

Průvodce systémovou dynamikou

- **Co je modelovač systémové dynamiky?** Obecný popis nástroje, včetně jeho principů a mechanismů.
- **Jak modelovač systémové dynamiky funguje?** Popisuje rozhraní a způsob použití.
- **Tutorial: Vlci a ovce (souhrnný).** Návod, jak vytvořit souhrnný model pomocí modelovače systémové dynamiky.

Co je modelovač systémové dynamiky NetLogo?

Systémová dynamika je typ modelování, v kterém se člověk snaží pochopit, jak spolu jednotlivé věci souvisejí. Tento způsob se trochu liší od přístupu založeného na agentech, který v modelech NetLoga běžně používáme.

U agentového přístupu naprogramujete chování jednotlivých agentů a sledujete, co vzniká z jejich interakce. Např. v modelu Vlci a ovce zadáte pravidla, jak na sebe působí vlci, ovce a tráva. Při modelování sledujete chování na úrovni souhrnu, agregátu – např. jak se populace vlků a ovcí mění v průběhu času.

U modelovače systémové dynamiky (System Dynamics Modeler) neprogramujete chování jednotlivých agentů, nýbrž chování populací agentů jako celek. Např. u modelu Vlci a ovce definujete, jak se změní celkový počet ovcí, když stoupne či klesne celkový počet vlků a obráceně. Potom model spustíte a vidíte, jak se obě populace postupně mění.

Modelovač systémové dynamiky rovněž umožňuje nakreslit diagram, jenž tyto populace, neboli „souhrny“ (stocks) definuje a ukazuje, jak se vzájemně ovlivňují. Modelovač diagram načte, vytvoří odpovídající kód NetLoga – globální proměnné, procedury a reportéry – a spustí model systémové dynamiky uvnitř NetLoga.

Základní principy

Diagram systémové dynamiky se skládá ze čtyř prvků: souhrny (stocks), proměnné (variables), toky (flows) a spoje (links).

Souhrn je soubor věcí, agregát. Může to být např. populace ovcí, voda v jezeře či počet prvků v továrně.

Tok přináší a odnáší věci do/ze souhrnu, je to jakési potrubí s kohoutkem kontrolujícím, kolik „vody“ jím proteče.

Proměnná je hodnota použitá v diagramu. Může být pomocí rovnice závisící na ostatních proměnných, nebo konstanta.

Spoj umožňuje, aby hodnota v jedné části diagramu byla dostupná pro ostatní části. Spoj přenáší číslo z proměnné či souhrnu do souhrnu nebo toku.

Modelovač systémové dynamiky počítá a stále znovu odhaduje, jak se bude hodnota souhrnů měnit v průběhu času. Odhady nejsou vždy stoprocentní, ale jejich přesnost můžete ovlivnit změnou

hodnoty dt . Když se dt snižuje, odhaduje se modelování častěji, takže se zvyšuje i přesnost. Na druhou stranu to ale způsobuje, že model běží pomaleji.

Ukázkové modely

V sekci ukázkové modely v knihovně modelů NetLoga se nacházejí čtyři modely používající modelovač systémové dynamiky. Všechny čtyři modely zkoumají růst populace (resp. v modelech s predátory úbytek populace).

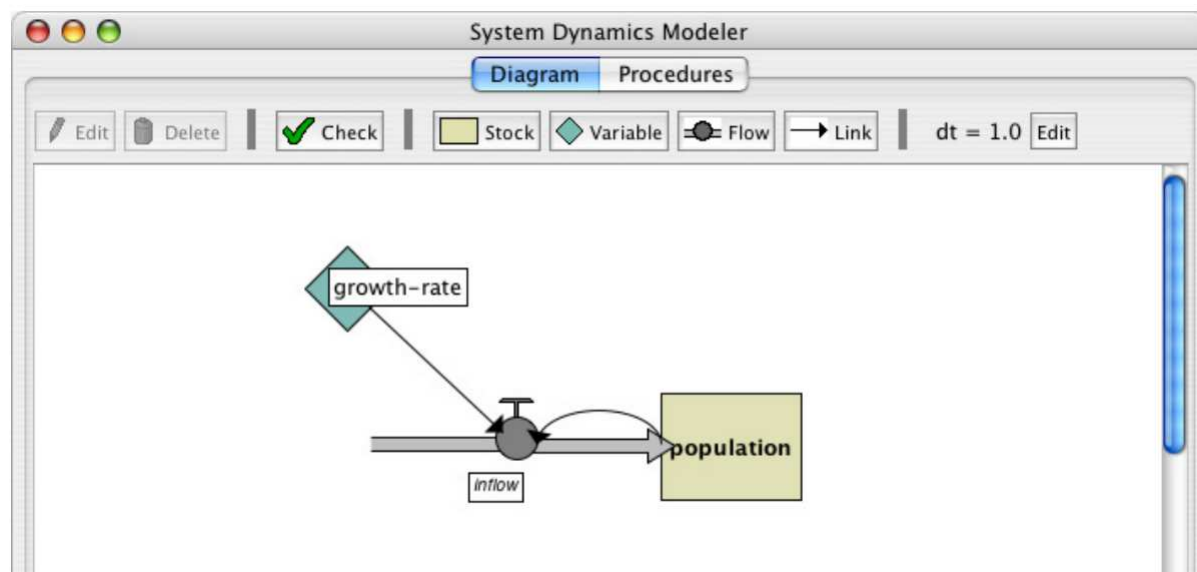
Exponenciální růst a logistický růst jsou jednoduchými příklady růstu v rámci jednoho souhrnu.

Model Vlci a ovce (souhrnný) je příkladem systému s více souhrny, jež se vzájemně ovlivňují. Pomocí modelovače systémové dynamiky modeluje ekosystém s lovcem a kořistí.

Model Vlci a ovce (kombinovaný) je příkladem modelu, v němž vedle sebe probíhá modelování systémové dynamiky i modelování na základě agentů. Nachází se ve složce **Biology** v ukázkových modelech.

Jak modelovač systémové dynamiky funguje?

Chcete-li spustit modelovač systémové dynamiky, vyberte položku **System Dynamics Modeler** v menu **Tools**. Objeví se následující okno.



Panel Diagram

Do panelu **Diagram** kreslíte diagram systémové dynamiky.

Nástrojová lišta obsahuje tlačítka, s jejichž pomocí upravíte, smažete či vytvoříte položky v diagramu.

Vytvoření prvků diagramu



Diagram systémové dynamiky je tvořen čtyřmi druhy prvků: souhrny, proměnnými, toky a spoji.

Souhrn. Souhrn vytvoříte tak, že stisknete tlačítko **Stock** na nástrojové liště a kliknete do oblasti diagramu pod ní. Objeví se nový souhrn. Každý souhrn potřebuje unikátní název, jenž bude globální proměnnou. Souhrny rovněž vyžadují počáteční hodnotu (initial value), což může být číslo, proměnná, komplexní výraz NetLoga či volání reportéru NetLoga.

Proměnná. Proměnnou vytvoříte stisknutím tlačítka **Variable** a kliknutím do oblasti diagramu. Každá proměnná v modelu systémové dynamiky potřebuje unikátní název, jenž bude názvem procedury či globální proměnné. Proměnné rovněž vyžadují výraz (expression), což může být číslo, proměnná, komplexní výraz NetLoga či volání reportéru NetLoga.

Tok. Tok vytvoříte tak, že stisknete tlačítko **Flow**. Klikněte do místa, kde chcete, aby tok začínal – buď na souhrnu, nebo v prázdné oblasti – a přesuňte kurzor do místa, kde má tok končit – opět na souhrnu, nebo v prázdné oblasti. Každý tok potřebuje unikátní název, jenž bude reportérem NetLoga. Toky vyžadují výraz (expression), který určuje míru toku ze vstupu do výstupu. Tento výraz může být číslo, proměnná, komplexní výraz NetLoga či volání reportéru NetLoga. Je-li hodnota záporná, probíhá tok opačným směrem.

Je-li k souhrnu připojeno více toků, je důležité zvážit, jak se budou vzájemně ovlivňovat. NetLogo nevyžaduje, aby toky vedoucí ze souhrnu byly uspořádány v určitém pořadí. NetLogo rovněž nezaručuje, že se suma toků vedoucích ze souhrnu rovná hodnotě souhrnu nebo je menší. Tato chování lze explicitně implementovat, když vytváříte výraz toku.

Definujeme-li např. tok jako konstantní hodnotu 10, zajistíme, že nikdy nebude čerpat víc, než je hodnota souhrnu, pomocí primitiva `min`: `min (list stock 10)`. Jestliže chceme, aby tok A vyčerpal souhrn dřív, než je vypočítán tok B, můžeme toky propojit a změnit tok B, aby odečetl hodnotu toku A ze souhrnu: `min (list (max (list 0 (stock - flow-a))) 10)`.

Spoj. Spoj vytvoříte tak, že kliknete do místa, v kterém má začínat – proměnná, souhrn či tok – a přesunete kurzor do cílového místa, proměnné nebo toku.

Práce s prvky diagramu

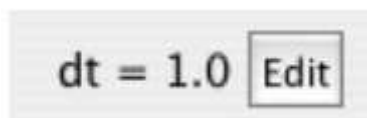
Když vytváříte souhrn, proměnnou či tok, objeví se na daném prvku červený otazník, který značí, že prvek ještě nemá jméno. Červená barva naznačuje, že souhrn ještě není celý – k vytvoření modelu systémové dynamiky mu chybí ještě minimálně jedna hodnota. V momentě, kdy je prvek kompletní, změní se písmo popisky v černé.

Výběr. Prvek diagramu vyberete tak, že na něj kliknete. Pro výběr více prvků držte klávesu Shift. Prvky lze rovněž vybrat přetáhnutím rámečku s výběrem.

Úprava. Chcete-li upravit prvek diagramu, vyberte prvek a stiskněte tlačítko **Edit** na nástrojové liště nebo klikněte dvakrát na vybraný prvek. (Upravovat můžete souhrny, toky a proměnné, nikoliv spoje.)

Přesun. Jestliže chcete prvek diagramu přesunout, vyberte ho a přetáhněte myší na místo, kam ho chcete umístit.

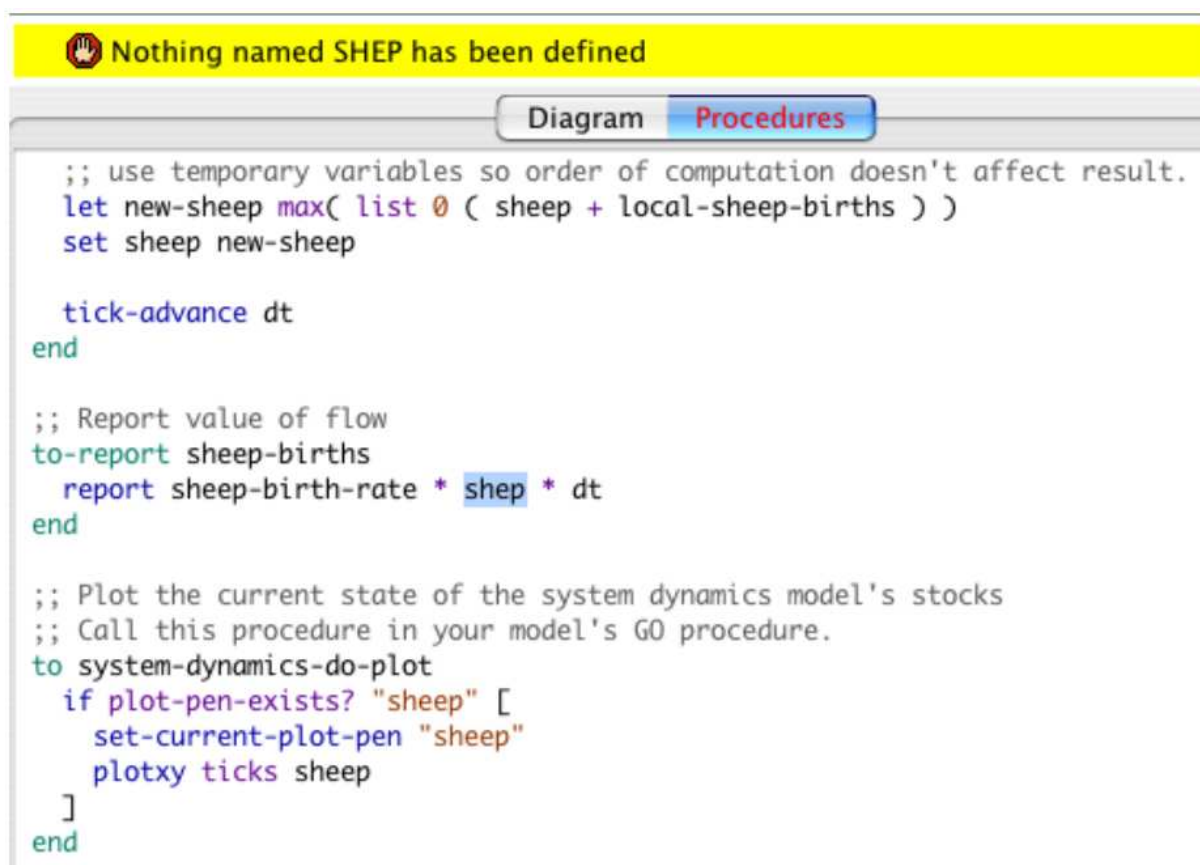
Úprava dt



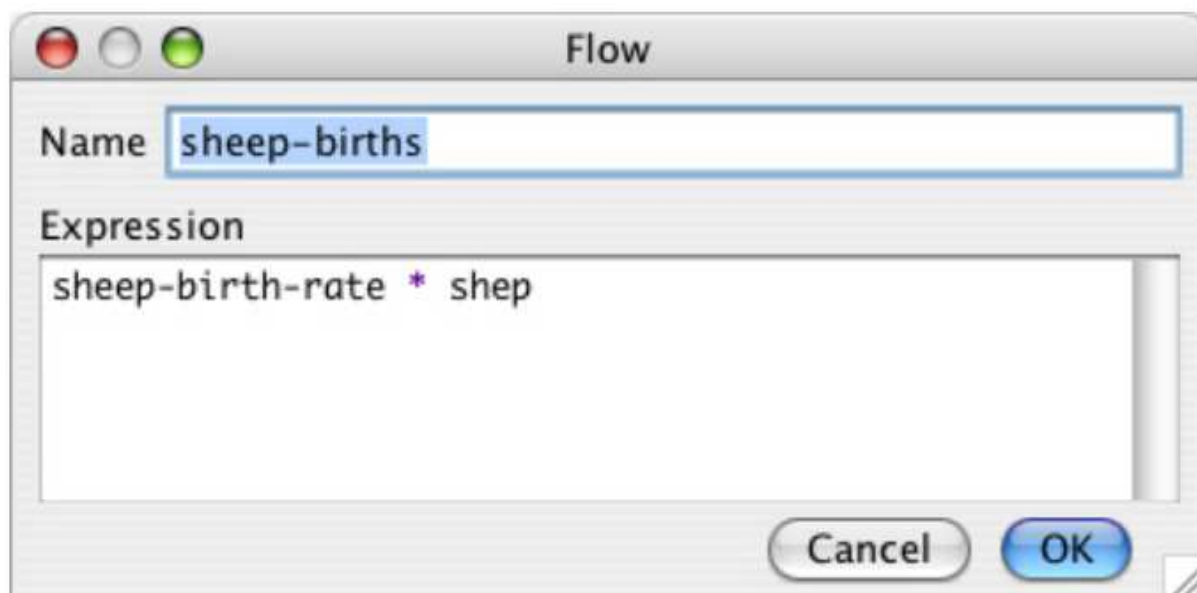
Na pravé straně nástrojové lišty je výchozí dt, tj. interval, jenž se používá k odhadování výsledků modelu systémové dynamiky. Chcete-li změnit výchozí hodnotu dt, stiskněte tlačítko **Edit**, nacházející vedle hodnoty dt, a zadejte novou hodnotu.

Chyby

Když kliknete na tlačítko **Check** nebo když upravujete souhrn, tok či proměnnou, vygeneruje modelovač automaticky kód NetLoga odpovídající diagramu a zkompile ho. Jestliže bude obsahovat chybu, zbarví se panel **Procedures** červeně, objeví se zpráva a zvýrazní se část vygenerovaného kódu, jež problém způsobila.



Tímto způsobem snáze zjistíte, který prvek diagramu způsobuje potíže.



Panel Procedures

Modelovač systémové dynamiky vygeneruje na základě obsahu diagramu proměnné a procedury NetLoga. Tyto procedury ve skutečnosti provádějí výpočty. Procedury NetLoga vygenerované z diagramu se zobrazí v okně modelovače systémové dynamiky v panelu **Procedures**.

Obsah panelu **Procedures** nelze měnit; chcete-li změnit kód, musíte upravit diagram.

Podívejme se blíže, jaký je vztah vygenerovaného kódu a diagramu.

- Souhrny odpovídají globální proměnné, která je spuštěna jako hodnota či výraz, jež jste zadali do pole **Initial value**. Všechny souhrny se aktualizují při každém kroku založeném na tocích dovnitř a ven.
- Toky odpovídají proceduře obsahující výraz, jenž byl zadán do pole **Expression**.
- Proměnnými mohou být buď globální proměnné, nebo procedury. Obsahuje-li pole **Expression** konstantu, jedná se o globální proměnnou a bude spuštěna v této hodnotě. Použijete-li k definování proměnné komplexní výraz, vznikne procedura jako tok.

Proměnné a procedury definované v tomto panelu jsou přístupné v hlavním okně NetLoga, stejně jako proměnné a procedury, jež definujete sami v hlavním panelu NetLoga **Procedures**. Procedury můžete volat z hlavního panelu **Procedures**, z příkazového panelu nebo tlačítka v panelu **Interface**. Na globální proměnné můžete odkazovat odkudkoliv, včetně hlavního panelu **Procedures** a ukazatelů.

Za pozornost stojí tři hlavní procedury: `system-dynamics-setup`, `system-dynamics-go` a `system-dynamics-do-plot`.

`system-dynamics-setup` spustí souhrnný model. Nastaví hodnotu `dt`, zavolá `reset-ticks` a spustí souhrny a konvertory. První se inicializují konvertory s konstantní hodnotou, pak následují souhrny s konstantní hodnotou. Ostatní souhrny jsou spuštěny v abecedním pořadí.

`system-dynamics-go` spustí souhrnný model po počet `dt` časových jednotek. Počítá hodnoty toků a proměnných a aktualizuje hodnotu souhrnů. Rovněž volá `tick-advance` s hodnotou `dt`. Konvertory a toky s nekonstantními výrazy budou spočítány pouze jednou, když je zavolána tato procedura, nicméně pořadí jejich hodnocení není definováno.

`system-dynamics-do-plot` vykreslí hodnoty souhrnů v souhrnném modelu. Nejdříve vytvořte graf v hlavním okně NetLogo. Potom musíte pro každý souhrn, který chcete vykreslit, určit pero grafu. Tato procedura použije aktuální graf, který můžete změnit pomocí příkazu `set-current-plot`.

Modelovač systémové dynamiky a NetLogo

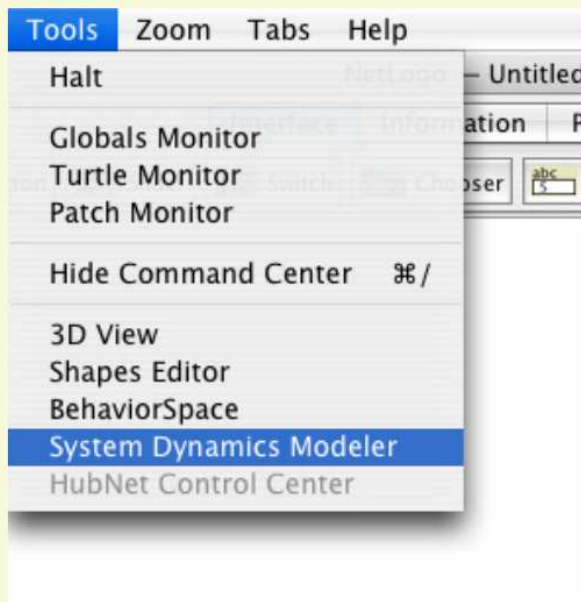
Diagram, který vytvoříte v modelovači systémové dynamiky, a procedury, jež jsou z diagramu vygenerovány, jsou součástí modelu NetLogo. Diagram se uloží při ukládání modelu NetLogo (do stejného souboru).

Tutorial: Vlci a ovce

Zkusme si v modelovači systémové dynamiky vytvořit model Vlci a ovce.

Krok jedna: Reprodukce ovcí

- Open a new model in NetLogo.
- Launch the System Dynamics Modeler in the Tools menu.



- Otevřete v NetLogu nový model.
- Spusťte modelovač systémové dynamiky z menu **Tools**.

Náš model bude mít populaci vlků a populaci ovcí. Začneme ovce. Nejdříve vytvoříme souhrn obsahující populaci ovcí.

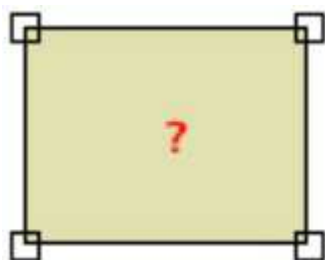
- Press the Stock button in the toolbar.



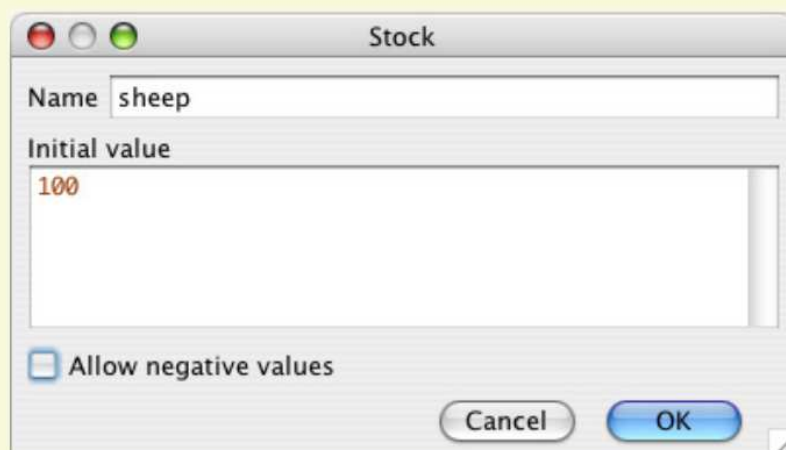
- Click in the diagram area.

- Stiskněte tlačítko **Stock** na nástrojové liště.
- Klikněte do oblasti diagramu.

Objeví se souhrn s červeným otazníkem uprostřed.



- Double-click the Stock to edit.
- Name the stock `sheep`
- Set the initial value to `100`.
- Deselect the Allow Negative Values checkbox. It doesn't make sense to have negative sheep!



- Souhrn upravíte tak, že na něj dvakrát kliknete.
- Nazvěte souhrn `sheep`.
- Nastavte počáteční hodnotu na `100`.

- Odškrtněte políčko **Allow Negative Values** (Povolit záporné hodnoty). Záporné ovce nedávají smysl!

Naše populace ovcí může vzrůst, když se narodí nové ovce. Do diagramu je přidáme tak, že vytvoříme tok do souhrnu `sheep`.

- Click on the **Flow** button in the toolbar and press the mouse button in an empty area to the left of the `sheep` Stock. Drag the Flow to the right until it connects to the `sheep` Stock and let go.
- Edit the Flow and name it `sheep-births`.

- Klikněte na tlačítko **Flow** na nástrojové liště a stiskněte tlačítko myši v prázdné oblasti nalevo od souhrnu `sheep`. Přetáhněte tok doprava, dokud se nespojí se souhrnem `sheep` a pak jej pusťte.
- Upravte tok a pojmenujte ho `sheep-births`.

Počet ovcí narozených během dané doby závisí na počtu živých ovcí: více ovcí znamená vyšší reprodukci.

- Draw a Link from the `sheep` Stock to the `sheep-births` Flow.
- For now, enter a constant, such as `1`, into the Expression field.

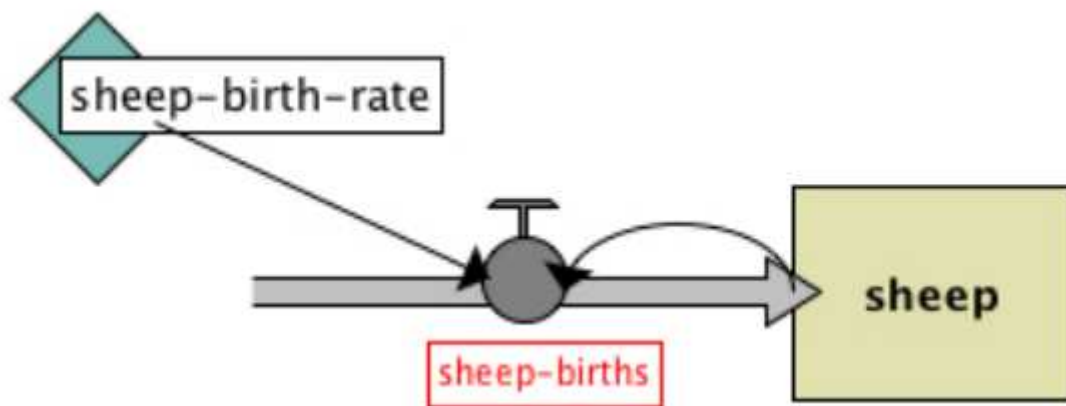
- Nakreslete spoj od souhrnu `sheep` k toku `sheep-births`.
- Do pole **Expressions** vložte pro teď konstantu, např. `1`.

Porodnost ovcí závisí rovněž na určitých faktorech konstanty, které jsou mimo rozsah tohoto modelu – míra reprodukce atd.

- Create a Variable and name it `sheep-birth-rate`. Set its value to `0.04`
- Draw a Link from the `sheep-birth-rate` Variable to the `sheep-births`.

- Vytvořte proměnnou a pojmenujte ji `sheep-birth-rate`. Nastavte její hodnotu na `0.04`.
- Nakreslete spoj od proměnné `sheep-birth-rate` k `sheep-births`.

Diagram by měl vypadat takto:



Tok `sheep-births` má červenou popisku, protože jsme ještě nezadali výraz. Červená naznačuje, že v této části diagramu něco chybí.

Počet ovcí tekoucích do souhrnu je přímo úměrný počtu ovcí a jejich porodnosti.

- Edit the `sheep-births` Flow and set the expression to `sheep-birth-rate * sheep`.

- Upravte tok `sheep-births` a nastavte výraz na `sheep-birth-rate * sheep`.

Ted' máme diagram dokončený. Jestliže se chcete podívat na kód NetLoga vygenerovaný diagramem, klikněte na panel **Procedures** v okně modelovače systémové dynamiky. Vypadá takto:

```

Diagram Procedures
;; System dynamics model globals
globals [
  ;; constants
  sheep-birth-rate
  ;; stock values
  sheep
  ;; size of each step, see SYSTEM-DYNAMICS-GO
  dt
]

;; Initializes the system dynamics model.
;; Call this in your model's SETUP procedure.
to system-dynamics-setup
  reset-ticks
  set dt 0.1
  ;; initialize constant values
  set sheep-birth-rate .04
  ;; initialize stock values
  set sheep 100
end

;; Step through the system dynamics model by performing next iteration of Euler's method.
;; Call this in your model's GO procedure.
to system-dynamics-go
  ;; compute variable and flow values once per step
  let local-sheep-births sheep-births

```

Krok 2: Integrace NetLoga

Když už máte v modelovači systémové dynamiky vytvořen souhrnný model, můžete s ním komunikovat z hlavního okna rozhraní NetLoga. Vytvořme si model NetLoga a spustíme kód vytvořený diagramem. Potřebujeme tlačítka PŘIPRAV a START, která zavolají procedury `system-dynamics-setup` a `system-dynamics-go` vytvořené modelovačem systémové dynamiky. Rovněž budeme chtít ukazatel a graf, abychom mohli sledovat změny v populaci ovcí.

- Select the main NetLogo window
- In the Procedures tab, write:

```
to setup
  ca
  system-dynamics-setup
end

to go
  system-dynamics-go
  system-dynamics-do-plot
end
```

- Move to the Interface tab
- Create a `setup` button
- Create a `go` button (don't forget to make it forever)

- Create a `sheep` monitor.
- Create a plot called "populations" with a pen named "sheep".

- Vyberte hlavní okno NetLoga.
- Do panelu **Procedures** napište kód z rámečku výše.
- Přesuňte se do panelu **Interface**.
- Vytvořte tlačítko PŘIPRAV.
- Vytvořte tlačítko START (nezapomeňte, že má být trvalé).
- Vytvořte ukazatel ovce .
- Vytvořte graf nazvaný populace a pero ovce.

Ted' můžeme model spustit.

- Press the `setup` button.
- Don't press the "go" button yet. Instead, type `go` four or five times into the Command Center

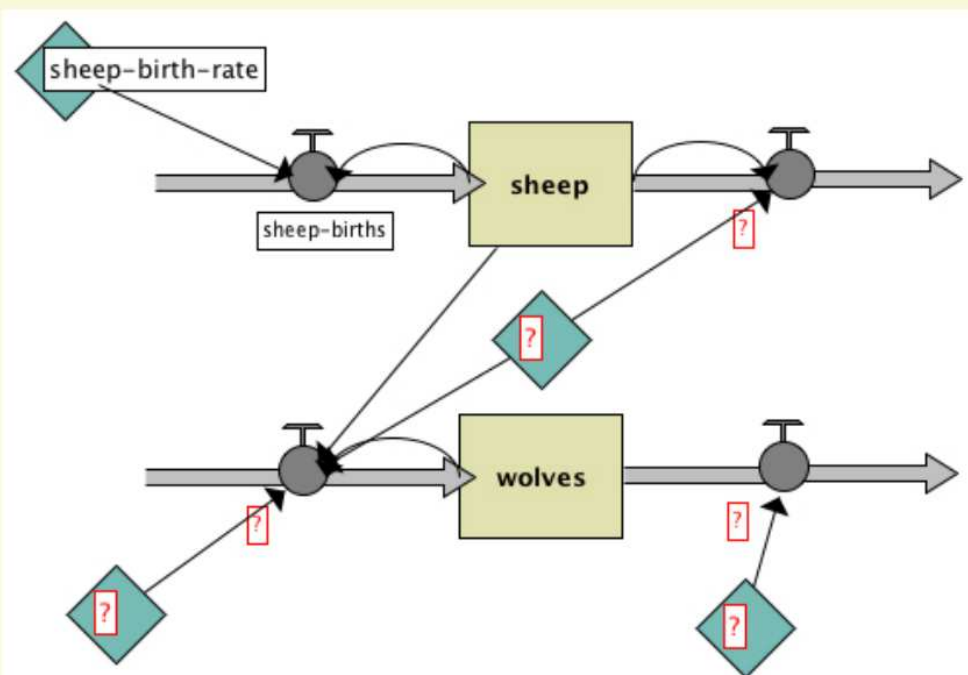
- Stiskněte tlačítko PŘIPRAV.
- Ještě nepoužívejte tlačítko START. Místo toho napište do příkazového panelu čtyřikrát nebo pětkrát `go`.

Všimněte si, co se stane. Populace ovcí poroste exponenciálně. Po čtyřech pěti iteracích už jich máme obrovský počet. Děje se tak proto, že se sice ovce reprodukují, ale nikdy neumírají.

Tento nedostatek napravíme tak, že do diagramu přidáme populaci vlků, kteří budou ovce požírat.

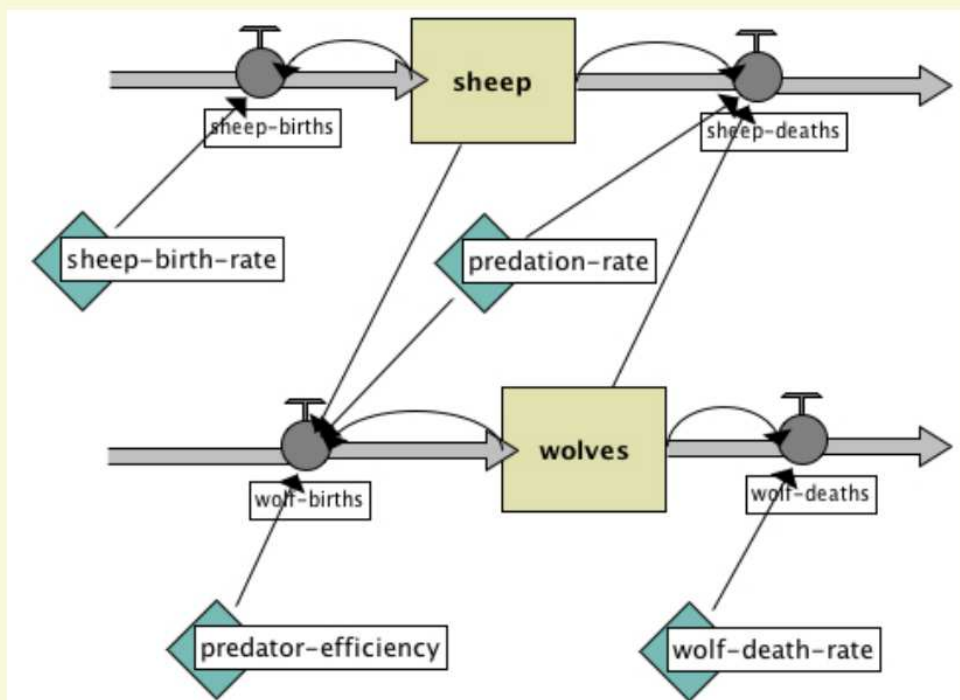
Krok 3: Vlci

- Move back to the System Dynamics window
- Add a stock of wolves
- Add Flows, Variables and Links to make your diagram look like this:



- Add one more Flow from the `wolves` Stock to the Flow that goes out of the Sheep stock.
- Fill in the names of the diagram elements so it looks like this:

- Vraťte se do okna systémové dynamiky.
- Přidejte souhrn `wolves`.
- Přidejte toky, proměnné a spoje, aby diagram vypadal jako na obrázku.
- Přidejte další tok od souhrnu `wolves` k toku, který vychází ze souhrnu `sheep`.
- Doplňte názvy prvků diagramu tak, aby vypadal takto:



where
 initial-value of wolves is 30,
 wolf-deaths is $\text{wolves} * \text{wolf-death-rate}$,
 wolf-death-rate is 0.15,
 predator-efficiency is .8,
 wolf-births is $\text{wolves} * \text{predator-efficiency} * \text{predation-rate} * \text{sheep}$,
 predation-rate is $3.0\text{E-}4$,
 and sheep-deaths is $\text{sheep} * \text{predation-rate} * \text{wolves}$.

kde

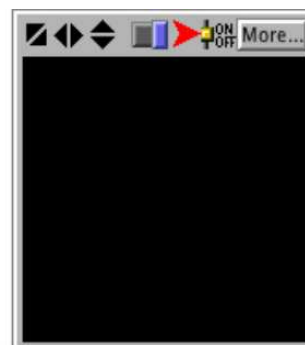
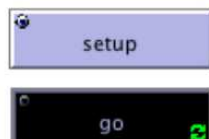
počáteční hodnota souhrnu wolves je nastavena na 30,
 wolf-deaths je $\text{wolves} * \text{wolf-death-rate}$,
 wolf-death-rate je 0.15,
 predator-efficiency je .8,
 wolf-births je $\text{wolves} * \text{predator-efficiency} * \text{predation-rate} * \text{sheep}$,
 predation-rate je $3.0\text{E-}4$,
 a sheep-deaths je $\text{sheep} * \text{predation-rate} * \text{wolves}$.

A teď máme diagram hotový.

- Go back to the main NetLogo window
- Add a plot pen named "wolves" to the population plot
- Press `setup` and `go` to see your System Dynamics Modeler diagram in action.

- Vraťte se do hlavního okna NetLogo.
- Do grafu `populace` přidejte pero grafu s názvem `vloci`.
- Stiskněte tlačítka PŘIPRAV a START a sledujte, jak diagram modelovače systémové dynamiky pracuje.

Vytvoří se graf populací, který bude vypadat takto:



Copyright 1999-2009 by Uri Wilensky.
Všechna práva vyhrazena.

Aplikace NetLogo, modely i dokumentace jsou šířeny veřejnosti zdarma pro účel tvorby a studia modelů. Software, modely a dokumentaci je možné pro studijní a výzkumné účely používat a měnit, a to za podmínky, že je výsledný produkt nabízen bezplatně a s uvedením informace o autorských právech a jménem původce na všech kopiích a související dokumentaci.

Pro jiné využití - než jsou výše zmíněné nekomerční způsoby - celku i jednotlivých částí (a to jak v původní, nebo změněné podobě) je třeba předem požádat o svolení od Uri Wilensky. Software, modely ani dokumentace nesmějí být užívány, přepisovány, ani upravovány jako součást komerčního softwaru nebo hardwaru bez předchozího získání licence od Uri Wilensky. Nezaručujeme kompatibilitu tohoto systému s jakýmkoliv jiným systémem a neposkytujeme žádné záruky.

Pro účely citování v akademických publikacích použijte tento odkaz:
Wilensky, U. (1999). NetLogo. <http://ccl.northwestern.edu/netlogo>. Center for Connected Learning and Computer-Based Modeling. Northwestern University, Evanston, IL.