

# **Formálny biochemický priestor pre špecifikáciu a analýzu biochemických procesov**

Obhajoba diplomovej práce

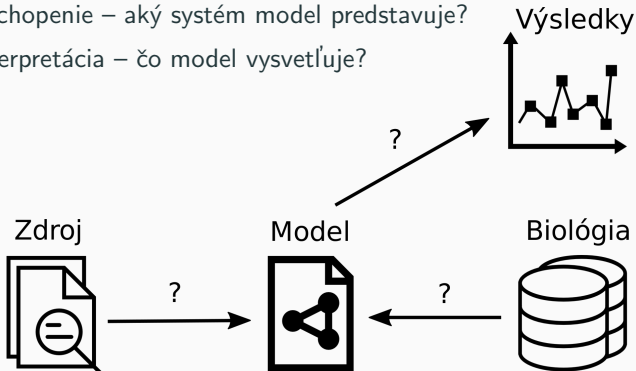
---

Matej Troják

Fakulta Informatiky, Brno, 5.2.2018

Časté problémy s (biologickými) modelmi:

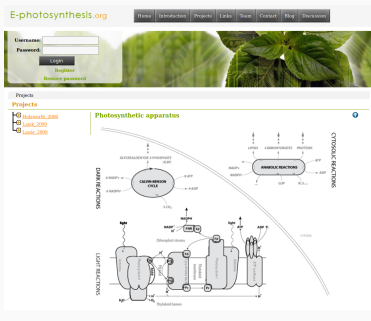
- rekonštrukcia – “spustiteľný” model
- pochopenie – aký systém model predstavuje?
- interpretácia – čo model vysvetľuje?



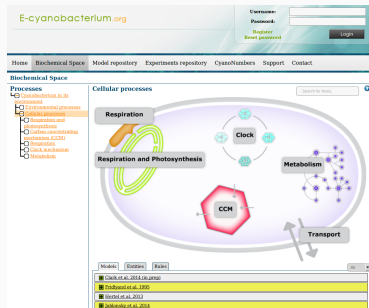
# Riešenie – Comprehensive Modelling Platform

Webový framework pre **integráciu** biologických znalostí s výpočtovými modelmi a experimentami.

- e-photosynthesis.org



- e-cyanobacterium.org



**Biochemický priestor** (Biochemical Space – BCS) je doména biologických znalostí s formálnymi prvkami, ktorá poskytuje

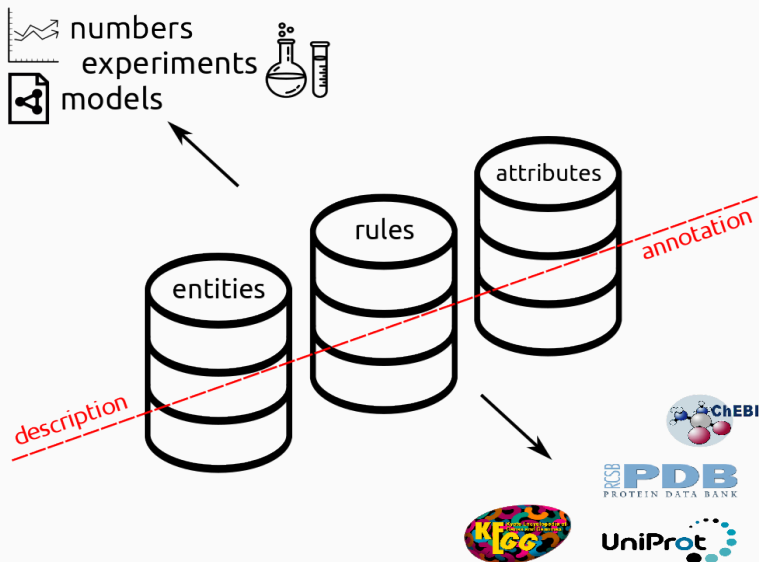
- popis,
- anotáciu,
- zdieľanie

špecificky zameraných biologických modelov.

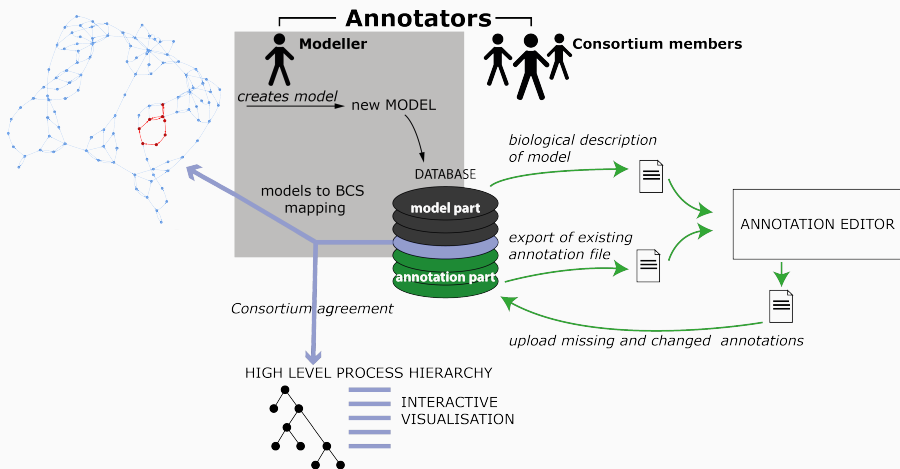
- formálna definícia jazyka Biochemického priestoru,
    - kompaktný zápis chemických reakcií tzv. *rule-based* prístupom,
    - abstrakcia od detailov chemických väzieb,
    - hierarchické usporiadanie individuálnych látok,
    - spustiteľná sémantika,
  - návrh analyzačných techník pre tento jazyk.
- 

## Nad rámec zadania

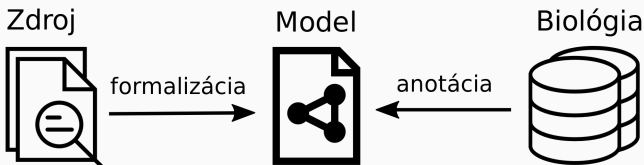
- prototypová implementácia na editovanie a analyzovanie modelov zapísaných v tomto jazyku.



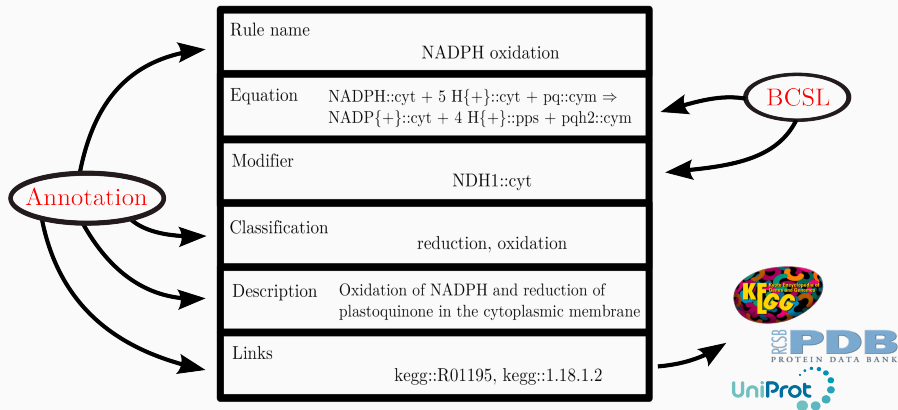
# Princíp platformy



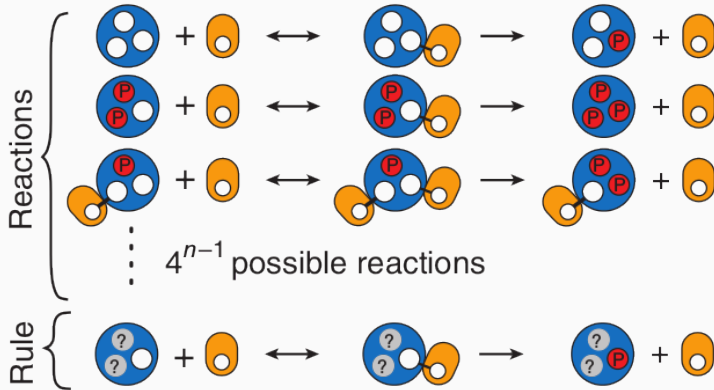
- model získa svoj biologický význam
  - anotácia pre individuálne komponenty/reakcie vždy ľahko dostupná
  - implementovaný model dostupný online
- vymedzením vzťahu model ku BCS získame popis modelu v BCSL
  - možnosť analýzy
  - jednotná forma modelov – porovnávanie

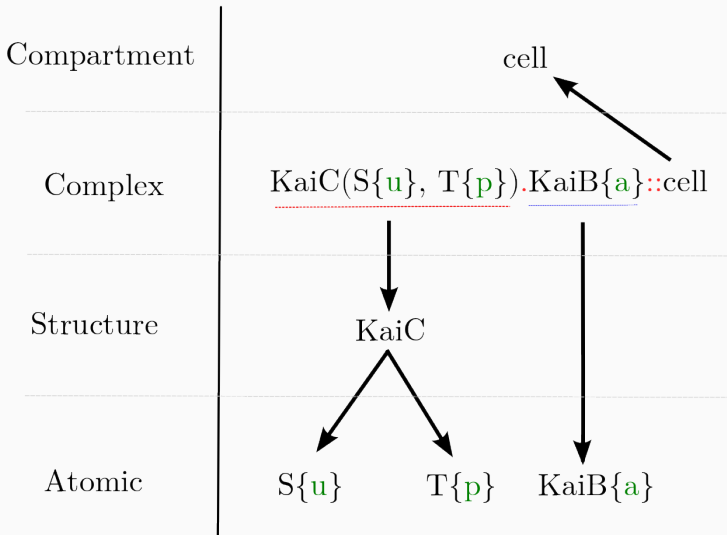


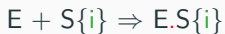




# Rule-based vs. reaction-based







# BCS doména umožňuje syntaktické rozšírenia

Entity ID	KaiC
Entity name	KaiC protein
Composition	S, T
Type	structure
...	

Entity ID	KaiC6
Entity name	KaiC complex
Composition	KaiC.KaiC.KaiC.KaiC.KaiC.KaiC
Type	complex
...	

$S\{u\}::KaiC::KaiC6::cyt \Rightarrow S\{p\}::KaiC::KaiC6::cyt$



$KaiC(S\{u\}).KaiC. \dots .KaiC::cyt \Rightarrow KaiC(S\{p\}).KaiC. \dots .KaiC::cyt$

- Odstránenie redundantných pravidiel
  - zvýšenie jednoduchosti a prehľadnosti modelu
- Redukcia kontextu
  - odstránenie stavov z pravidiel – pravidlá vytvárania a zániku komplexov
- Statická analýza **nedosiahnuteľnosti**
  - overenie **nedosiahnuteľnosti** na základe vlastností definovanej na pravidlách

Prototypová implementácia na editovanie a analyzovanie modelov zapísaných v jazyku Biochemického priestoru.

Funkcie:

- interaktívny editor modelov,
- generovanie prechodového systému,
- generovanie odvodených reakcií z daných pravidiel,
- overovanie dosiahnuteľnosti skupiny zvolených látok,
- vizualizácia prechodových systémov (vrátane dosiahnuteľnosti),
- simulovanie modelov s definovanou kinetikou,
- ukladanie a načítanie modelov/prechodových systémov/vizualizácií.

- zadanie práce splnené
- COMBINE, SBML
  - štandardizácia
- čiastočné výsledky publikované
  - zborník SASB 2015 a CMSB 2016
- nad rámec prototypová implementácia BCSgen
  - krátke (2:53) demo



Ďakujem za pozornosť.

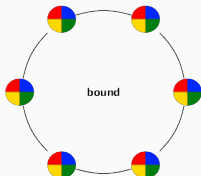
Otázky?

# Otázky oponenta

- Vo druhej kapitole sa spomína, že jazyk Kappa nedokáže vyjadriť obecný vzťah koexistencie bez udania štruktúry, je preto nutné zvoliť nejakú kruhovú/lineárnu štruktúru. Nedala by sa ale takáto vlastnosť vyjadriť úplným grafom?

## Biology

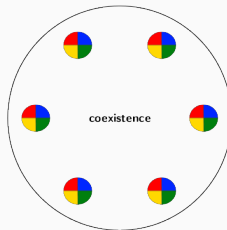
- graph "isomorphism"
- 700 different derivations



- Unphosphorylated protein
- Serine residue phosphorylated protein
- Threonine residue phosphorylated protein
- Both residues phosphorylated protein

## BCS abstraction

- mixture  $\rightarrow$  order not important
- 84 different derivations

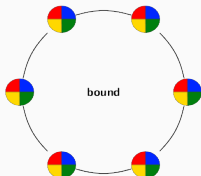


# Otázky oponenta

- Vo druhej kapitole sa spomína, že jazyk Kappa nedokáže vyjadriť obecný vzťah koexistencie bez udania štruktúry, je preto nutné zvoliť nejakú kruhovú/lineárnu štruktúru. Nedala by sa ale takáto vlastnosť vyjadriť úplným grafom?

~~Biology~~ Kappa

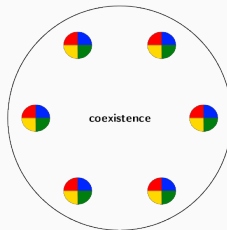
- graph "isomorphism"
- 700 different derivations



- Unphosphorylated protein
- Serine residue phosphorylated protein
- Threonine residue phosphorylated protein
- Both residues phosphorylated protein

BCS abstraction

- mixture  $\rightarrow$  order not important
- 84 different derivations

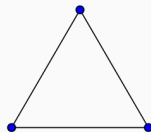


# Otázky oponenta

- Vo druhej kapitole sa spomína, že jazyk Kappa nedokáže vyjadriť obecný vzťah koexistencie bez udania štruktúry, je preto nutné zvoliť nejakú kruhovú/lineárnu štruktúru. Nedala by sa ale takáto vlastnosť vyjadriť úplným grafom?



`Node(bs!1), Node(bs!1)`

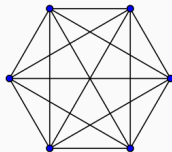


`Node(bs1!1, bs2!2),  
Node(bs1!1, bs2!3),  
Node(bs1!3, bs2!2)`

# Otázky oponenta

- Vo druhej kapitole sa spomína, že jazyk Kappa nedokáže vyjadriť obecný vzťah koexistencie bez udania štruktúry, je preto nutné zvoliť nejakú kruhovú/lineárnu štruktúru. Nedala by sa ale takáto vlastnosť vyjadriť úplným grafom?

...



```
Node(bs1!1, bs2!2, bs3!3, bs4!4, bs5!5),  
Node(bs1!5, bs2!6, bs3!7, bs4!8, bs5!9),  
Node(bs1!9, bs2!4, bs3!10, bs4!11, bs5!12),  
Node(bs1!12, bs2!8, bs3!3, bs4!13, bs5!14),  
Node(bs1!14, bs2!11, bs3!7, bs4!2, bs5!15),  
Node(bs1!15, bs2!13, bs3!10, bs4!6, bs5!1)
```

## Otázky oponenta

- Je nejaký dôvod prečo komplexný agent vo svojej definícií nedefinuje ako multiset, ale ako sekvenciu, ak (pokiaľ dobre vidím) sa všade aj tak táto sekvencia uvažuje len vzhľadom k všetkým jej permutáciám?

možné stavy:

$p1 \sim \{2+, +, *, -\}$

$p2 \sim \{+, -\}$

parciálne kompozície:

$A \sim \{p1, p2\}$

---

$$A(p1\{2+\}).A(p1\{-\})::com \Rightarrow A(p1\{+\}).A(p1\{*\})::com$$

# Otázky oponenta

- Je nejaký dôvod prečo komplexný agent vo svojej definícií nedefinuje ako multiset, ale ako sekvenciu, ak (pokiaľ dobre vidím) sa všade aj tak táto sekvencia uvažuje len vzhľadom k všetkým jej permutáciám?

možné stavy:

$p1 \sim \{2+, +, *, -\}$

$p2 \sim \{+, -\}$

parciálne kompozície:

$A \sim \{p1, p2\}$

---

$A(p1\{2+\}).A(p1\{-})::com \Rightarrow A(p1\{+\}).A(p1\{*\})::com$

$A(p1\{2+\}, p2\{+\}).A(p1\{-}, p2\{-})::com$

# Otázky oponenta

- Je nejaký dôvod prečo komplexný agent vo svojej definícií nedefinuje ako multiset, ale ako sekvenciu, ak (pokiaľ dobre vidím) sa všade aj tak táto sekvencia uvažuje len vzhľadom k všetkým jej permutáciám?

možné stavy:

$p1 \sim \{2+, +, *, -\}$

$p2 \sim \{+, -\}$

parciálne kompozície:

$A \sim \{p1, p2\}$

---

$A(p1\{2+\}).A(p1\{-})::com \Rightarrow A(p1\{+\}).A(p1\{*\})::com$

$A(p1\{+\}, p2\{+\}).A(p1\{*\}, p2\{-})::com$

$A(p1\{2+\}, p2\{+\}).A(p1\{-}, p2\{-})::com$



# Otázky oponenta

- Je nejaký dôvod prečo komplexný agent vo svojej definícii nedefinuje ako multiset, ale ako sekvenciu, ak (pokiaľ dobre vidím) sa všade aj tak táto sekvencia uvažuje len vzhľadom k všetkým jej permutáciám?

možné stavy:

$p1 \sim \{2+, +, *, -\}$

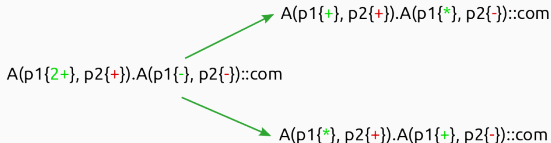
$p2 \sim \{+, -\}$

parciálne kompozície:

$A \sim \{p1, p2\}$

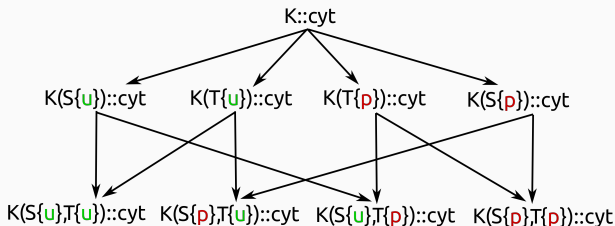
---

$A(p1\{2+\}).A(p1\{-\})::com \Rightarrow A(p1\{+\}).A(p1\{*\})::com$



# Otázky oponenta

- Nebolo by jednoduššie využiť v definícií grounding function kompatibilitu agentov (a zvlášť kompatibilnú podmnožinu agenta) tak ako je definovaná v nasledujúcej kapitole?



- Nebolo by jednoduššie využiť v definícií grounding function kompatibilitu agentov (a zvlášť kompatibilnú podmnožinu agenta) tak ako je definovaná v nasledujúcej kapitole?

Áno, ale ...

... nespĺňa to kritéria *praktickej* definície – bez danej definície grounding function by nebolo ukázané, ako kompatibilnú podmnožinu agenta zostrojiť.

## Rules

```

KaiC::cyt + KaiB::cyt => KaiBC::cyt
KaiBC::cyt => KaiC::cyt + KaiB::cyt
S(u)::KaiC::KaiBC::cyt => S(p)::KaiC::KaiBC::cyt
KaiC::cyt + KaiBC::cyt => KaiBC2::cyt
KaiC::cyt + KaiBC2::cyt => KaiBC3::cyt
KaiC::cyt + KaiBC3::cyt => KaiBC4::cyt
KaiBC4::cyt + KaiA::cyt => KaiABC4::cyt
S(u)::KaiA::KaiABC4::cyt => S(p)::KaiA::KaiABC4::cyt
S(p)::KaiA::KaiABC4::cyt => KaiC3::cyt + S(p)::KaiA::KaiABC::cyt
KaiC3::cyt => 3 KaiC::cyt
S(p)::KaiA::KaiABC::cyt => KaiC::cyt + KaiB::cyt + KaiA(S(p))::cyt

```

## Initial state

```

4 KaiC(S(u))::cyt
1 KaiA(S(u))::cyt
1 KaiB::cyt

```

## Definitions

Name	Definition
KaiABC4	KaiC.KaiC.KaiC.KaiC.KaiB.KaiA
KaiBC	KaiC.KaiB
KaiBC2	KaiC.KaiC.KaiB
KaiBC3	KaiC.KaiC.KaiC.KaiB
KaiBC4	KaiC.KaiC.KaiC.KaiC.KaiB
KaiABC	KaiC.KaiB.KaiA
KaiC3	KaiC.KaiC.KaiC

Transition system file

Cancel

Compute

Show graph

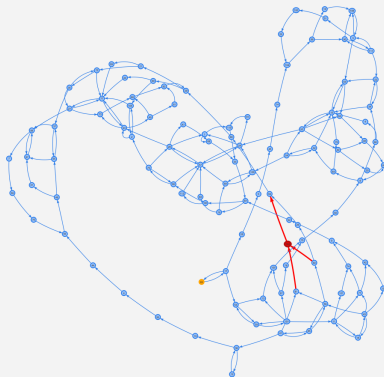
Save reactions to file

Statistics of the model

No. of States:

No. of Edges:

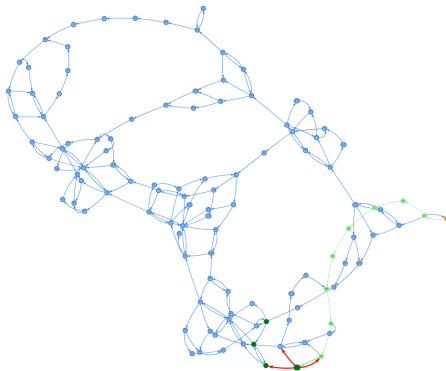
No. of Reactions: 52



1 KalB.KalC(S(p)).KalC(S(p)).KalC(S(u))::cyt

1 KalC(S(p))::cyt

1 KalA(S(u))::cyt



1 KalC(S(u)).KalC(S(u)).KalC(S(u))::cylt  
 1 KalA(S(p))::cylt  
 1 KalB.KalC(S(u))::cylt

File Edit View Help

#### Rules

```
X::rep => @ 'k1'*X::rep'  
Y::rep => @ 'k2'*Y::rep'  
Z::rep => @ 'k1'*Z::rep'  
=> X::rep @ 1/(1+(Z::rep/5)**4)  
=> Y::rep @ 1/(1+(X::rep/5)**4)  
=> Z::rep @ 1/(1+(Y::rep/5)**4)
```

#### Initial state

```
1 X::rep  
1 Y::rep
```

Transition system Model analysis Simulation

#### Simulation settings

☒ Deterministic ☐ Stochastic

Maximum time: 200 (in seconds)

Number of total runs: 1

☐ Apply interpolation

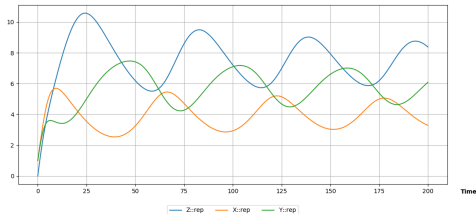
100%

Cancel

Simulate

#### Simulation results

Concentration



Simulate current model

File Edit View Help

#### Rules

```
X::rep => @ 'k1'*X::rep'  
Y::rep => @ 'k2'*Y::rep'  
Z::rep => @ 'k1'*Z::rep'  
=> X::rep @ 1/(1+(Z::rep/5)**4)  
=> Y::rep @ 1/(1+(X::rep/5)**4)  
=> Z::rep @ 1/(1+(Y::rep/5)**4)
```

#### Initial state

```
1 X::rep  
1 Y::rep
```

Transition system Model analysis Simulation

#### Simulation settings

☐ Deterministic ☒ Stochastic  
Maximum time: 200 (in seconds)  
Number of total runs: 1  
☐ Apply interpolation  
100%  
Cancel Simulate

