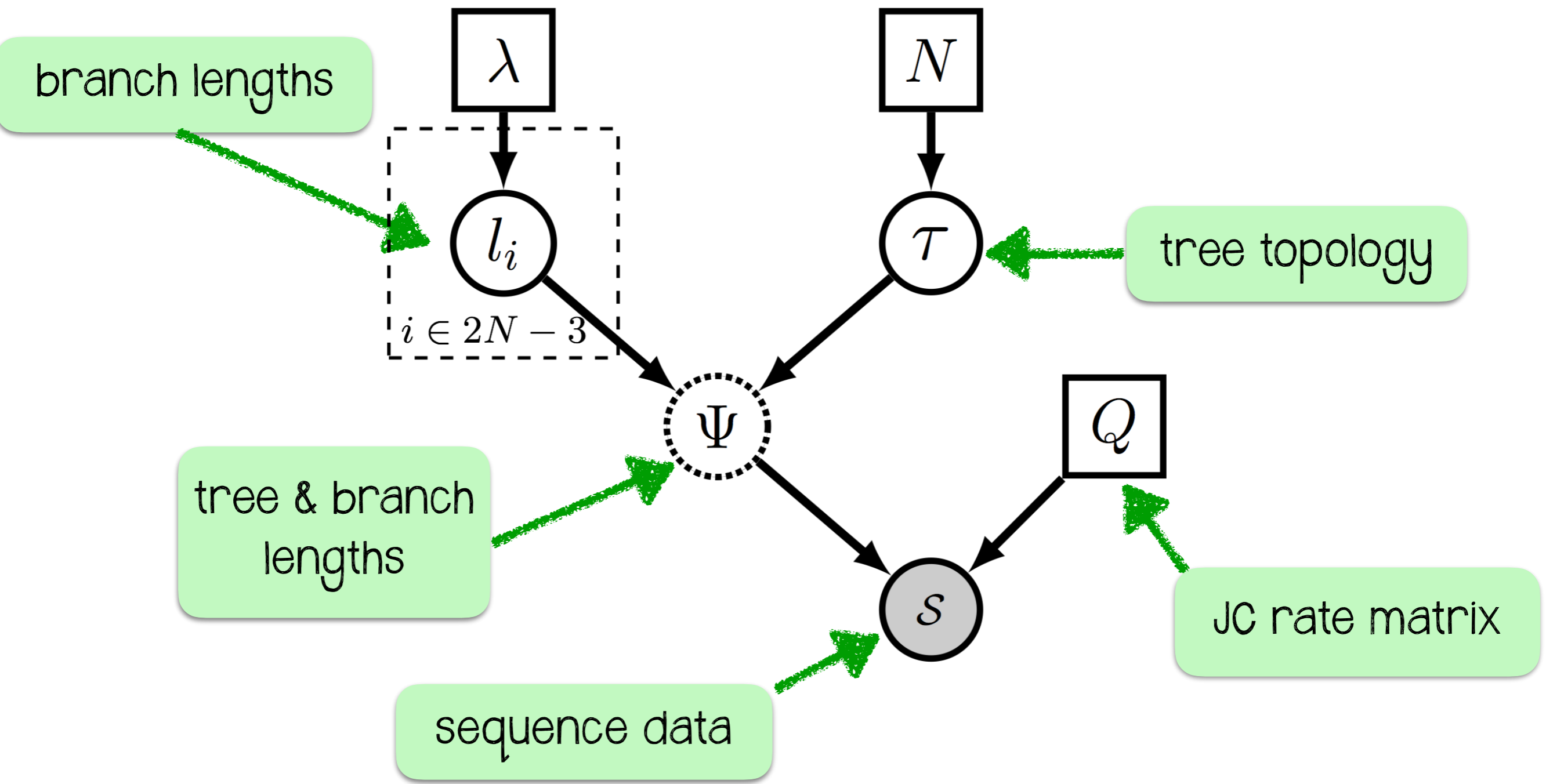
Probabilistic Graphical Models

This model: observed sequence data are generated by a Jukes-Cantor substitution process, with a uniform distribution over topologies and branch lengths drawn from an exponential distribution



Höhna et al. 2014. RevBayes: Probabilistic graphical model representation in phylogenetics. *Systematic Biology*. (doi: 10.1093/sysbio/syu039)

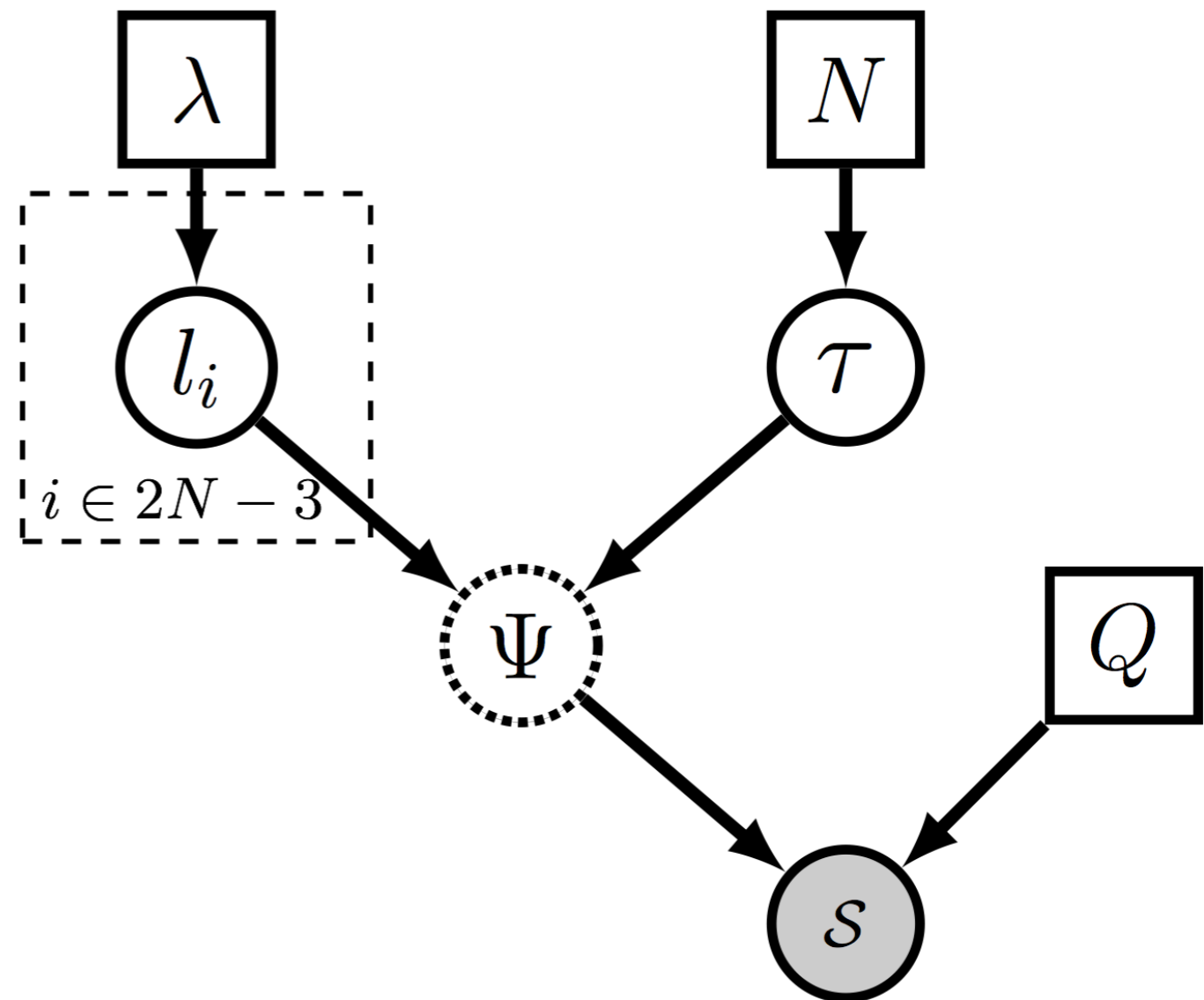
Probabilistic Graphical Models



Components of a Graphical Model

A probabilistic graphical model is a visual representation of the model structure

The elements of the graph denote the different kinds of variables in the model



Components of a Graphical Model

Constant node:

“equals”

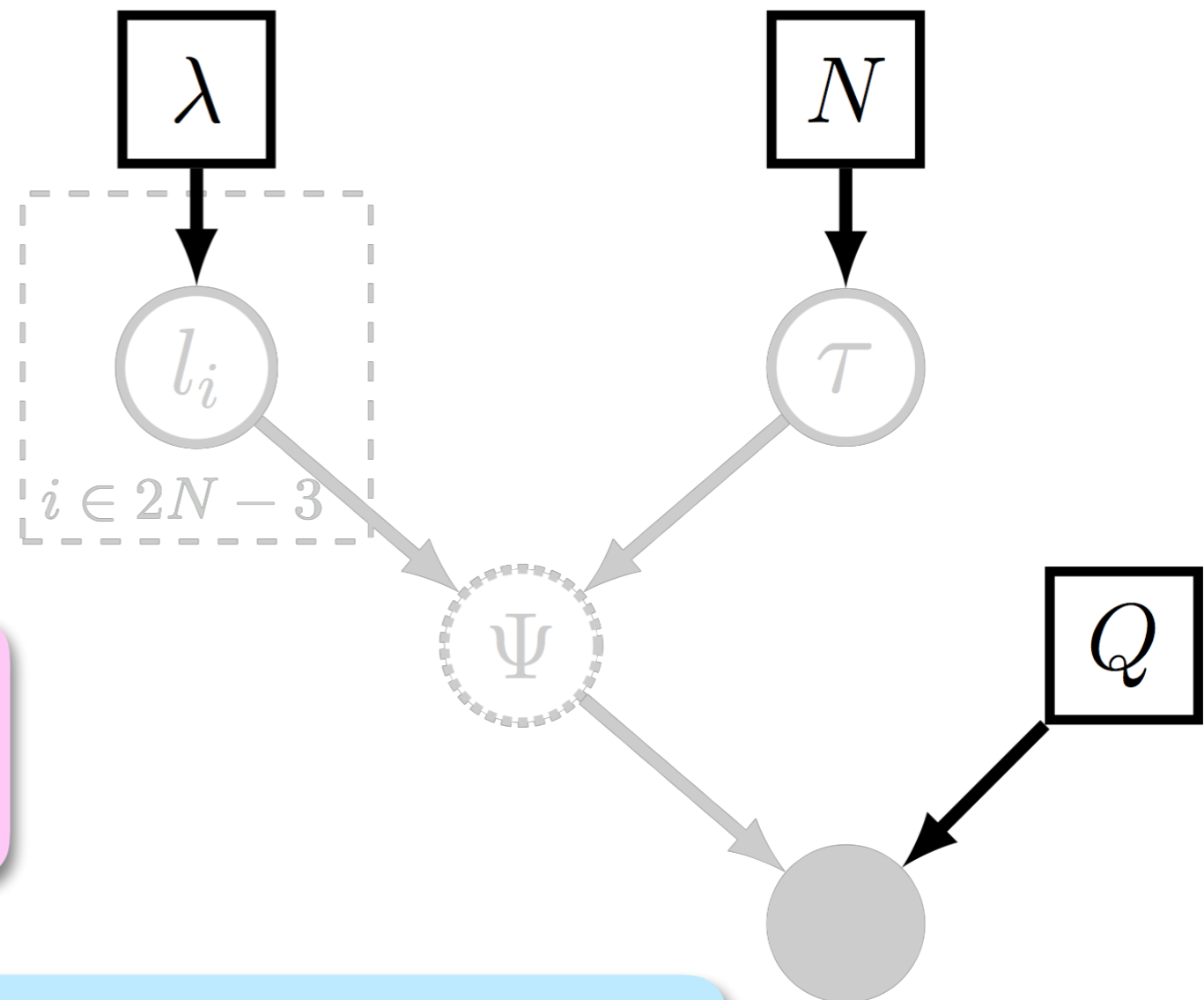
represents a fixed value that is asserted or known

value of a prior parameter

$$\lambda = 10$$

dimension of a parameter

$$N = 127$$

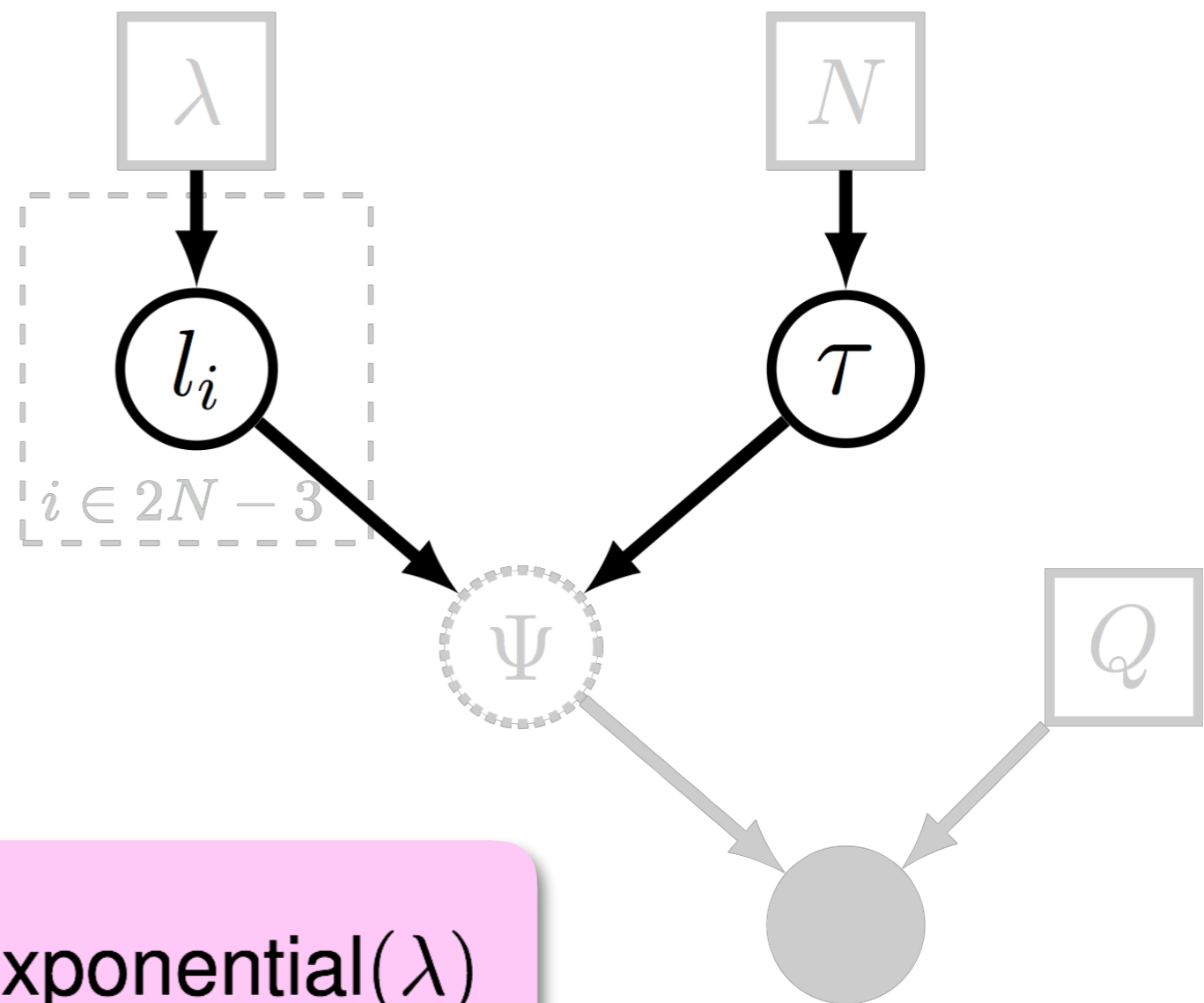


Components of a Graphical Model

Stochastic node:

“distributed by”

represents a random variable that is unknown and estimated



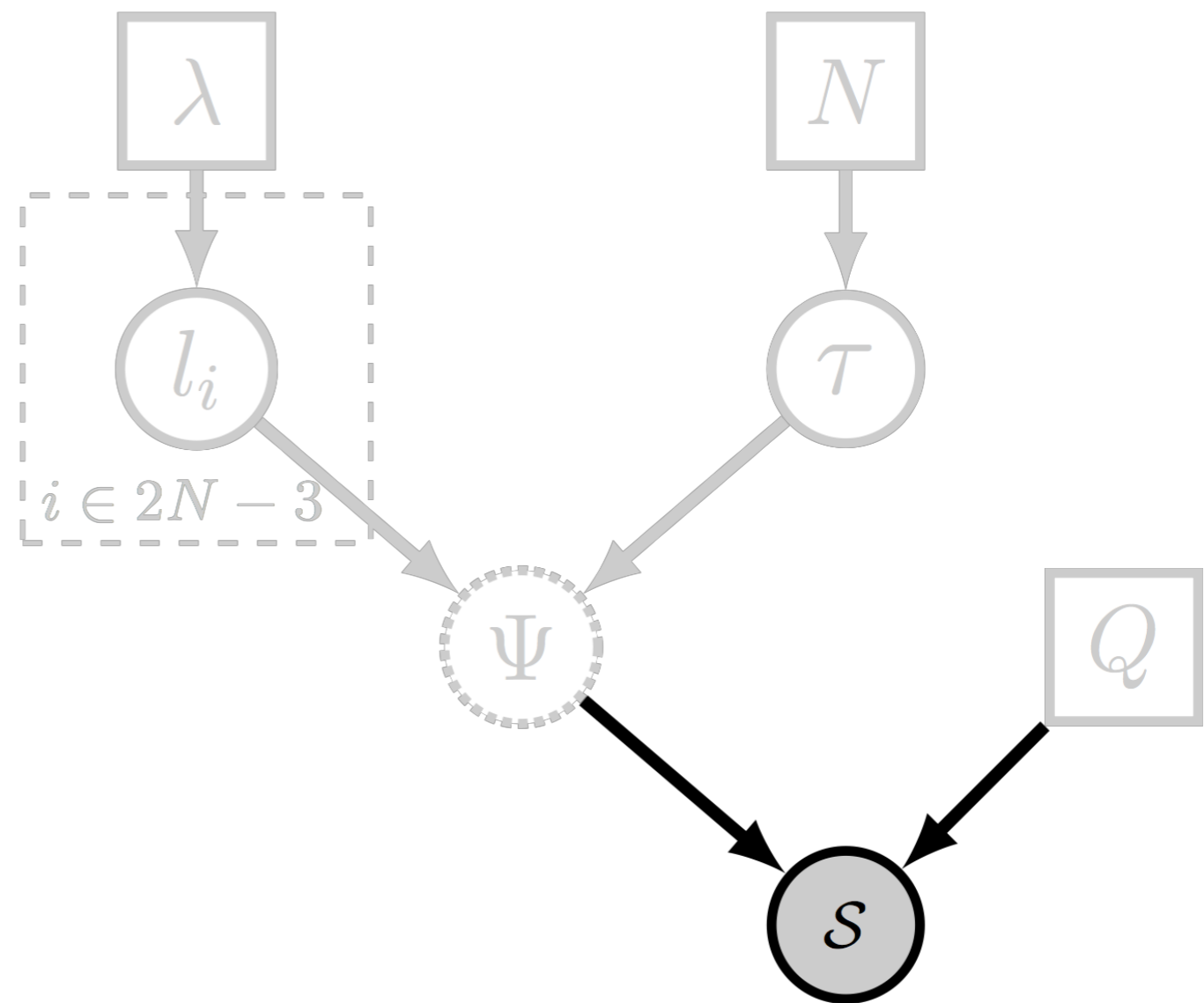
parameters that have a distribution

$$l_i \sim \text{Exponential}(\lambda)$$

Components of a Graphical Model

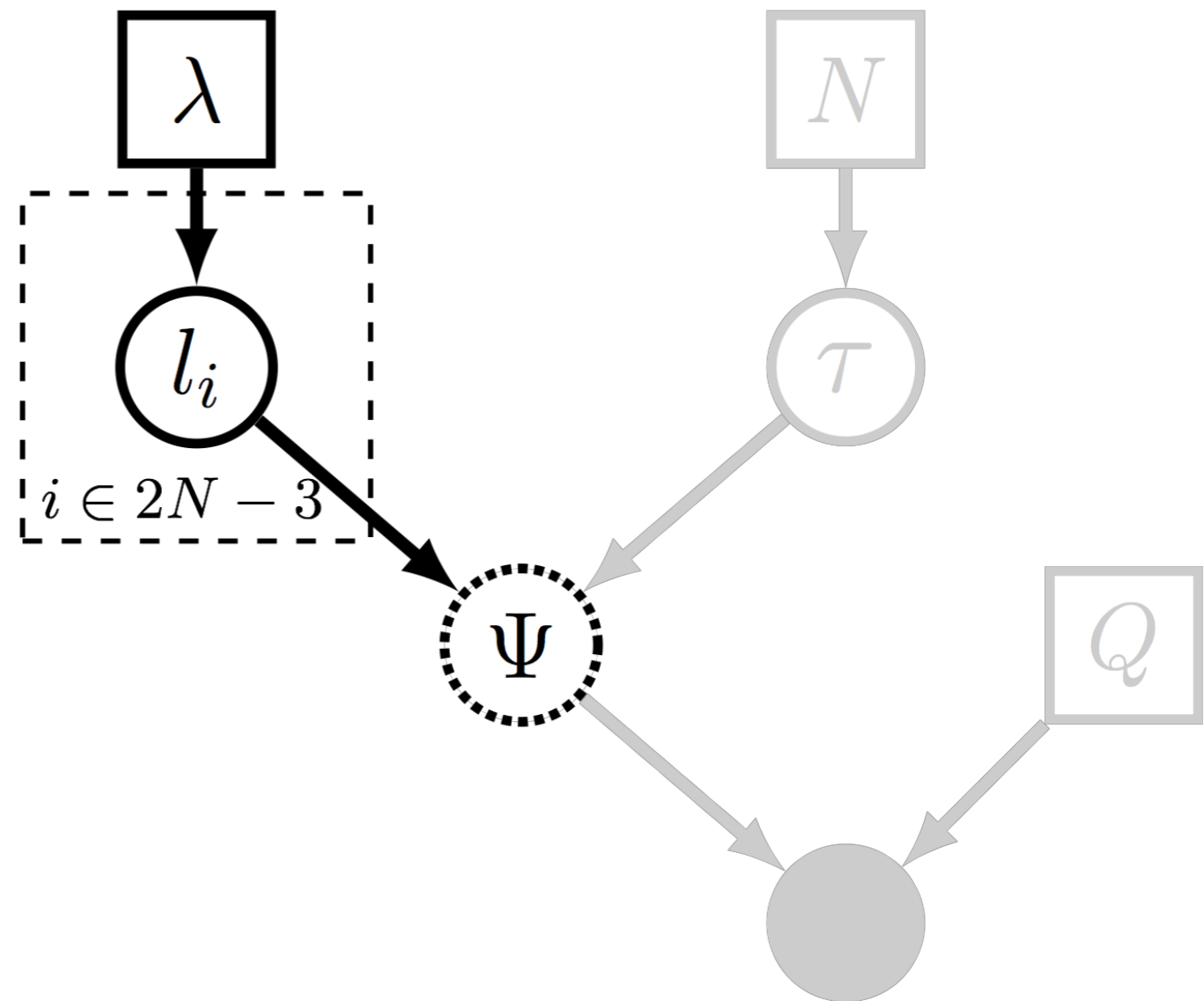
Observed stochastic node: “distributed by”

represents a random variable that is the observed outcome of the model and fixed to observed value



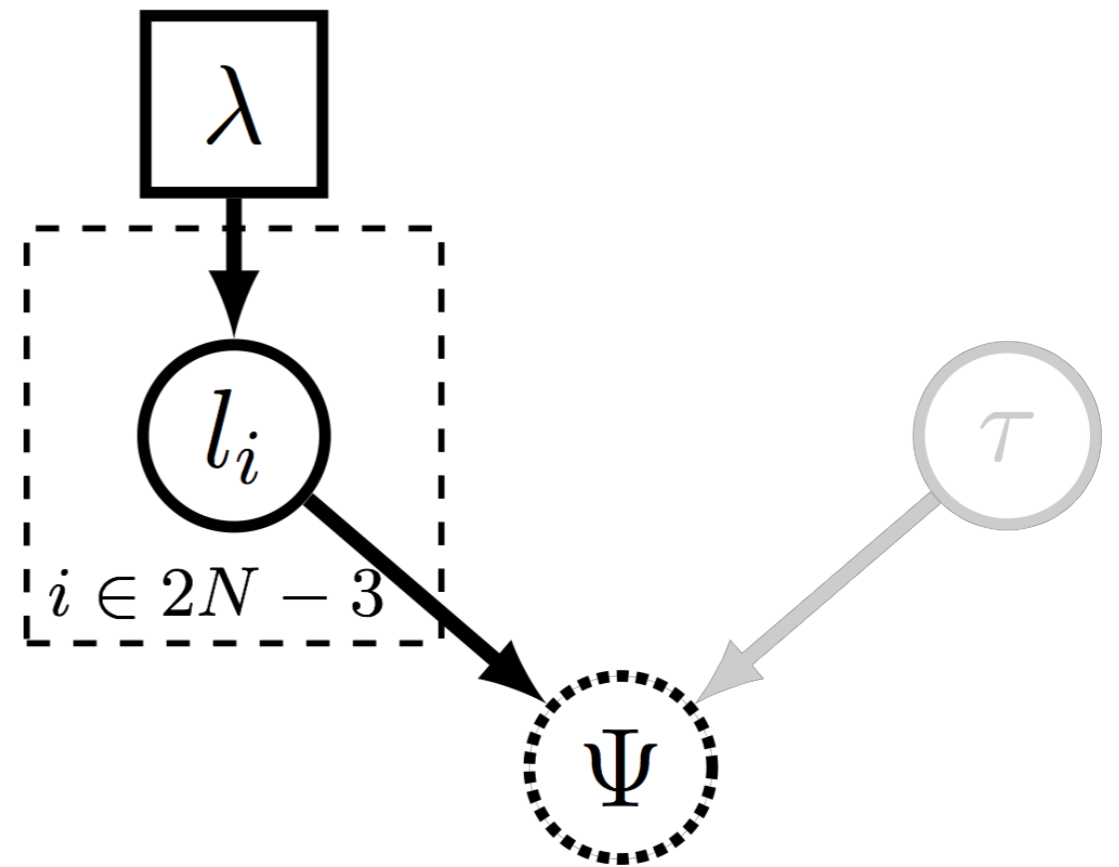
Components of a Graphical Model

Plate: repetition
repeats model
structure to simplify
visualization



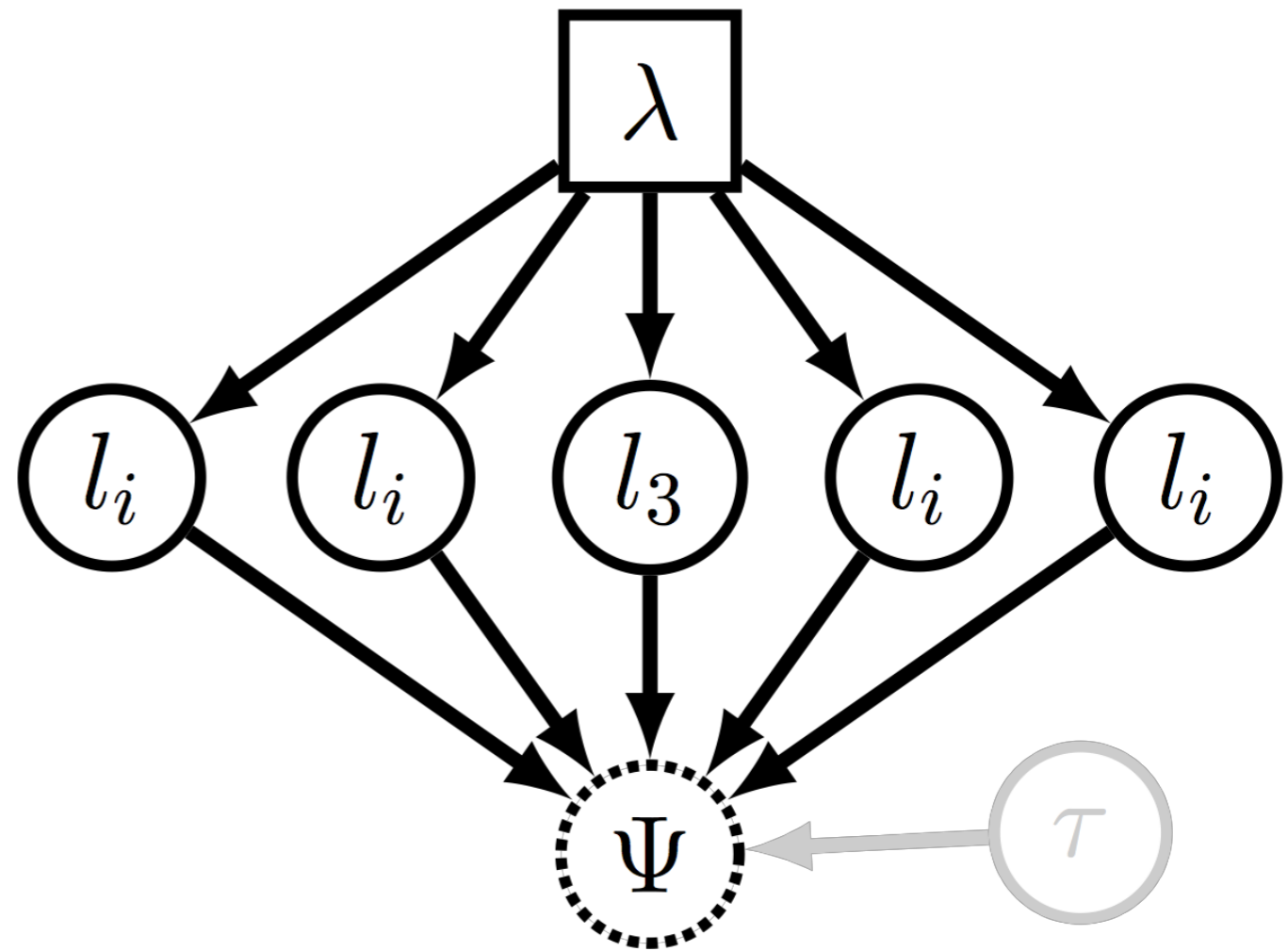
Components of a Graphical Model

Plate: repetition
repeats model
structure to simplify
visualization

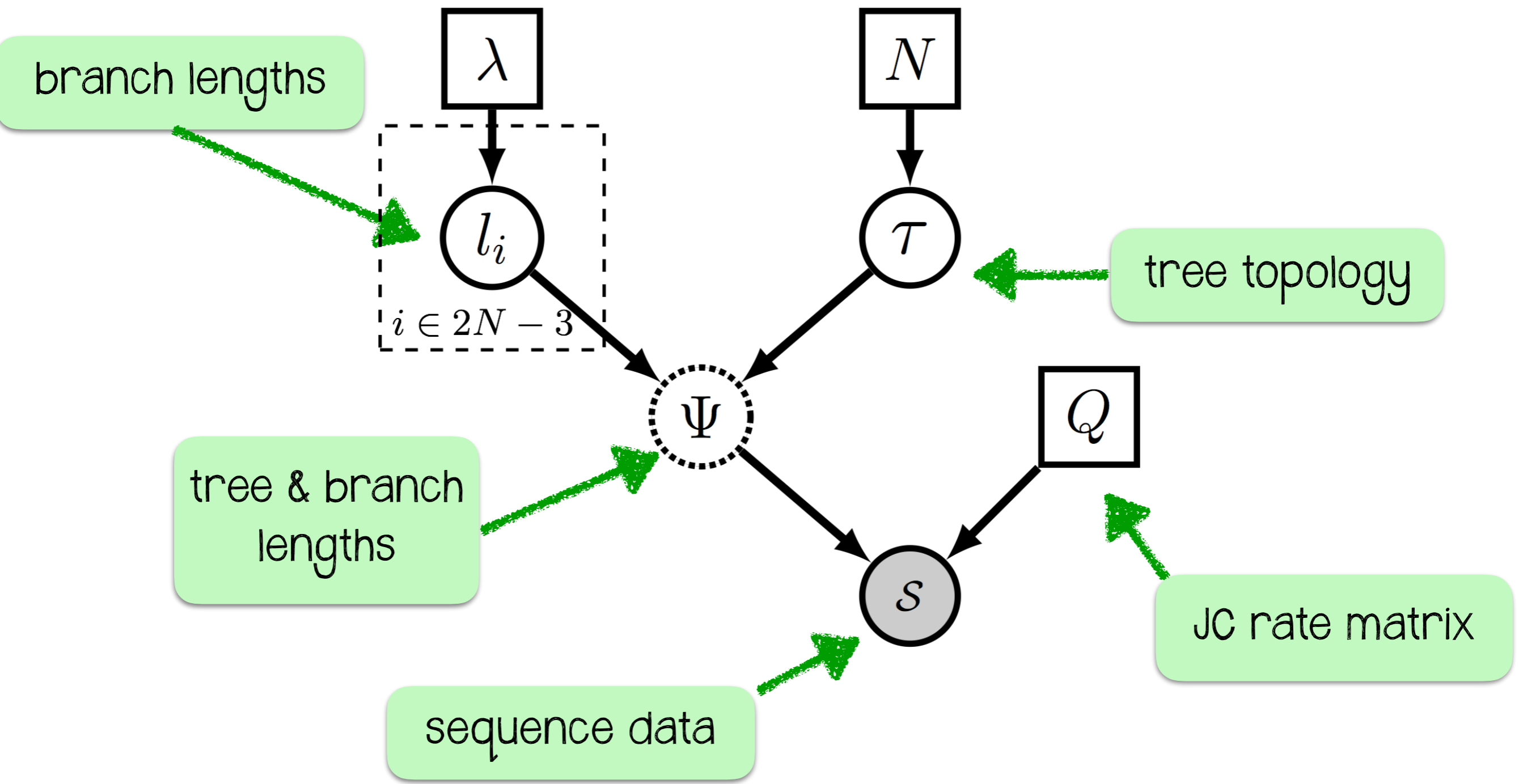


Components of a Graphical Model

Plate: repetition
repeats model
structure to simplify
visualization

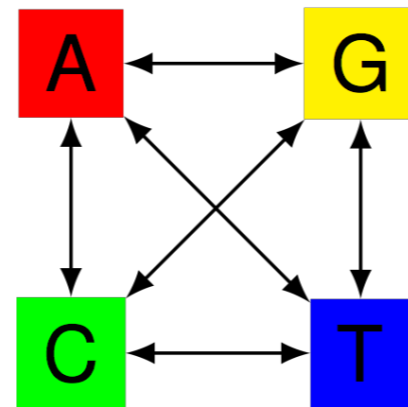
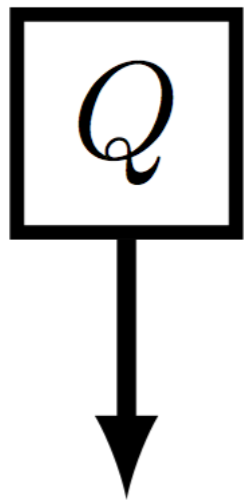


Probabilistic Graphical Models



Jukes-Cantor Model

Substitution model: with equal rates of substitution between bases and equal base frequencies

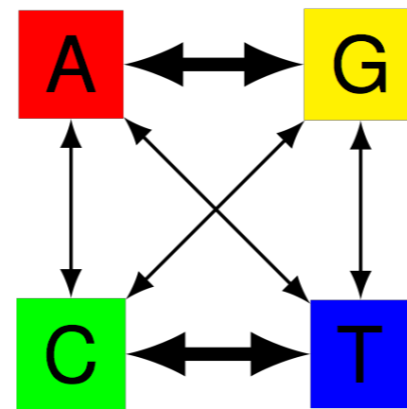
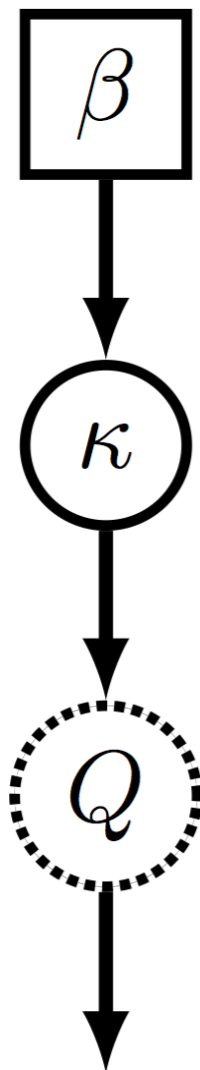


$$Q_{JC} = \begin{pmatrix} - & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & - & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & - & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & - \end{pmatrix}$$

```
Q <- fnJC(4)
```

Kimura 2-Parameter Model

Transitions happen at a higher rate than transversions and equal base frequencies

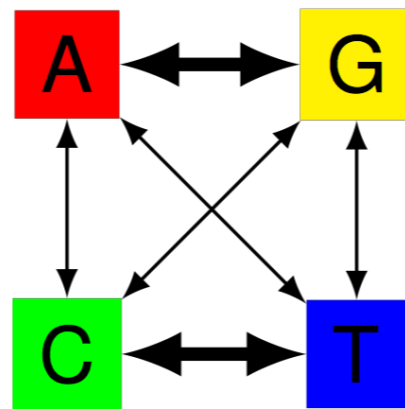
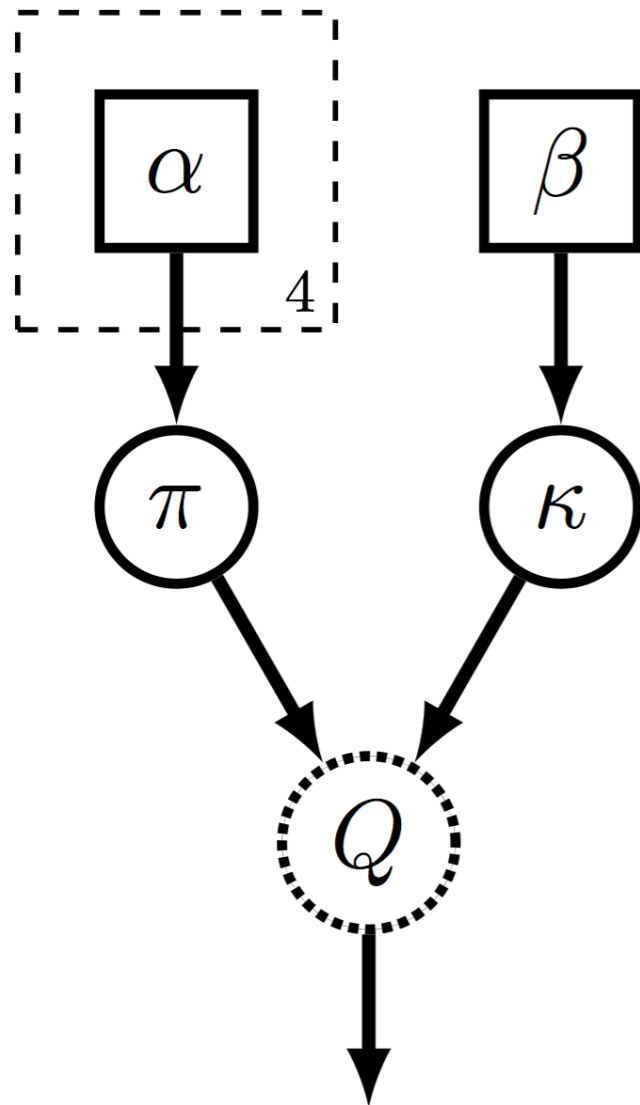


$$Q_{K80} = \begin{pmatrix} - & 1 & \kappa & 1 \\ 1 & - & 1 & \kappa \\ \kappa & 1 & - & 1 \\ 1 & \kappa & 1 & - \end{pmatrix}$$

```
beta <- 1
kappa ~ dnGamma(beta, beta)
Q := fnK80(kappa)
```

HKY Model

Transitions happen at a higher rate than transversions, with unequal base frequencies



$$Q_{HKY} = \begin{pmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{pmatrix}$$

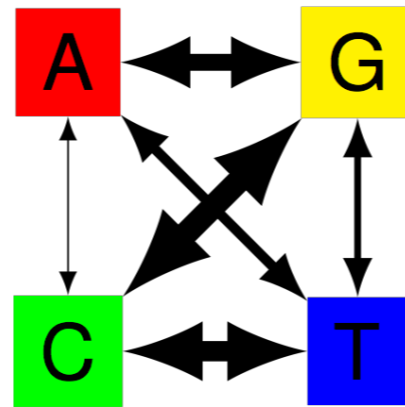
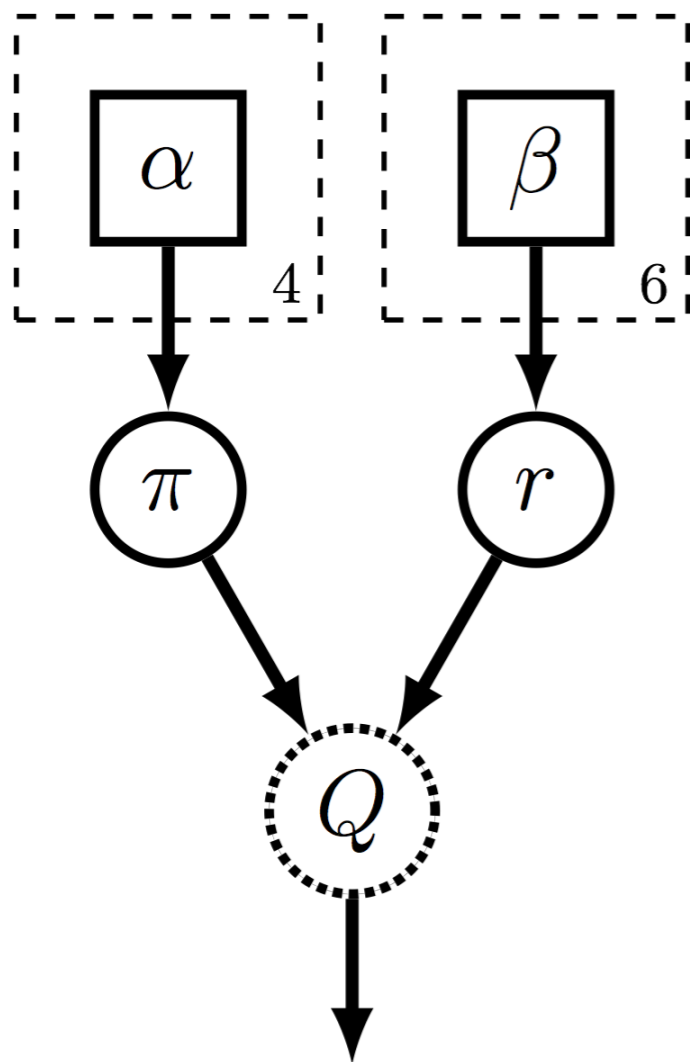
```
alpha <- [1, 1, 1, 1]
pi ~ dnDirichlet(alpha)

beta <- 1
kappa ~ dnGamma(beta, beta)

Q := fnHKY(pi, kappa)
```

General Time Reversible Model

Unequal substitution rates and unequal base frequencies



$$Q_{HKY} = \begin{pmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{pmatrix}$$

```
alpha <- [1, 1, 1, 1]
pi ~ dnDirichlet(alpha)

beta <- [1, 1, 1, 1, 1, 1]
r ~ dnDirichlet(beta)

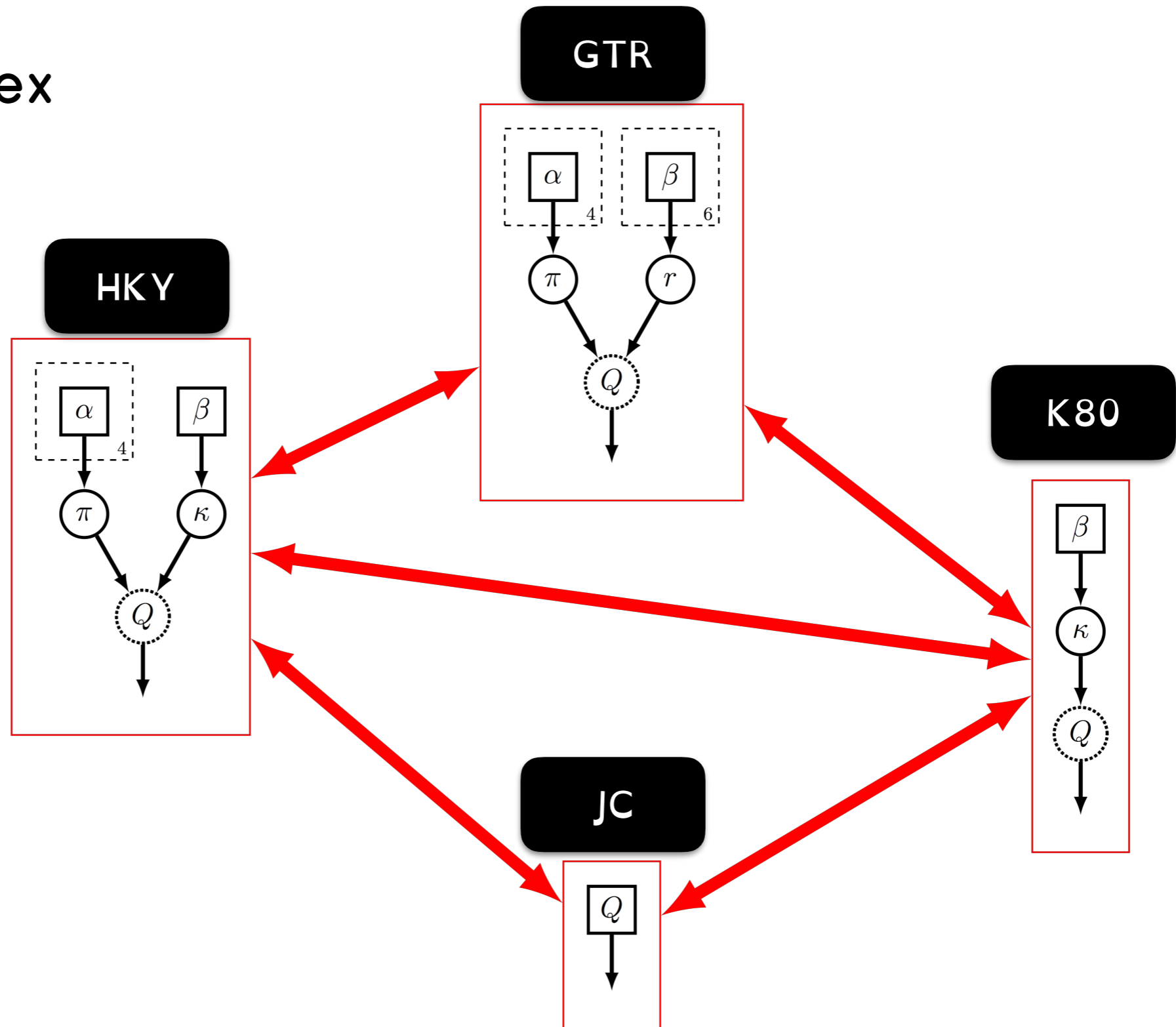
Q := fnGTR(pi, r)
```


Model Hierarchy

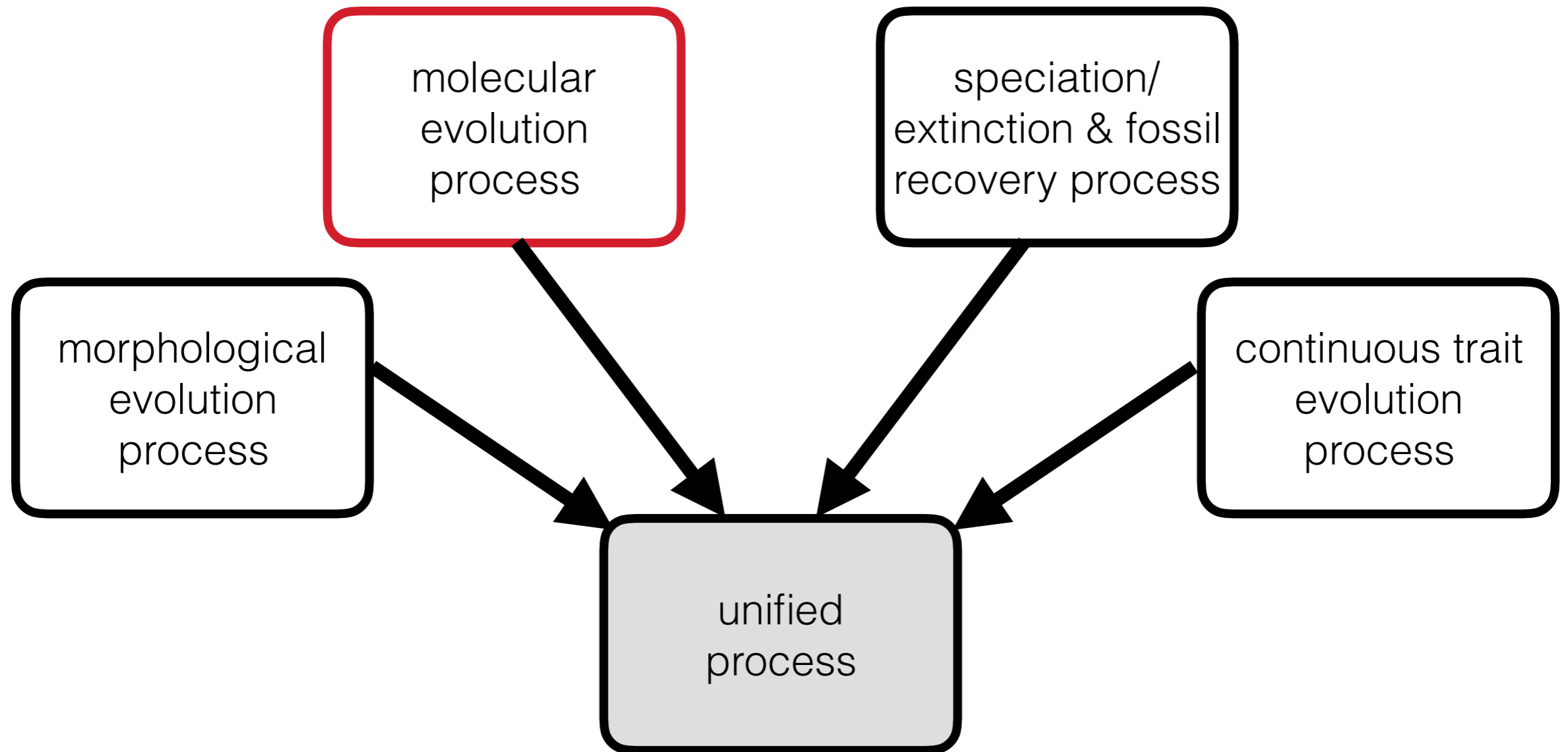
complex



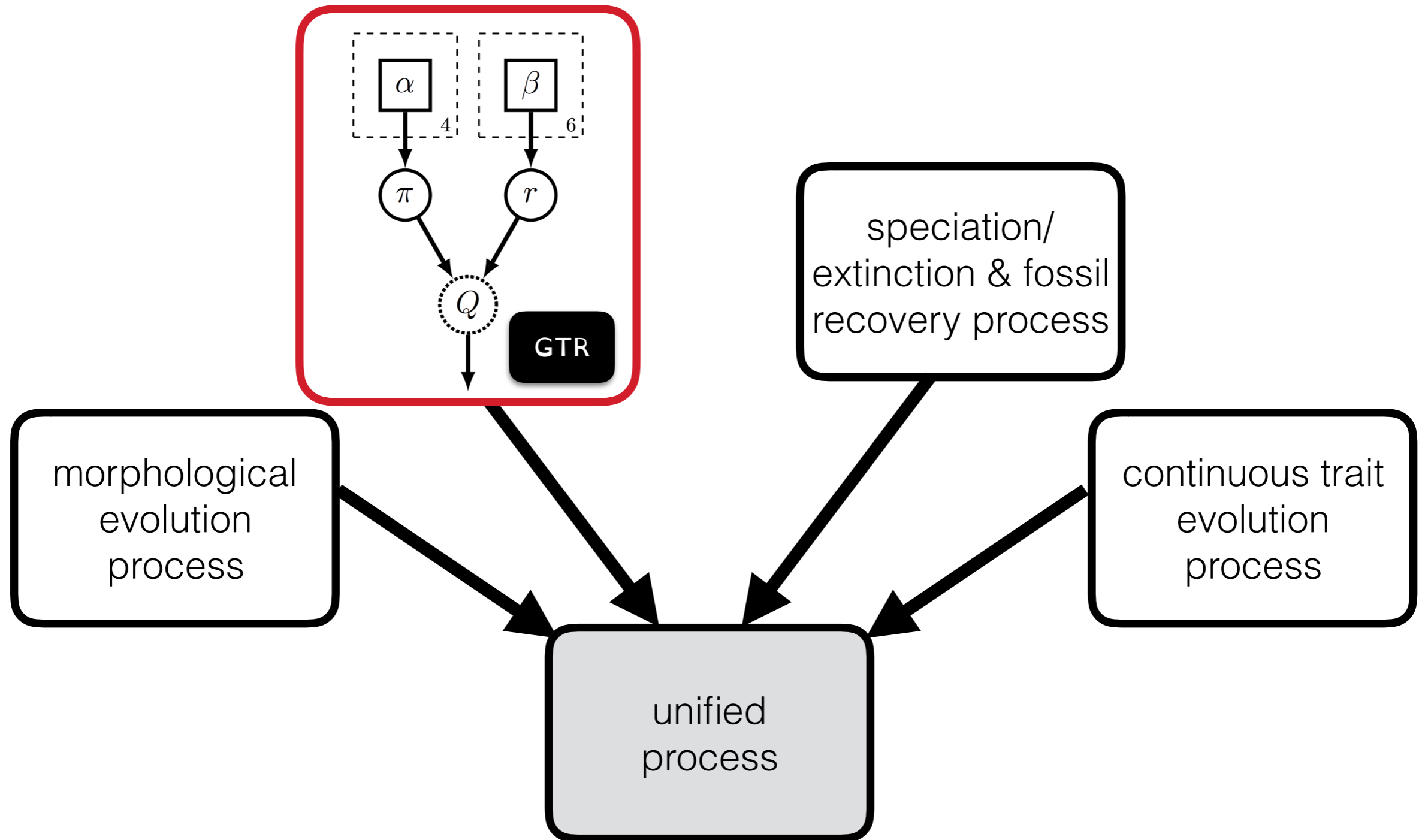
simple



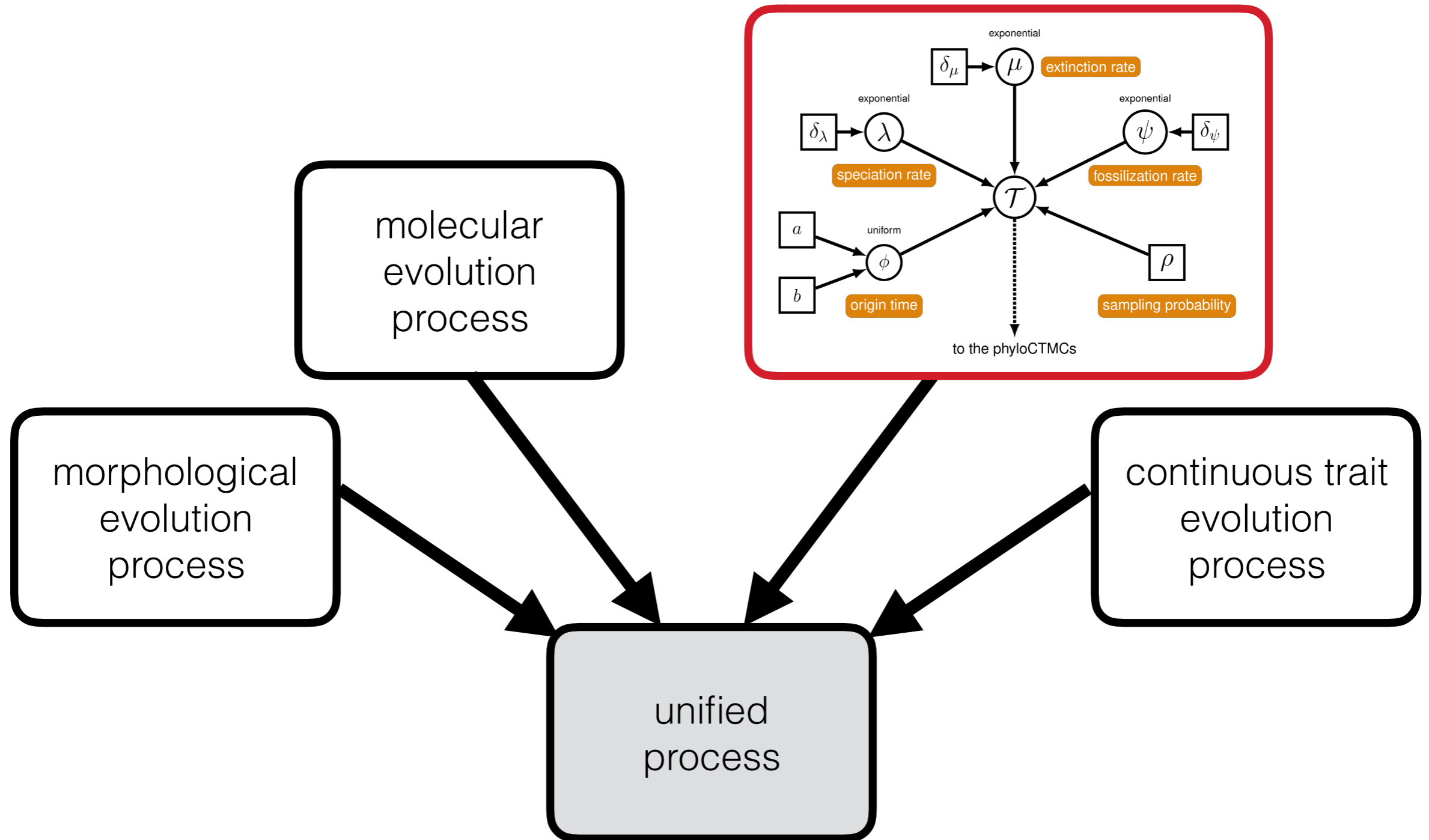
RevBayes Model Modularity



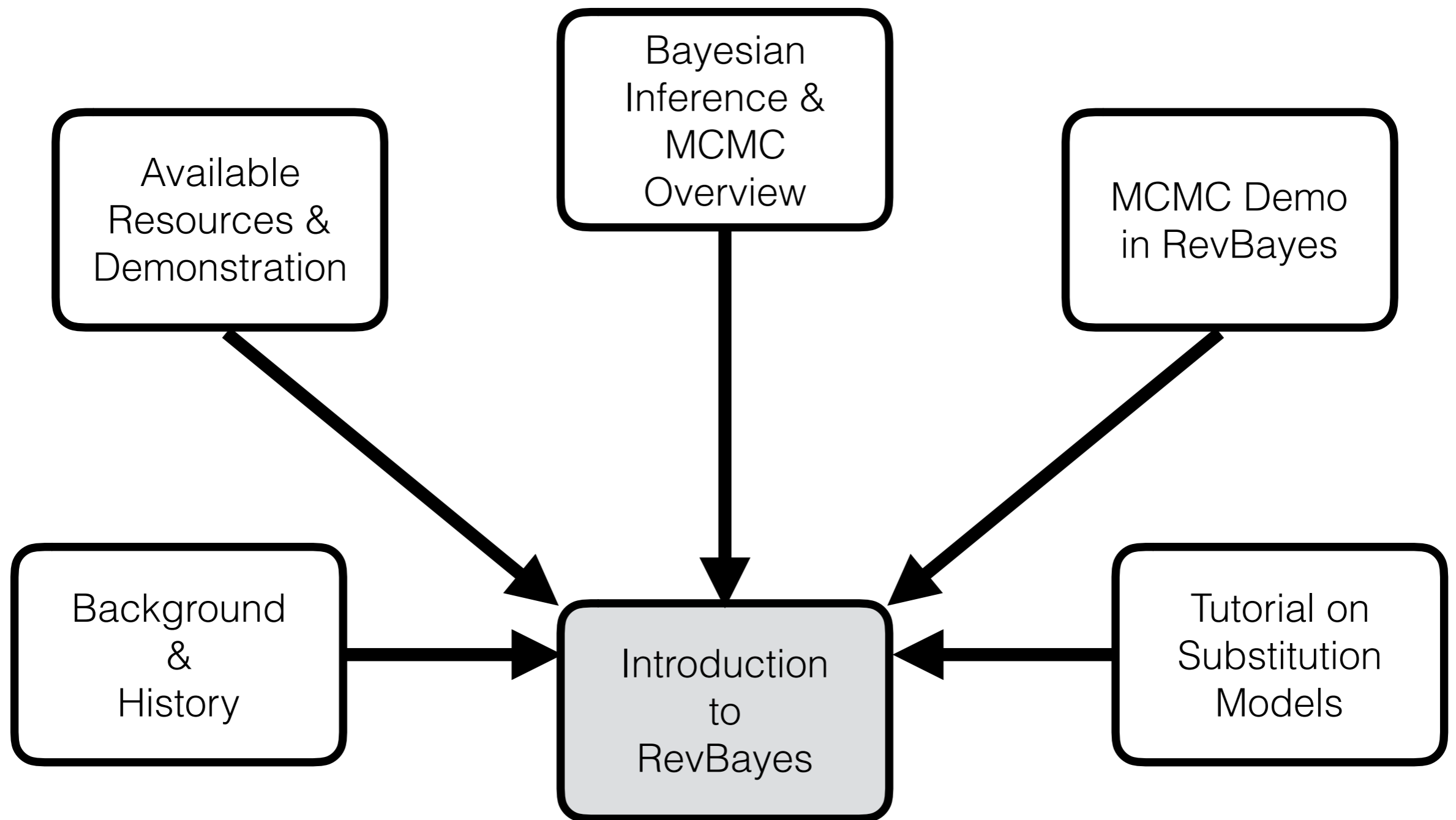
RevBayes Model Modularity



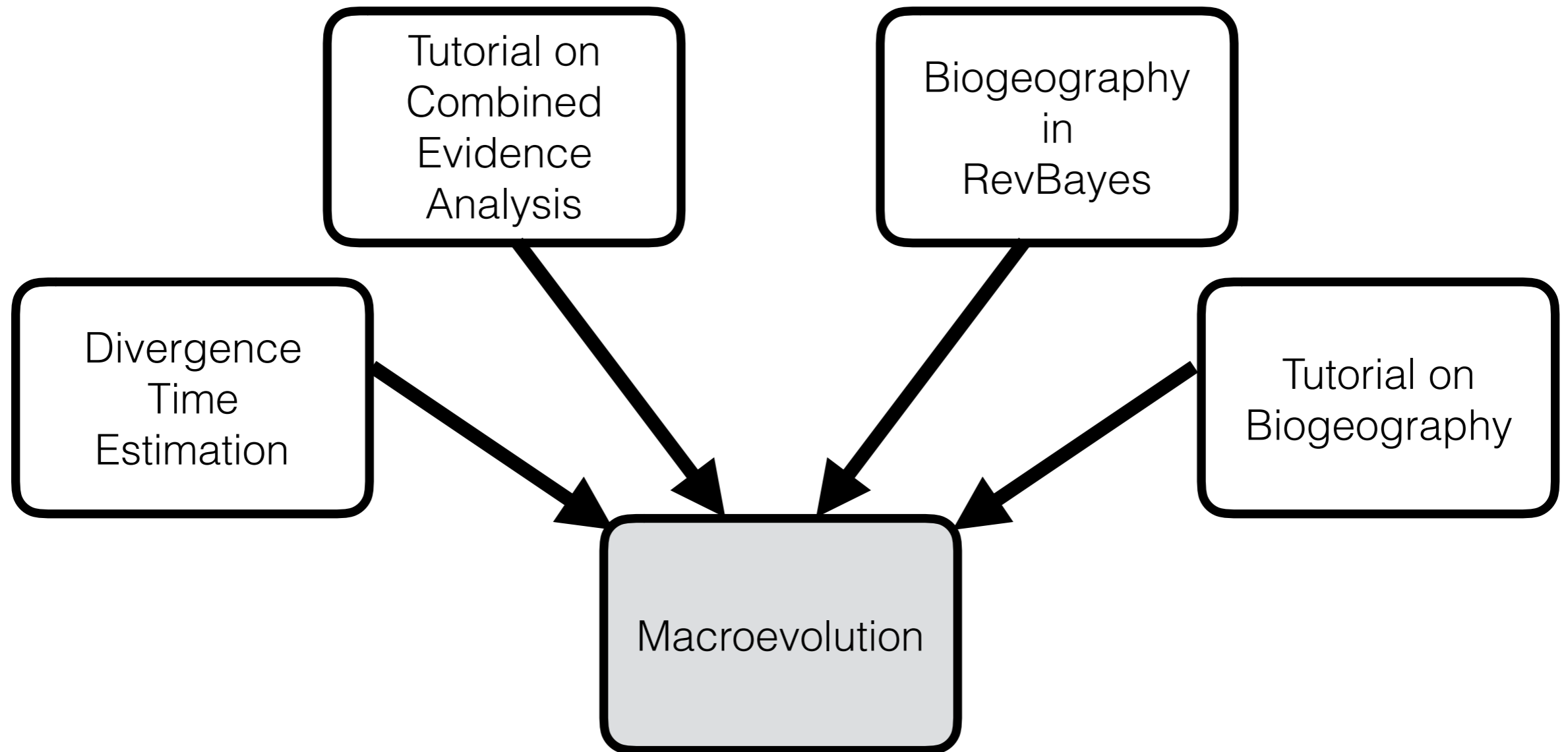
RevBayes Model Modularity



Workshop Day 1



Workshop Day 2



Workshop Day 3

