



# Počítačová simulace logistických procesů I. - cvičení 02

- › ŠAVŠ, Štoček, Karpeta, Varjan
- › 30.4.2013





# Počítačová simulace logistických procesů I. - cvičení 02

## Přehled témat

- › Zobrazení stavu objektů pomocí LED symbolů
- › Standardní strategie pro pohyb MUs z objektu na objekt
- › Strategie na výstupu z objektu
- › Statistické tabulky
- › Statistika
- › Graf
- › Graf – zdroje dat
- › Editor oken
- › Změna ikony
- › Hlavní okno editoru ikon
- › Editor oken
- › Animace ikon
- › Postup pro vytvoření jednoduché animace ikony
- › Editor ikon - doplňkové informace
- › Atributy
- › Tabulka atributů
- › Datové typy
- › Uživatelem definované atributy





# Počítačová simulace logistických procesů I. - cvičení 02

## Přehled témat

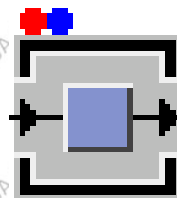
- › Method / metoda
- › Struktura metody
- › Operace SimTalk
- › Konvence
- › Komentáře
- › Příkazy
- › Variable / proměnná
- › Vzorový příklad: PSLP1\_CV02\_M01
- › TableFile / tabulka
- › Dědičnost tabulky
- › Sloupkový a řádkový index tabulky
- › Formát tabulky - dimenze
- › Formát tabulky - datový typ
- › Vnořené tabulky - tabulky v tabulkách
- › Vzorový příklad: PSLP1\_CV02\_M02
- › Vzorový příklad: PSLP1\_CV02\_M03






## Počítačová simulace logistických procesů I. - cvičení 02

### Zobrazení stavu objektů pomocí LED symbolů



- stav objektů materiálového toku lze znázornit pomocí LED-symbolu na horním okraji ikony objektu
- je možné zároveň znázornit více stavů
- stavem objektu se rozumí následující -
 

červená	objekt má poruchu - <b>failed</b>
modrá	objekt má přestávku - <b>paused</b>
zelená	objekt pracuje - <b>working</b>
žlutá	objekt je blokován - <b>blocked</b>
hnědá	objekt se seřizuje - <b>setting-up</b> (setUp)
světle modrá (no_entry)	objekt se „zotavuje“ - <b>recovering</b>
oranžová	objekt čeká na zdroje
- stav operational - připraven - nemá žádný LED symbol
- aby zobrazování pomocí LED bylo aktivní, musí být stisknuté tlačítko  v menu





# Počítačová simulace logistických procesů I. - cvičení 02

## Standardní strategie pro pohyb MUs z objektu na objekt



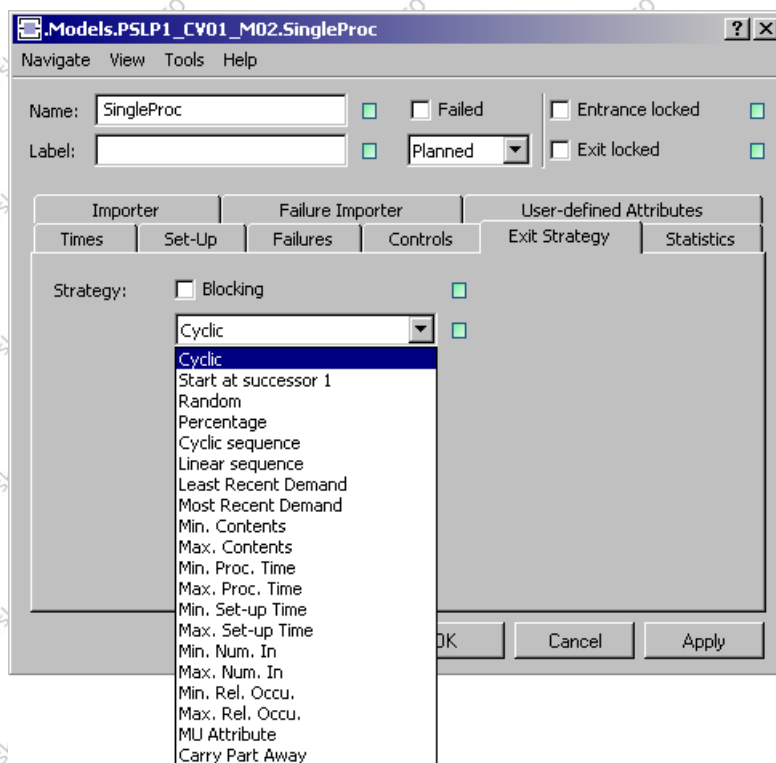
- › Defaultním chování na místech rozvětvení je popořadě/diverging, tzn., že se MUs předávají po řadě na všechny následující objekty.
- › Pro slučování platí pravidlo FIFO - first in first out. Je-li následující objekt obsazený, zapíše MU svůj pokus o přesun u následníka do seznamu blokováných MU zpředu.
- › zobrazení informace o čísle následníka „View/Options/Show Predecessors“ a „View/Options/Show Successors“ předchůdce „View/Options/Show Predecessors“ v příkazové liště dané sítě





# Počítačová simulace logistických procesů I. - cvičení 02

## Strategie na výstupu z objektu



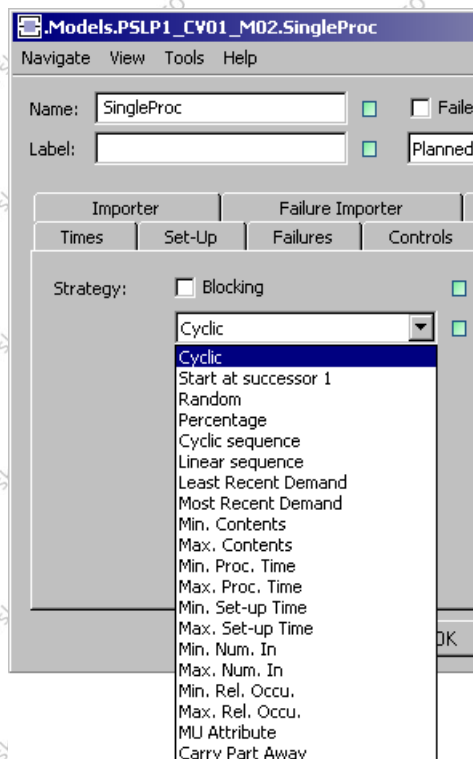
- › Na záložce Exit Strategy u objektů materiálového toku se nastavuje typ předávání MUs na následující spojené objekty.
- › Standardním nastavením je Cyclic/popořadě. Políčko Blocking určuje, zda se počítá s blokování předávání MU dále nebo ne. (Pokud je pole neaktivní, znamená to, že pokud právě přiřazený následník MU přijmout zrovna nemůže, protože je např. obsazený, otestuje se další následník v řadě. Pokud je pole aktivní, čeká se s předáním až do doby, kdy bude tento následník schopen MU přijmout.)





# Počítačová simulace logistických procesů I. - cvičení 02

## Strategie na výstupu z objektu



- › Cyclic - přesun v pořadí spojení objektů
- › Start at successor 1 - pokus o přesun začíná vždy u následníka 1
- › Random - následník se vybírá náhodně
- › Percentage - umožňuje procentuální rozdělení
- › Cyclic sequence - následník se vybere z řady určené tabulkou, další následník při pokusu o předání je určen dalším řádkem v tabulce
- › Linear sequence - následník se začíná vybírat vždy od prvního řádku v tabulce
- › Least/Most Recent Demand - nejstarší/nejmladší požadavek, následníkem je ten objekt, který na MU čekal nejdelší/nejkratší dobu
- › ostatní možnosti berou v úvahu nastavené nebo statistické parametry (např. následníkem je objekt s minimálním/maximálním obsahem, minimálním/maximálním časem opracování atp.)





# Počítačová simulace logistických procesů I. - cvičení 02

## Statistické tabulky

Models.Frame1.Produkce

Name:   Failed  Entrance locked

Label:   Planned  Exit locked

Attributes | Times | Set-Up | Failures | Controls | Exit Strategy

Statistics | Importer | Failure Importer | Custom Attributes

Resource statistics

Resource type:

Working:	78.20%	Setting-up:	0.00%	Contents:	6
Waiting:	0.00%	Empty:	0.00%	Minimum contents:	0
Blocked:	21.80%	Rel. occupation:	99.98%	Maximum contents:	6
Failed:	0.00%			Entries:	5368
Paused:	0.00%			Exits:	5362
Unplanned:	0.00%				

OK Cancel Apply

.MUs.Entity:6463

Navigate View Tools Help

Name:   Number: 06463

Label:   Conveying direction: 0 (forward)

Attributes | Product Statistics | Custom Attributes

Product statistics

Production:	100.00%	(Waiting: 23.76%)	Working: 76.24%
Transport:	0.00%	(Waiting: 0.00%)	Working: 0.00%
Storage:	0.00%	(Waiting: 0.00%)	Working: 0.00%

OK Cancel Apply

- › nejdůležitější statistické údaje lze zjistit na záložce Statistics popř. Product Statistics u objektů materiálového toku popř. u MUs
- › při stisknutí tlačítka Reset v objektu EventController se tyto hodnoty nulují





# Počítačová simulace logistických procesů I. - cvičení 02

## Statistika

Statistics Report

### Tecnomatix Plant Simulation 10 Statistics

Resource Statistics

Created on	15. října 2012 8:53
Model name	C:\Dokumenty\UNI\SkodaAuto\PSLP1\03_Stocek\121002_PSLP1_CV02_V01.spp
Simulation time	1:00.0000

#### Resource Statistics - Resource Statistics

Portions of the States of the Statistics Collection Period

Object	Working	Set-up	Waiting	Blocked	Stopped	Failed	Paused	Unplanned	Empty	Start of Statistics Collection	Portion
Source	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.0000	

#### Material Flow Properties

Object	Number of Entries	Number of Exits	Minimum Contents	Maximum Contents	Relative Occupation without Interruptions	Relative Occupation with Interruptions
Source	4	4	0	1	100.00%	100.00%

#### Working Time

Object	Portion	Count	Sum	Mean Value	Standard Deviation
Source	0.00%	0	0.0000	0.0000	0.0000

- › detailní statistickou zprávu požadovaného objektu lze získat aktivováním daného objektu levým tlačítkem na myši a stisknutím tlačítka F6
- › hodnoty se za běhu simulace aktualizují pouze přes tlačítko Refresh



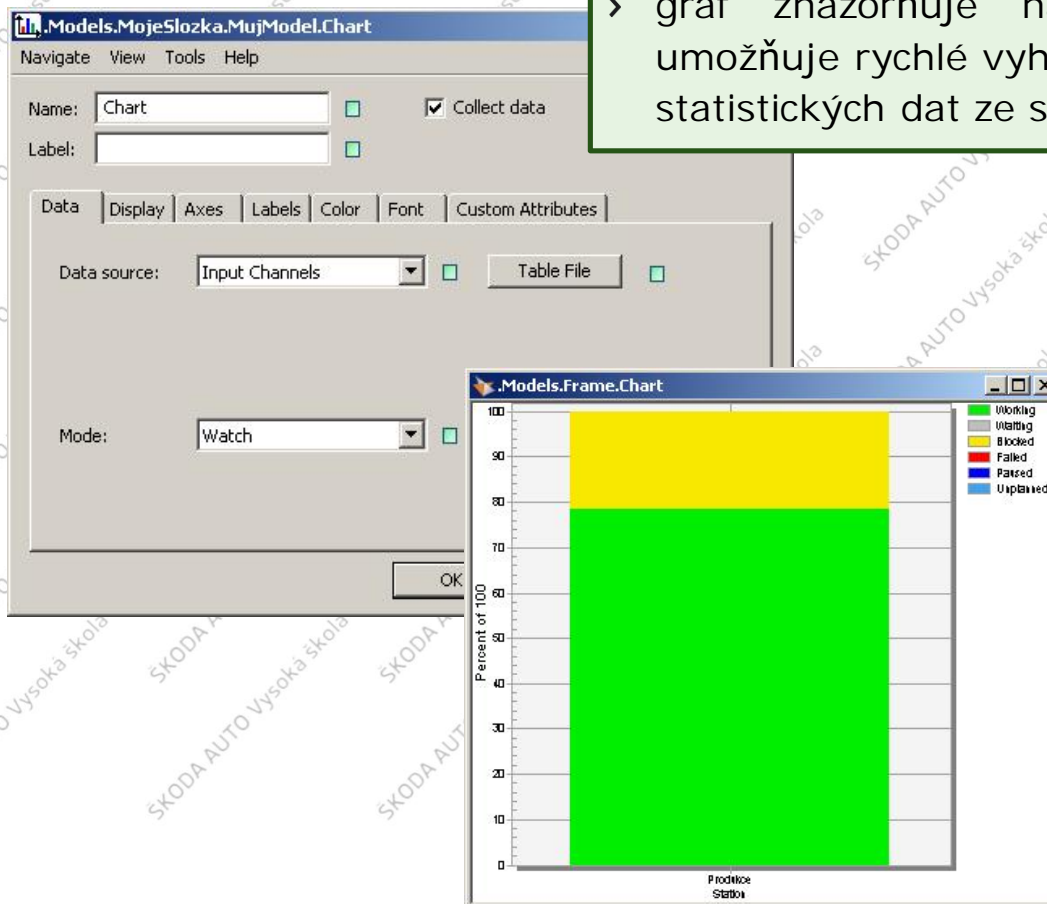


# Počítačová simulace logistických procesů I. - cvičení 02

## Graf



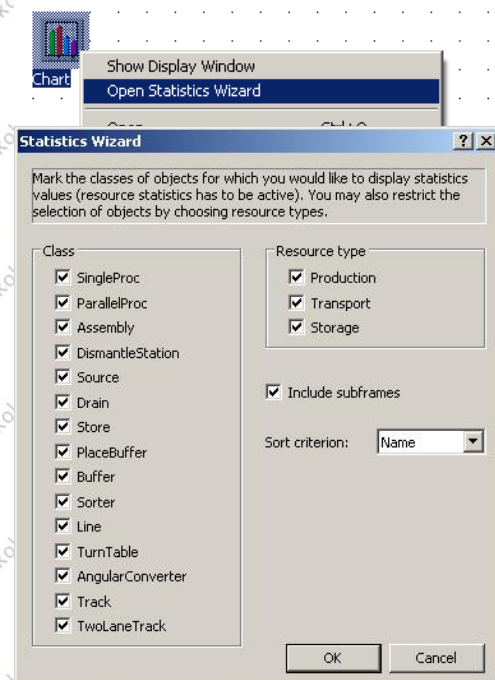
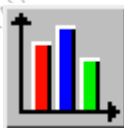
- › kapacita - 0
- › prvek plochy (Toolbox\User Interface)
- › graf znázorňuje např. statistické údaje a umožňuje rychlé vyhodnocení jakož i prezentaci statistických dat ze simulačních běhů



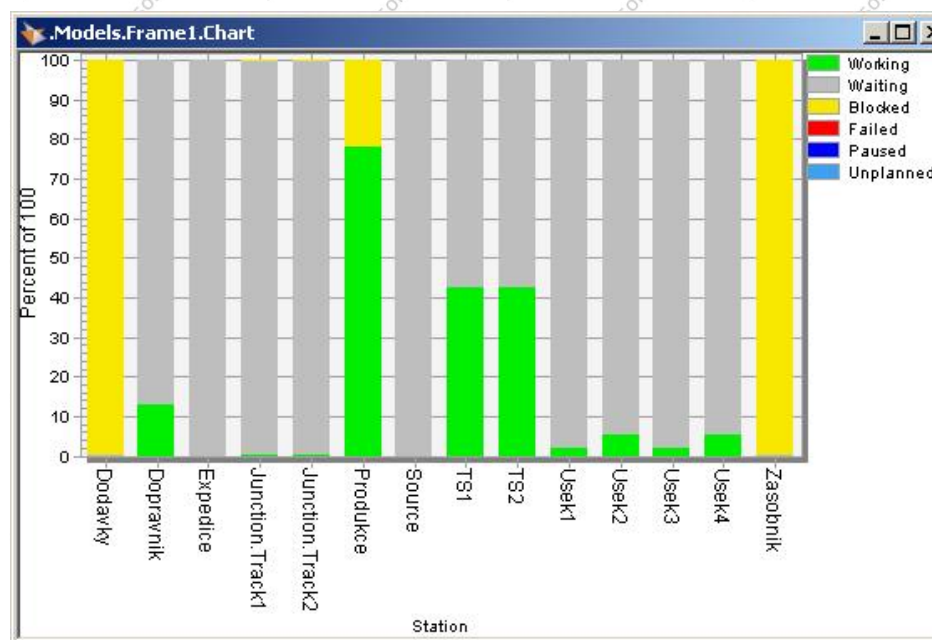


# Počítačová simulace logistických procesů I. - cvičení 02

## Graf



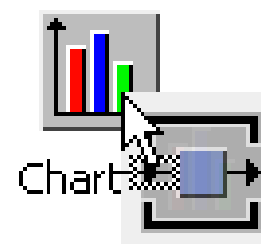
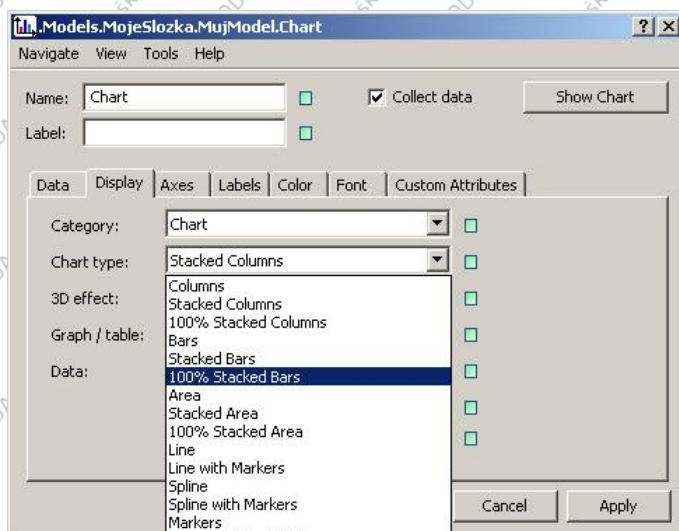
- › otevřeme-li menu pravým tlačítkem na myši u aktivovaného objektu grafu a vybereme-li Open Statistics Wizard, aktivujeme defaultně všechny objektové třídy v modelu
- › ty, které si nepřejeme zobrazit, jednoduše deaktivujeme





# Počítačová simulace logistických procesů I. - cvičení 02

## Graf – zdroje dat



- › chceme-li aktivovat v grafu pouze jednotlivé objekty, jednoduše přetáhneme příslušné objekty materiálového toku na ikonu grafu
- › velice pohodlně zvolíme typ grafu (sloupcový, bodový atd.)

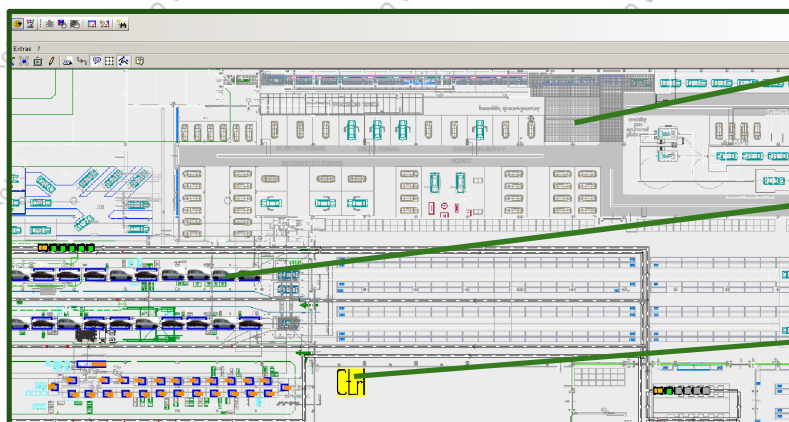




## Počítačová simulace logistických procesů I. - cvičení 02

### Editor oken

- › Editor ikon umožňuje grafickou vizualizaci objektů prostřednictvím zvolených obrázků a jejich animaci.
- › Pro zobrazení téměř všech objektů v modelu lze použít jakékoli další vlastní ikony, nejen ty přednastavené. U objektů materiálového toku, MUs a sítí lze kromě toho definovat též animační body, popř. linie pro vizualizaci MUs na daných objektech.
- › Je možné též použít obrázky na pozadí sítě. Můžeme použít např. layout nějaké výrobní haly, který převedeme do podoby kompatibilní s Plant Simulation a vložíme ho na pozadí. Jednotlivé objekty modelu pak umísťujeme přesně tak, jak jsou, popř. budou, orientovány ve skutečnosti.



layout montážní haly použitý jako pozadí sítě

pohybující se objekty

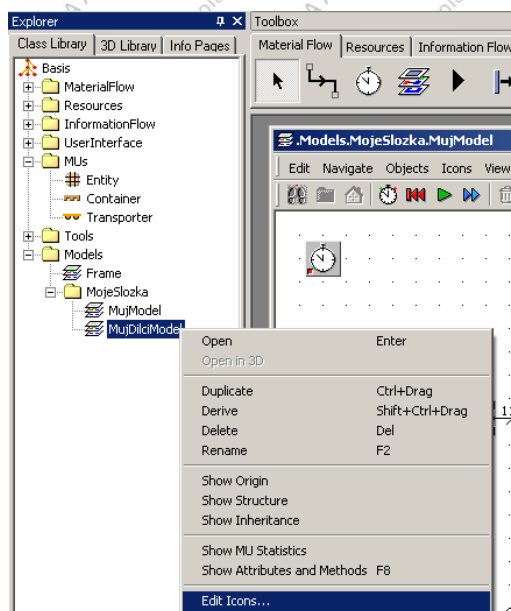
vložená síť řízení



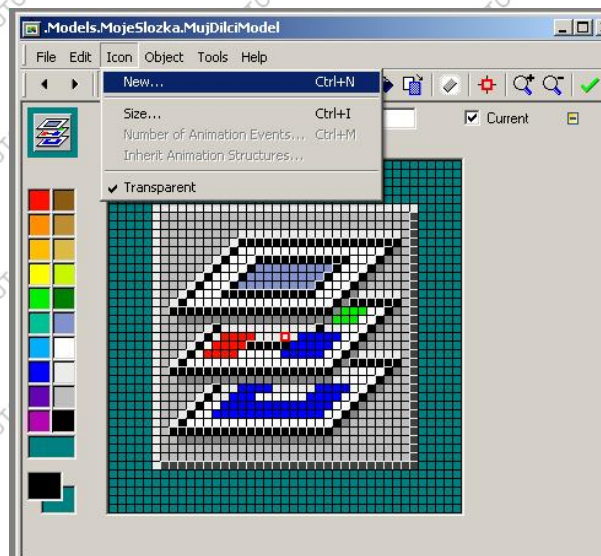


# Počítačová simulace logistických procesů I. - cvičení 02

## Změna ikony



- › Editor ikon se otevře v Class Library aktivováním objektu sítě a rozbalením příkazové nabídky pravým tlačítkem na myši.
- › Lze ponechat stávající ikony a přiřadit libovolné množství dalších ikon, kde lze měnit velikost atp.





# Počítačová simulace logistických procesů I. - cvičení 02

## Hlavní okno editoru ikon

pořadí ikon

listování mezi ikonami

velikost originálu

paleta barev

transparentní barva

aktuální barva

aktuální poloha kurzoru

přepnutí do animačního módu

převzetí barvy „pipeta“

kreslicí nástroje (čáry, tvary, vyplňování, mazání)

přesun referenčního bodu

uložení změn

aktuálně používaná ikona

název ikony

kreslicí plocha

referenční bod

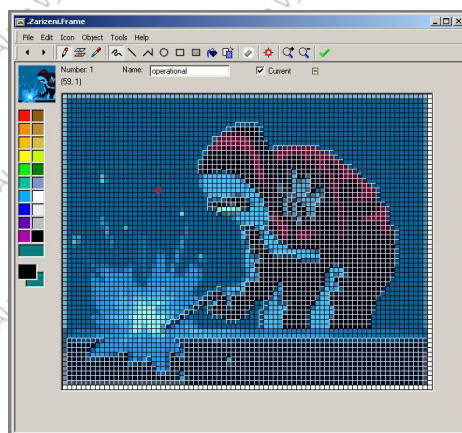
Obrázek ikony je možné buď vytvořit, nebo vložit soubor s obrázkem -  
File ⇒ Open (vybrat požadovaný soubor kompatibilního formátu - .gif, .bmp, .ppm, .ppm raw, .dxf, .dwg)  
nebo použít funkcionalitu Drag&Drop a soubor do kreslicí plochy „přetáhnout“.



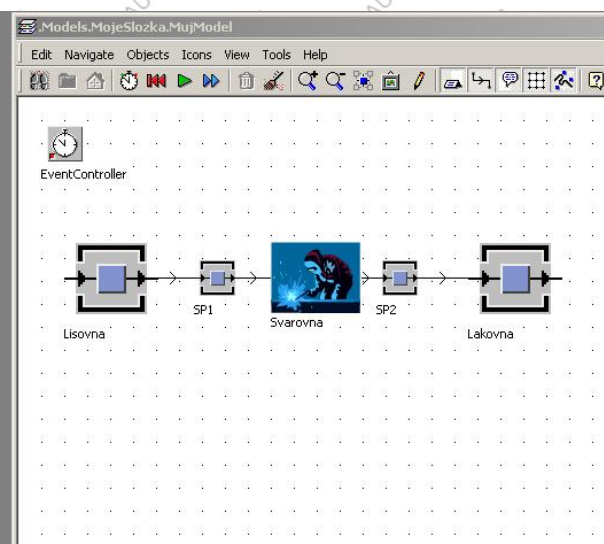
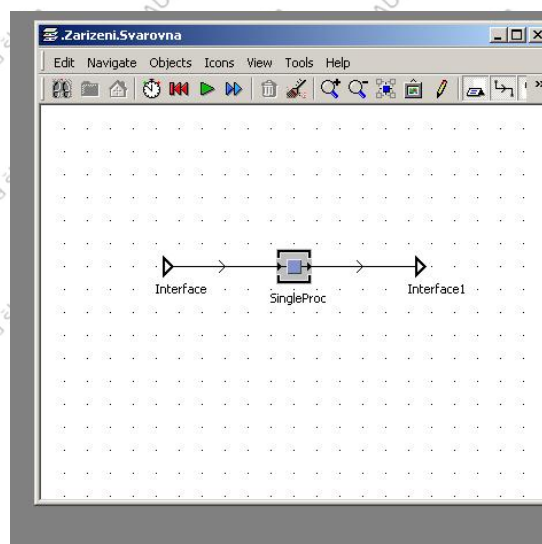


# Počítačová simulace logistických procesů I. - cvičení 02

## Editor oken



- Vlastní ikonu lze přiřadit také síti. Zde je uveden příklad, kdy je v modelu jeden objekt SingleProc nahrazen samostatnou sítí. V síti je pak ikoně s názvem operational přiřazen obrázek.

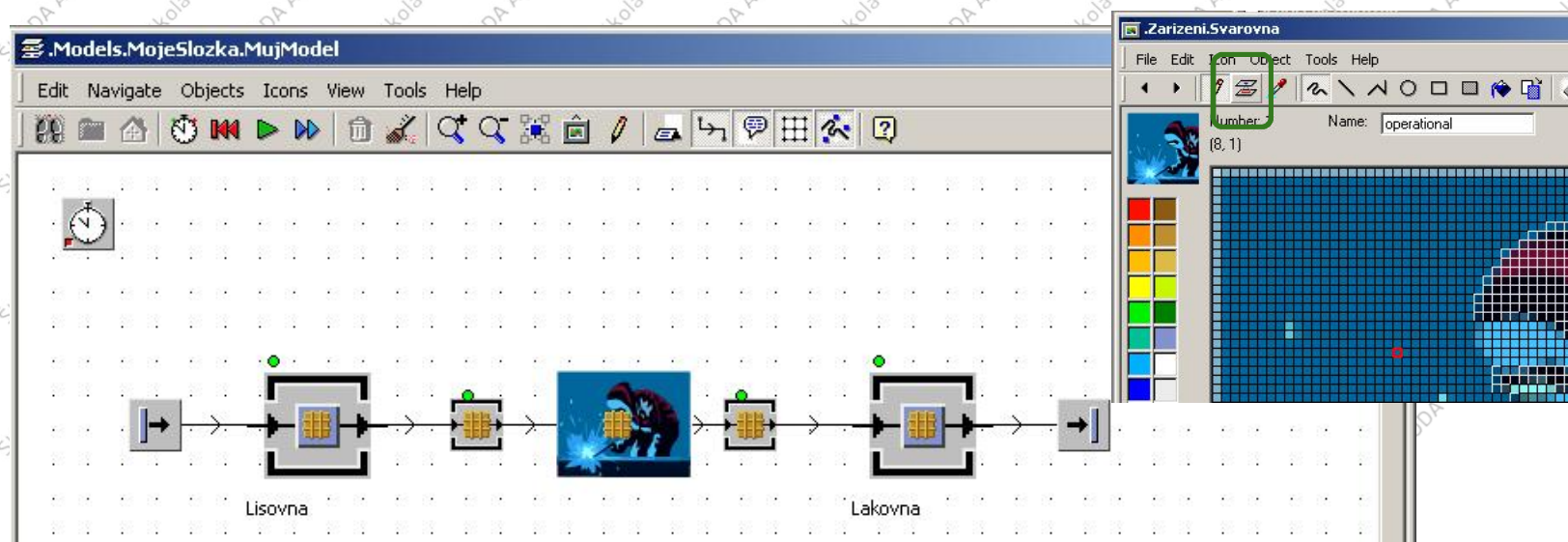




# Počítačová simulace logistických procesů I. - cvičení 02

## Animace ikon

- Pomocí animačních bodů lze vizualizovat pohyb MU i v rámci hierarchie sítí (tedy u sítí vložených do jiných sítí).
- Tyto body představují spojení mezi ikonou a objektem, který leží v dané síti. MUs, které při simulačním běhu jsou momentálně na daném objektu, se objeví na příslušném animačním bodě ikony.
- Podle typu objektu lze přiřadit i animační přímky.
- Nastavení animačních bodů/přímek se provádí v animačním módu editoru ikon.
- Animační body lze přiřadit pouze objektům tříd.

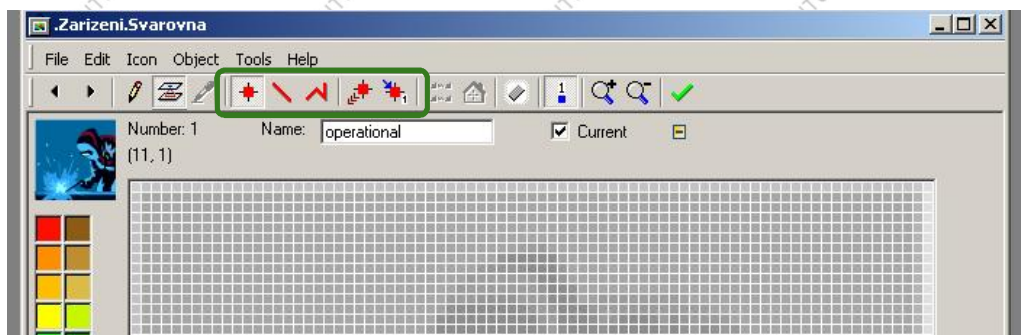




## Počítačová simulace logistických procesů I. - cvičení 02

### Postup pro vytvoření jednoduché animace ikony

- Nastavení animačních bodů/přímek se provádí v animačním módu editoru ikon.
- V menu editoru jsou kreslicí nástroje (pro vkládání animačních bodů a přímek) a nástroje pro propojení (ikony s objekty v síti).

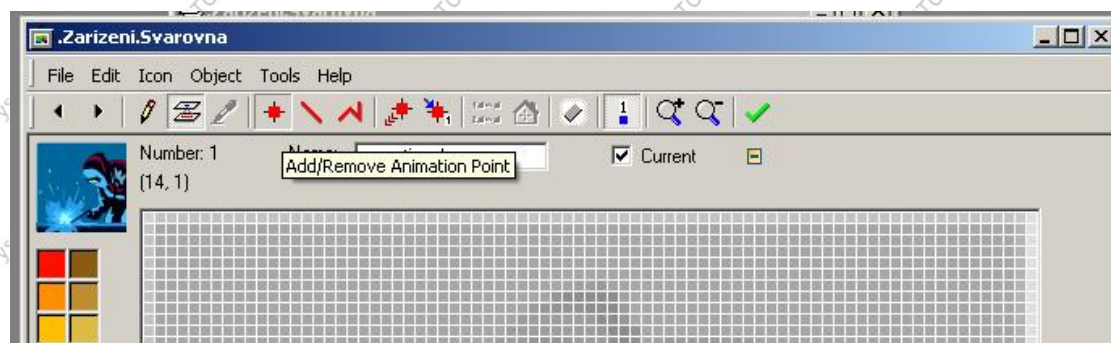




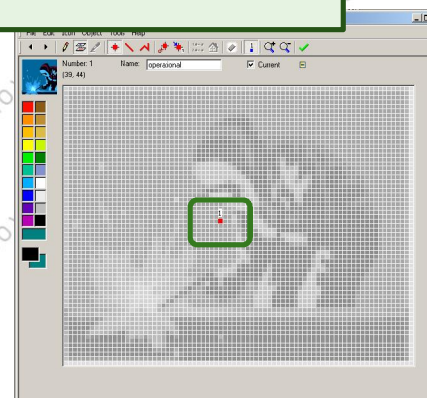
# Počítačová simulace logistických procesů I. - cvičení 02

## Postup pro vytvoření jednoduché animace ikony

- Nejprve je nutné aktivovat nějaký animační nástroj, např. přidání animačního bodu




- Poté klikneme na požadované místo vložení na ikoně, kde chceme, aby animace byla viditelná.

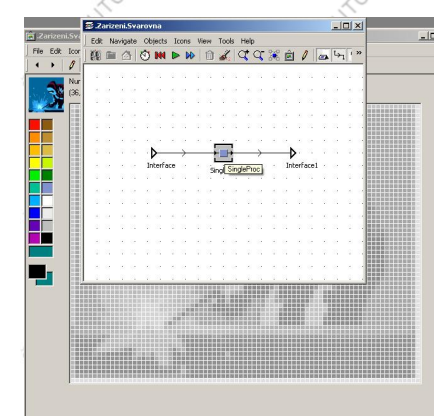
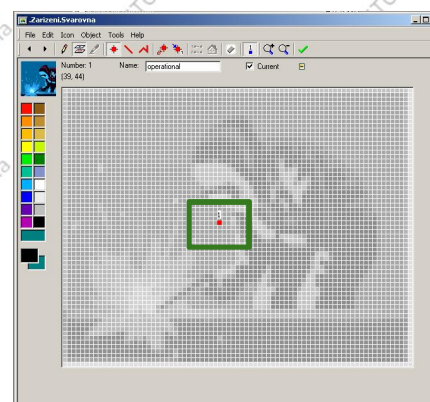
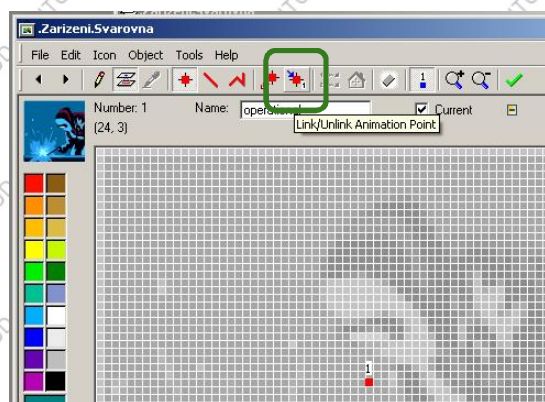




# Počítačová simulace logistických procesů I. - cvičení 02

## Postup pro vytvoření jednoduché animace ikony

- Aktivujeme nástroj propojení animačního bodu.
- Klikneme na animační bod, tím se automaticky otevře síť, pro kterou ikonu upravujeme.
- Klikneme na animovanou stanici (SingleProc). Pokud bychom v síti měli více stanic, mohli bychom tento postup opakovat pro další stanice.
- Ujistěte se, že je aktivováno pole Current (tzn., že daná ikona bude použita).
- Je možné změnit i název ikony.
- Všechny změny je nutné uložit pomocí  - Apply Changes.
- Daný obrázek ikony lze též uložit na pevný disk přes *File - Save As...*





## Počítačová simulace logistických procesů I. - cvičení 02

### Editor ikon - doplňkové informace

- Pokud vytváříme vlastní ikony (ne vkládáme žádné hotové obrázky) používáme kreslicí nástroje na nástrojové liště v menu editoru. Můžeme kreslit volně, přímky, lomené přímky, elipsy, obdélníky, plné obdélníky, můžeme vyplňovat oblasti barvou, nebo oblasti kopírovat a vkládat jejich kopie.
- Pipeta - tento nástroj používáme k výběru barvy z kreslicí plochy, aktivní barva bude pak barva vybraná na ploše.
- Referenční bod se používá pro určení pozice dané použité ikony objektu v síti a k znázornění MUs na tomto objektu (příklad - při animaci se kryje referenční bod SingleProc s referenčním bodem pohybujícího se objektu, který se momentálně v daném objektu nachází. Defaultně leží referenční bod v pozici 20 x 20 pixelů od levého horního okraje ikony.
- Transparentní barva - použijeme-li v ploše ikony transparentní barvu, znamená to, že v tomto místě bude ikona „průhledná“ a bude vidět pozadí sítě, do které je daný objekt vložen.
- Aktuálně použitá ikona má aktivované políčko Current.
- Maximální velikost ikony je 4000 x 4000 pixelů.





## Počítačová simulace logistických procesů I. - cvičení 02

### Editor ikon - doplňkové informace

- Pokud pojmenujeme ikonu názvem background, použije se tento obrázek jako pozadí sítě.
- Používání uživatelsky definovaných ikon a jejich animací zpomaluje běh simulace.
- Mají-li ikony speciální názvy, potom je možné jejich automatické přepínání (working, setUp, no\_Entry, failed, pause, waiting).
- Pokud nechcete, aby se ikony objektů měnily, je třeba tyto ikony přejmenovat (nesmí mít žádný ze speciálních názvů) nebo vymazat.





# Počítačová simulace logistických procesů I. - cvičení 02

## Atributy

- Každý objekt má celou řadu standardních atributů, jako např. délka, rychlost, čas, kapacita, název ikony, číslo ikony atp.
- Seznam standardních atributů a metod lze zobrazit přes kontextové menu objektu v knihovně tříd pomocí příkazu Show Attributes and Methods nebo aktivováním objektu a stisknutím tlačítka F8.
- Kromě toho může být většině objektů přiřazen libovolný počet dalších, uživatelem definovaných znaků - uživatelsky definované atributy. Práce s nimi je pak stejná jako se standardními atributy.





# Počítačová simulace logistických procesů I. - cvičení 02

## Tabulka atributů

- Atributy objektů začínají v seznamu velkým písmenem a mají nějakou hodnotu, na rozdíl od metod. Dvojitým kliknutím na řádek s atributem se otevře dialog, v němž lze hodnotu změnit nebo přiřadit. Tato funkcionality není aktivní u všech atributů.

datový typ  
atributu nebo  
argumentu

hodnota  
atributu

hodnota zděděná/  
nezděděná

je možné tento atribut  
monitorovat?

název atributu

německý ekvivalent

Name	Signature	Value	inherited / not inherited	Watchable	Name
addObserver	(string,method)				FügeBeobachterHinzu
assignedWorkplaces	(table)				zugeordneteArbeitsplatze
attributeWatchable	(string) : boolean				attributUberwachbar
AutomaticSetUp	boolean	true	i	*	AutomatischesRuesten
Availability	real	100	i		Verfuegbarkeit
ChangePathCtrl	method	VOID	i		PfadAendernStellen
ChannelID	string	3d	i	*	KanalID
ChildNo	(integer) : object				KindNr
Class	object	.MaterialFlow.SingleProc			Klasse
closeDialog	((boolean) : boolean)				schliesseDialog
closeImg	(boolean) : boolean				schliesseBild
connectAutomatically	boolean				verbindeAutomatisch
ConnectCtrl	method	VOID	ni		VerbindenStellen
ConstructorCtrl	method	VOID	ni		KonstruktorStellen
cont	object	VOID		*	Inhalt
ContentsList	(table)				Inhaltsliste
copyToClipboard					kopiereInZwischenspeicher
createAttr	(string,string) : boolean				erzeugeAttr
createIcon	((string,integer,integer) : i...				erzeugeBild
createObject	(object,integer,integer[,stri...				erzeugeObjekt
CurrIcon	string	operational	i	*	BildName
CurrIconNo	integer	1	i	*	BildNr
CurrIconTransparent	boolean	true	i		BildTransparent
CycleTime	time	0.0000	i		Zykluszeit
delAniPoint	(integer) : boolean				loescheAniPunkt
delAniPoints	boolean				loescheAniPunkte
deleteAttr	(string) : boolean				vernichteAttr
deleteIcon	(any)				loescheBild
deleteMovables					vernichteRFFs





# Počítačová simulace logistických procesů I. - cvičení 02

## Datové typy

- Datové typy definují rozsahy hodnot dat, která jsou v objektech jako parametry, vstupní údaje nebo proměnné.
- Standardně nabízí Plant Simulation několik datových typů.

<b>boolean</b>	TRUE nebo FALSE
<b>integer</b>	celé číslo
<b>real, length, weight, speed, money</b>	číslo s plovoucí desetinnou čárkou
<b>string</b>	posloupnost znaků
<b>date</b>	datum (rrrr/mm/dd)
<b>time</b>	čas (hh:mm:ss.ssss)
<b>datetime</b>	datumočas (rrrr/mm/dd hh:mm:ss.ssss)
<b>list, stack, queue</b>	seznam s jedním sloupcem (fronta - FIFO, stoh - LIFO)
<b>table</b>	tabulka s více sloupci
<b>object</b>	odkaz na objekt





# Počítačová simulace logistických procesů I. - cvičení 02

## Uživatelsky definované atributy

MojeMUs.Entity

Name: Entity

Label: Conveying direction: 0 (forward)

Attributes | Product Statistics | Custom Attributes

New Edit Delete

Name	Value	Type	C.. I..	Channel

OK Cancel Apply

- Pomocí uživatelsky definovaných atributů lze objektům přiřadit volitelné vlastnosti a znaky. Na tyto vlastnosti a znaky se lze během simulace dotázat, je možné je přiřazovat nebo měnit jako standardní atributy.
- Příkladem takového atributu je barva, velikost, typ zboží, stupeň zpracování, číslo zakázky atd.
- Počet uživatelsky definovaných atributů není nijak omezený.
- Skládá se z názvu (Name), datového typu (Data type) a hodnoty (Value). Hodnota nemusí být vyplněna, lze ji přiřadit či změnit později.

Custom Attributes

Name: barva

Value Statistics Communication

Data type: string

Value: cervena

OK Cancel Apply

MojeMUs.Entity

Name: Entity

Label: Conveying direction: 0 (forward)

Attributes | Product Statistics | Custom Attributes

New Edit Delete

Name	Value	Type	C.. I..	Channel
barva	cervena	string	*	





# Počítačová simulace logistických procesů I. - cvičení 02

## Method / metoda

- Metody jsou malé části programu, srovnatelné s procedurami nebo funkcemi v programovacích jazycích Basic, Pascal nebo C++.
- Programovací jazyk užívaný v Plant Simulation „SimTalk“ vznikl z programovacího jazyka „Eiffel“ a je velmi podobný ostatním jazykům.
- Metoda se skládá ze standardních metod, klíčových slov, přiřazování a kontrolních struktur. Seznam metod použitelných na objekt a ovlivnitelných atributů lze zobrazit přes kontextové menu ⇒ Show Attributes and Methods.
- Kromě toho lze definovat a ovlivňovat libovolně mnoho vlastností.
- Objekt Method je plně integrován do objektově orientovaného konceptu Plant Simulation. Překladač zpracovává zdrojový kód během běžící simulace.





# Počítačová simulace logistických procesů I. - cvičení 02

## Method / metoda

### Možnosti metod

- reagovat na určité události během simulačního běhu
- dotázat se na podmínky a stanovit je
- provést příkazy
- změnit a rozšířit chování objektů
- připravit model pro uživatele s novými dialogy

### Užitek a přednosti

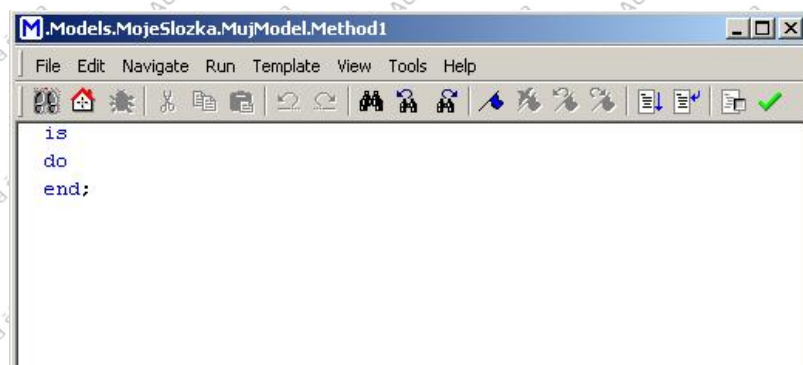
- realizovat představy, projektové požadavky, speciální požadavky na model
- model je velice výkonný
- model je velmi flexibilní





# Počítačová simulace logistických procesů I. - cvičení 02

Method / metoda



- objekt informačního toku
- nulová kapacita
- je-li vytvořena instance z originálu metody z knihovny tříd, je mechanismus dědičnosti z důvodů zjednodušení deaktivován
- dědičnost se uplatňuje v síťové hierarchii a u duplikovaných metod, jak je obvyklé





# Počítačová simulace logistických procesů I. - cvičení 02

## Struktura metody

Metoda se rozděluje do oddílů. Pokud nějaký oddíl není potřebný, odpadá:

### Parametry

dodatečné informace objektu, který metodu vyvolal, metoda se tak stává funkcí, počet předaných a deklarovaných parametrů i datové typy musí souhlasit

### Výsledek funkce

výsledek funkce (musí být při vrácení hodnoty přiřazen na vyvolávací objekt pomocí klíčového slova result

### **is**

uvádí deklaraci lokálních proměnných, ty jsou k dispozici pouze v této metodě a ztrácejí svoji hodnotu po provedení kódu

### **do**

uvádí příkazový oddíl - vlastní zpracování metody - standardní metody, přiřazování, rozbočky, smyčky, volání jiných metod atp.

### **end;**

uzavírá metodu, poté jsou možné už jen komentáře





# Počítačová simulace logistických procesů I. - cvičení 02

## Operace SimTalk

### Přiřazení hodnot

slouží k přiřazení hodnoty proměnné:

např.:

```
Lisovna.ProcTime := 120;
```

- objektu lisovna se přiřadí doba zpracování 120 s

### Aritmetické operace

sčítání, odčítání, násobení, dělení ( + - \* / ) a standardní funkce -  
trigonometrické funkce, logaritmus, exponenciální funkce

### Logické operace

výrazy se sloučí do relací, symboly relací ( =, /=, >, >=, <, <=, == ),  
hodnota relace může nabít hodnoty TRUE nebo FALSE, typ Boolean,  
dále sem patří negace, logické operace NEBO a A: **and, or, not**

### Vstupní a výstupní operace

slouží k datovému transportu vstupních a výstupních parametrů





# Počítačová simulace logistických procesů I. - cvičení 02

## Konvence

V nápovědě k Plant Simulation platí pro usnadnění čtení následující konvence týkající se pravopisu:

- formulace v metodách jsou psány doporučním písmem
- názvy atributů začínají velkým písmenem (např. **Lisovna.Class**)
- názvy metod začínají malým písmenem, skládá-li se metoda ze dvou slov, začíná druhé slovo velkým písmenem (např. **Lisovna.removeObserver**)
- klíčová slova jsou psána **modrým písmem**
- komentáře jsou psána **zeleným písmem**





# Počítačová simulace logistických procesů I. - cvičení 02

## Komentáře

- Komentáře slouží k připojení poznámek k textu metody pro lepší srozumitelnosti formulovaných příkazů. Doporučuje se také připojit, kterým objektem je metoda volána.
- Jednořádkové komentáře začínají dvěma pomlčkami „- -“ a končí na konci řádku.
- Víceřádkové komentáře se uvádějí posloupností „/\*” a končí znaky „\*/”.
- Text komentáře se v okně metody znázorní zelenou barvou.

```
M.Models.MojeSlozka.MujModel.Method1
File Edit Navigate Run Template View Tools Help
-- jednořádkový komentář
is -- použití možné i na konci řádku
do
  /* víceřádkový
  komentář */
end;
```





# Počítačová simulace logistických procesů I. - cvičení 02

## Příkazy

- jednoduchá sekvence příkazů, lokální proměnné, vstupní a výstupní parametry:

```
M.Models.MojeSlozka.MujModel.M1
File Edit Navigate Run Template View Tools Help
is
  soucet : real;
do
  soucet := M2(13, 7.6);
  print "vysledek: ", soucet;
end;
```

přirazení – volání metody M2 s předáním parametrů

```
M.Models.MojeSlozka.MujModel.M2
File Edit Navigate Run Template View Tools Help
[(cislo1, cislo2 : real):real]
is
do
  result := cislo1 + cislo2;
end;
```

vstupní parametry, vratný parametr

přirazení hodnoty na klíčové slovo „result“ – nutné pro vrácení hodnoty na vyvolávací metodu

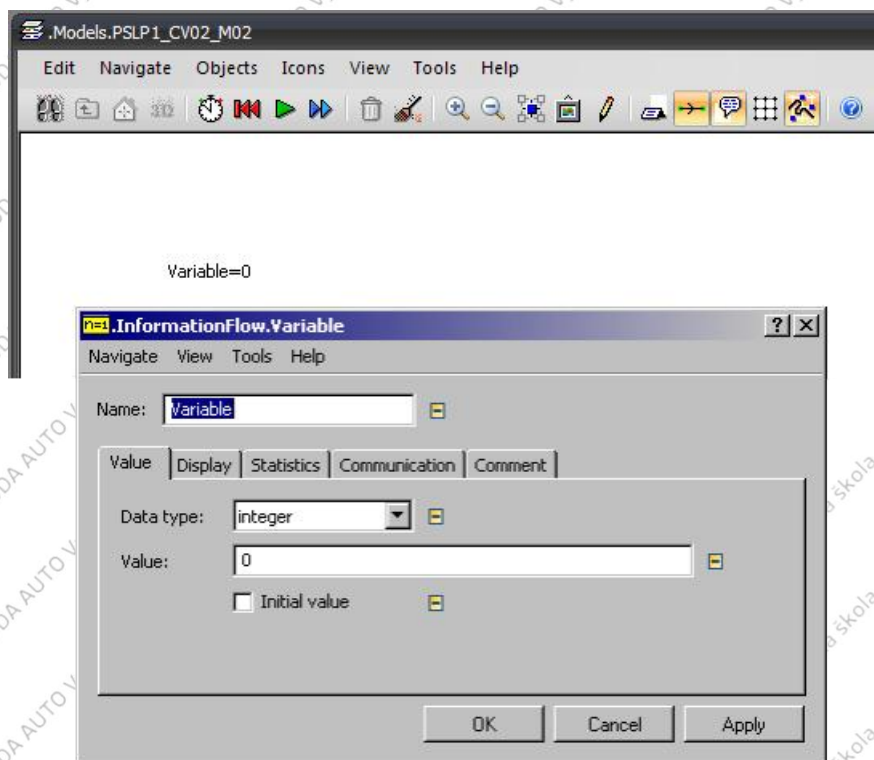




# Počítačová simulace logistických procesů I. - cvičení 02

## Variable / proměnná

n=1



- informační objekt
- nulová kapacita
- proměnná slouží k ukládání a zobrazení libovolných dat používaných v modelu
  
- proměnná může nabývat různých datových typů (string, integer, real, boolean, object, table, ...)
- proměnnou vkládáme přímo do sítě
- pro lepší čitelnost modelu je zachována následující barevná konvence:
  - **zelená** – proměnná, kterou je možno uživatelem libovolně definovat (vstupní parametrizace modelu)
  - **modrá** – proměnná použita pro statistiku
  - **černá** – proměnná, použita pro vlastní chod modelu (tuto proměnnou nelze uživatelem měnit, je použita výhradně pro zajištění chodu modelu)





# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M01

Téma:

- › průběžný zápis celkového počtu prošlých objektů do proměnné

Hlavní body:

- › každý průchod objektem bude zaznamenán do proměnné, která bude vložena do modelové sítě
- › tato proměnná bude nastavena na nulovou hodnotu, vždy při začátku nového simulačního experimentu
- › nastavení nulové hodnoty bude provedeno pomocí INIT metody, případně jako vlastní funkce vložené proměnné

Výchozí model:

- › PSLP1\_CV01\_M01

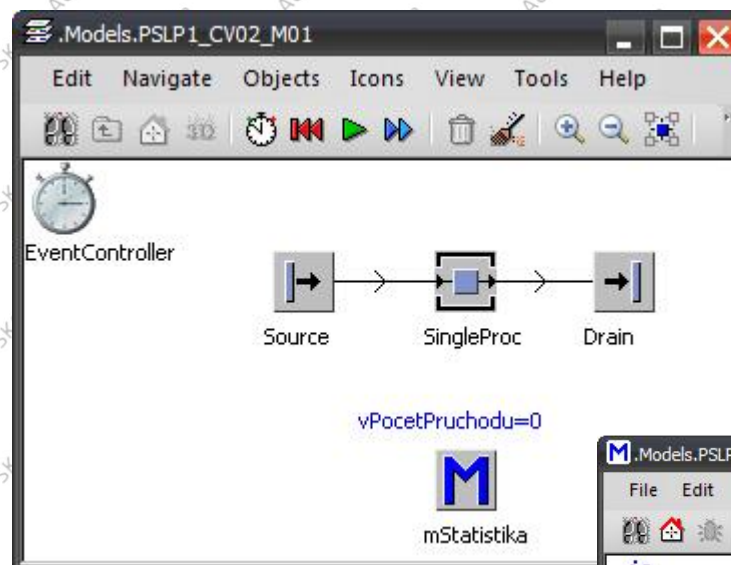
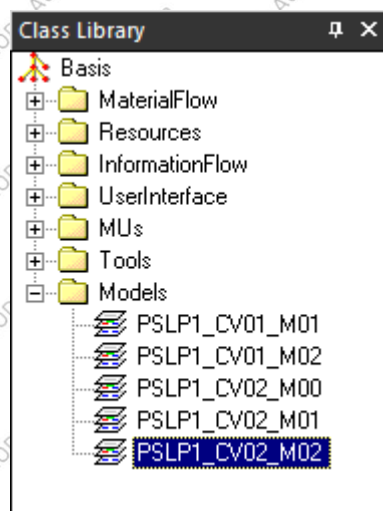




## Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M01

- založení modelu „PSLP1\_CV02\_M01“ na základě kopie modelové sítě „PSLP1\_CV01\_M01“
- vložení do sítě proměnnou „ vPocetPruchodu “ datového typu Integer (proměnná bude zobrazena modře)
- vložení metody s názvem „ mStatistika“ a zápis uvedených příkazů



Importer		Failure Importer		User-defined Attributes	
Times	Set-Up	Failures	Controls	Exit Strategy	Statistics
Entrance:	mStatistika			<input type="checkbox"/> Before actions	<input type="checkbox"/>
Exit:				<input checked="" type="checkbox"/> Front	<input type="checkbox"/> Rear
Set-up:				<input type="checkbox"/>	<input type="checkbox"/>

```

File Edit Navigate Run Template View Tools Help
is
do
    vPocetPruchodu := vPocetPruchodu + 1;
end;
  
```

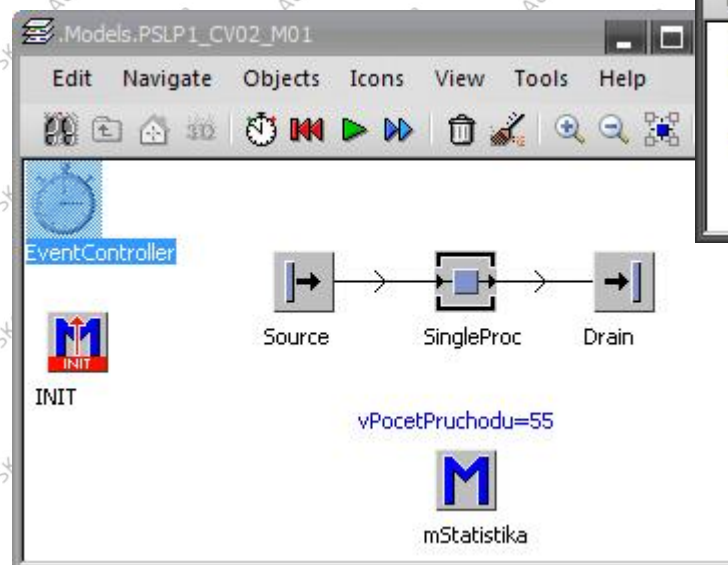
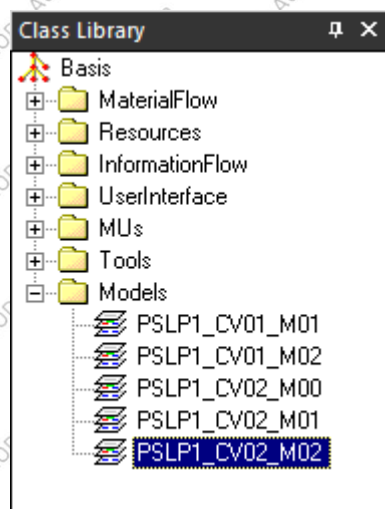




# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M01

- nastavení nulové hodnoty pomocí INIT metody



```
Models.PSLP1_CV02_M01.INIT
File Edit Navigate Run Template View Tools Help
is
dd
vPocetPruchodu := 0;
end;
```

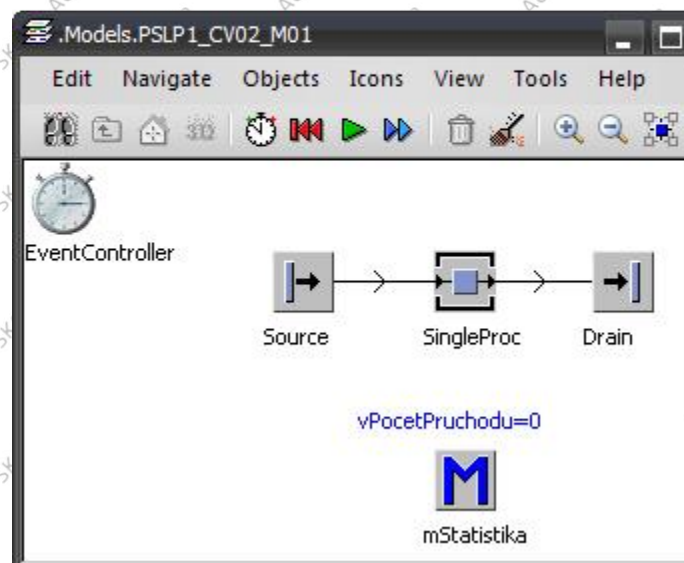
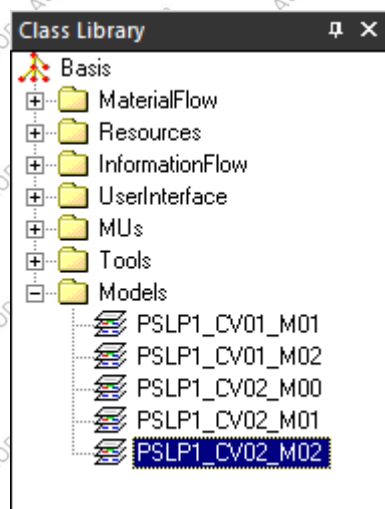




# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M01

- nastavení nulové hodnoty přímo v proměnné



n=1.Models.PSLP1\_CV02\_M01.vPocetPruchodu

Navigate View Tools Help

Name: vPocetPruchodu

Value Display Statistics Communication Comment

Data type: integer

Value: 0

Initial value

0

OK Cancel Apply





# Počítačová simulace logistických procesů I. - cvičení 02

## TableFile / tabulka



The screenshot shows a window titled ".InformationFlow.TableFile" with a menu bar (File, Edit, Format, Navigate, View, Tools, Help) and a toolbar. The main area contains a table with the following structure:

	string 1	string 2	string 3	string 4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				

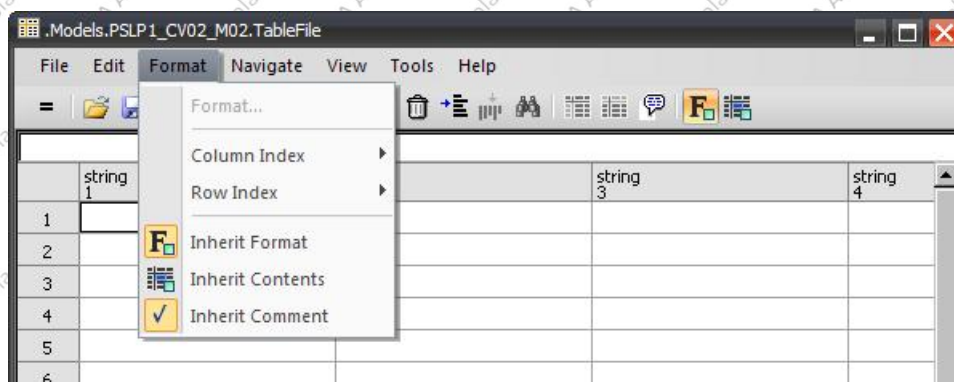
- informační objekt
- nulová kapacita
- tabulka slouží jako dvou nebo vícedimenzionální struktura k ukládání libovolných dat
- tabulku lze formátovat podle toho, k čemu bude sloužit
- je možné stanovit počet sloupků a řádků, přiřadit datový typ a dovolený rozsah hodnot jednotlivým sloupkům, různá přístupová práva, opatřit tabulku sloupkovým i řádkovým indexem pro lepší přehlednost atp.





# Počítačová simulace logistických procesů I. - cvičení 02

## Dědičnost tabulky



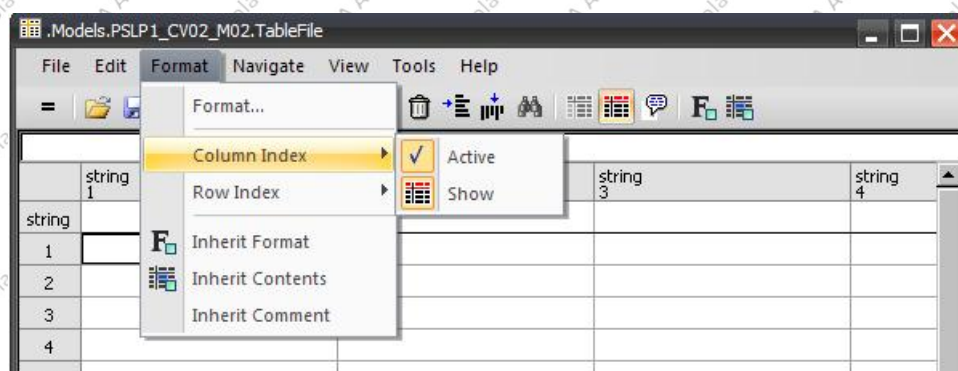
- vložíme-li tabulku do sítě z Toolboxu, je nejprve nutné deaktivovat dědění formátu - Inherit Format, popř. dědění komentáře - Inherit Comment v menu Format
- poté je možné provádět změnu formátu, jako např. změnu defaultního datového typu všech sloupků - string
- nastavení dědění obsahu - Inherit Contents je deaktivováno automaticky, aby vůbec bylo možné do tabulky něco vložit





# Počítačová simulace logistických procesů I. - cvičení 02

## Sloupkový a řádkový index tabulky



- pro lepší přehlednost je lepší aktivovat sloupkový, popř. řádkový index v menu Format ⇒ Column Index, Row Index - tím je možné vyplňovat záhlaví řádků a sloupků
- sloupkový a řádkový index se zvýrazní černým okrajem a má vždy číselné označení 0, tzn. nultý sloupek, popř. nultý řádek





# Počítačová simulace logistických procesů I. - cvičení 02

## Formát tabulky - dimenze

string 0	string 1	string 2	string 3	
string				
1				
2				
3				
4				
5				

List Format

Settings | Permissions | Dimension | Data Type

Data type: String

Format string:

Range: {1,1}..{\*,\*}

OK Cancel Apply

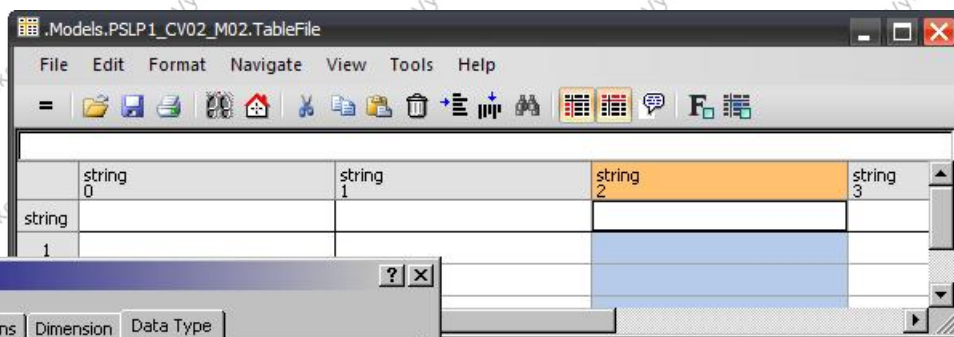
- pro změnu týkající se celé tabulky, např. jejích rozměrů tabulky, je nutné označit celou tabulku pomocí levého horního okraje tabulky - kliknutím sem se označí celá tabulka kromě záhlaví (zmodrá)
- v menu Format ⇒ Format... se nastavují nejdůležitější formáty tabulky
  - datový typ sloupku / tabulky
  - dimenze - počet řádků, sloupců (pokud je chceme z nějakého důvodu omezit)
  - oprávnění - přístup pro čtení / zápis
  - nastavení - barva písma, velikost písma, barva pozadí





# Počítačová simulace logistických procesů I. - cvičení 02

## Formát tabulky - datový typ



- datový typ sloupku změním tak, že označím záhlaví a v menu Format ⇒ Format... (nebo přes rozbalenou příkazovou nabídku po kliknutí pravou myší Format...) nastavím požadovaný typ, změnu potvrdím pomocí Apply a celou nabídku uzavřeme přes OK

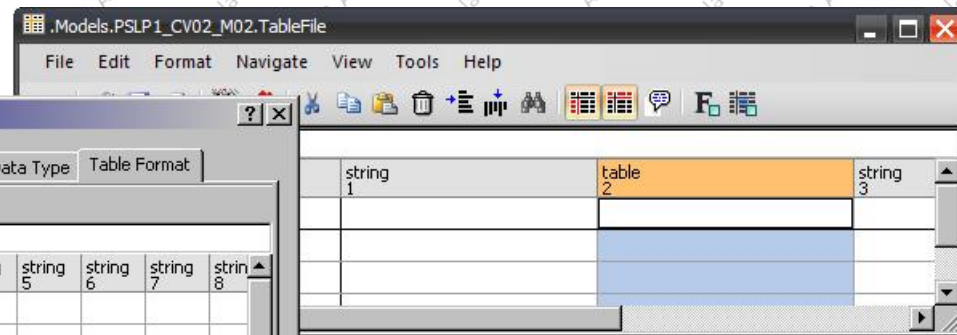
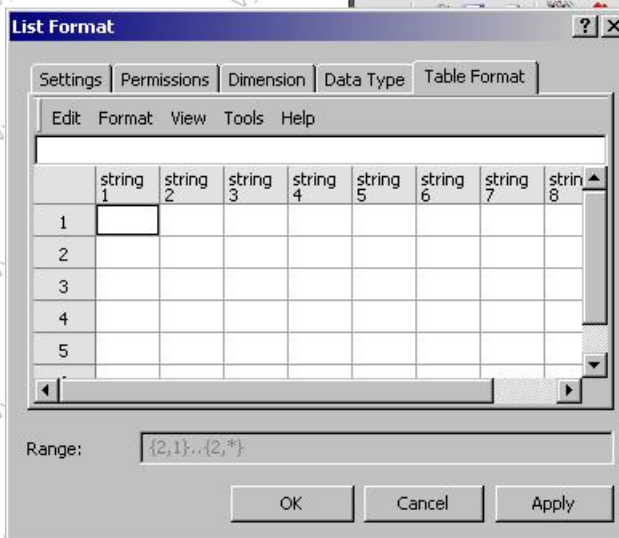
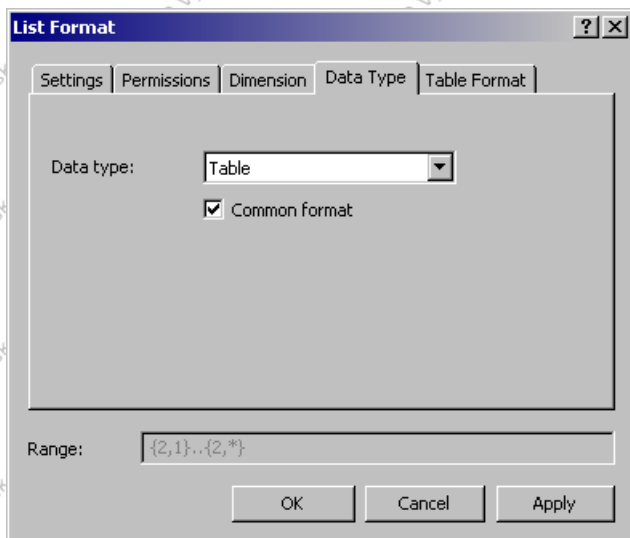
- datové typy jsou možné - boolean, integer, real, string, object, table, list, stack, queue, time, money, length, weight, speed, acceleration, data a datetime
- pro datové typy integer, real, length, weight, speed, acceleration, money a string lze pomocí Format string (formát posloupnosti znaků) omezit zadávání dat v dialogu (např. A - pouze písmena, C - pouze velká písmena a číslice, N - pouze číslice, u datového typu real znamená Format string „-15.2“, že číslo může mít celkem 15 číslic, z toho jsou 2 číslice po desetinné čárce, znak „-“ znamená, že jsou možná i záporná čísla)





# Počítačová simulace logistických procesů I. - cvičení 02

## Vnořené tabulky - tabulky v tabulkách



- zvolíme-li pro nějaký sloupec datový typ Table, vložíme vlastně do tabulky další tabulku - vnořenou tabulku
- tato vnořená tabulka se formátuje přes záložku Table Format
- každý řádek takového sloupku tedy představuje další tabulku, je-li aktivována volba Common format, znamená to, že všechny vnořené tabulky v daném sloupcu budou mít stejný formát
- vnořenou tabulku otevřeme buď přes kontextové menu (příkazová nabídka rozbalená přes pravé tlačítko na myši) nebo tlačítko F2, je-li kurzor v daném řádku a tato buňka není prázdná (stačí napsat „x“)



# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M02

Téma:

- › vytvoření tabulky pomocí metody

Hlavní body:

- › osvojení práce s tabulkami
- › využívání proměnných vložených do sítě modelu
- › použití zastavení metody s následným krokováním
- › získání znalostí s programování pomocí příkazů určených pro tabulky

Výchozí model:

- › prázdná základní síť

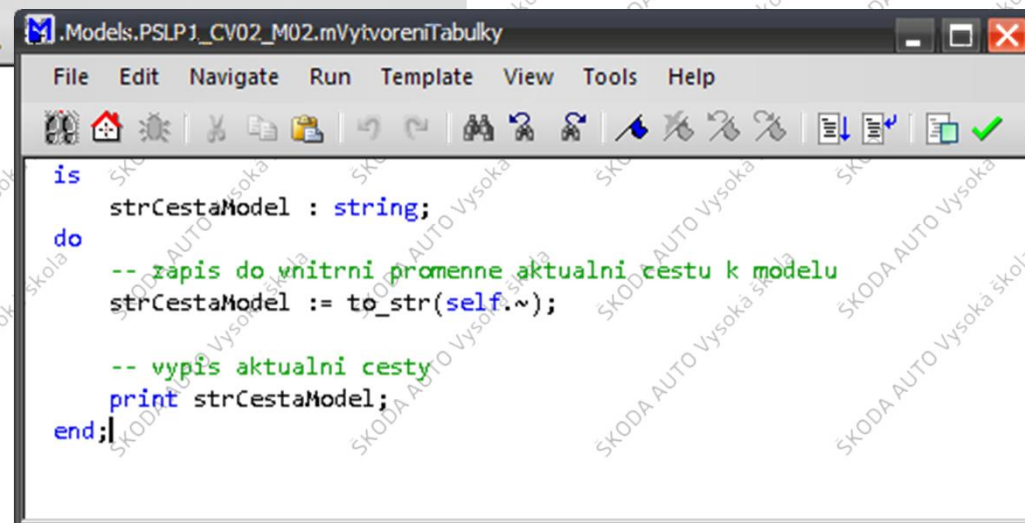
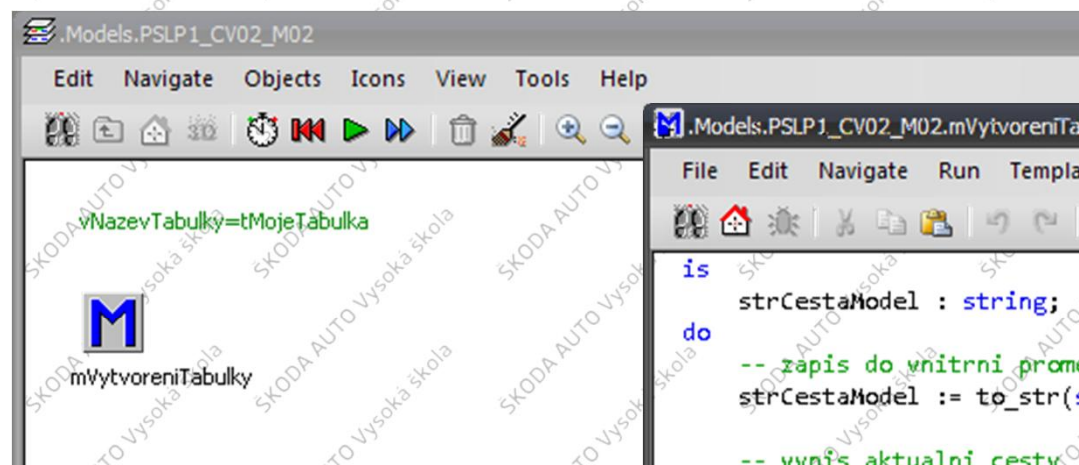
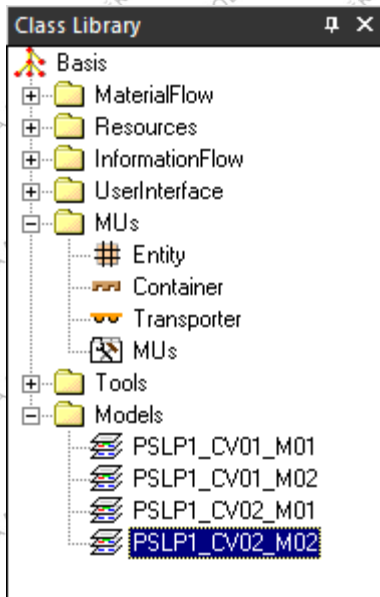




## Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M02

- založení modelové sítě s názvem „PSLP1\_CV02\_M02“
- vložení do sítě proměnnou „vNazevTabulky = tMojeTabulka“ datového typu String (proměnná bude zobrazena zeleně)
- vložení metody s názvem „mVytvoreniTabulky“
- v prvním kroku zápis do metody následujících příkazů
- pomocí těchto příkazů je získána aktuální pozice modelu (self.~)

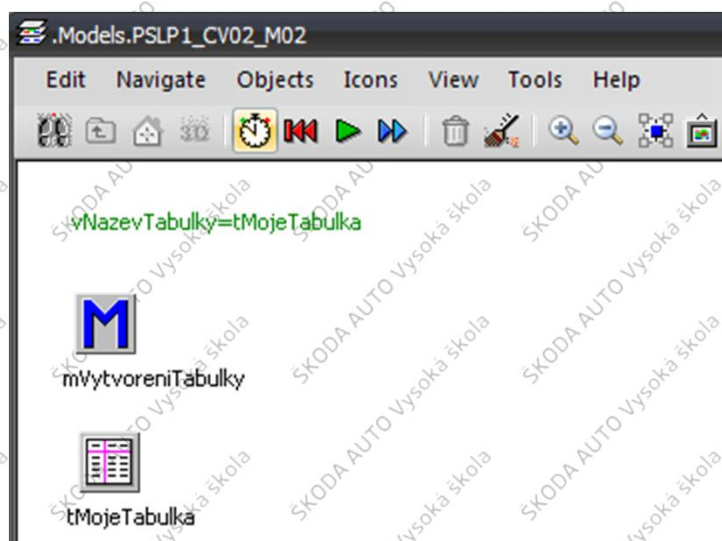




# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M02

- rozšíření metody o následující příkazy
- po spuštění této metody, dojde k vytvoření požadované tabulky v aktuální síti.



```

M.Models.PSLP1_CV02_M02.mVytvoreniTabulky
File Edit Navigate Run Template View Tools Help
is
  strCestaModel : string;
  strHodnota : string;
do
  debug;
  -- zapis do vnitřní proměnné aktuální cestu k modelu
  strCestaModel := to_str(self.~);

  -- overení existence tabulky
  strHodnota := strCestaModel + "." + vNazevTabulky;
  IF existsObject(strHodnota) = True THEN
    -- smazání stávající tabulky
    str_to_obj(strHodnota).deleteObject;
  END;

  -- vytvoření tabulky
  .InformationFlow.TableFile.erzeugeObjekt(self.~,45,150);

  -- změna názvu Tabulky na MojeTabulka
  self.~.TableFile.Name := vNazevTabulky;

  -- vypis aktuální cesty
  -- print strCestaModel;
end;|

```





# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M03

Téma:

- › základní funkce používané při práci s tabulkou (záznam hodnota)

Hlavní body:

- › průběžný záznam informací o prošlých objektech určitým pracovištěm
- › osvojení práce s tabulkami
- › využívání proměnných vložených do sítě modelu
- › použití zastavení metody s následným krokováním
- › získání znalostí s programování pomocí příkazů určených pro tabulky

Výchozí model:

- › PSLP1\_CV01\_M02

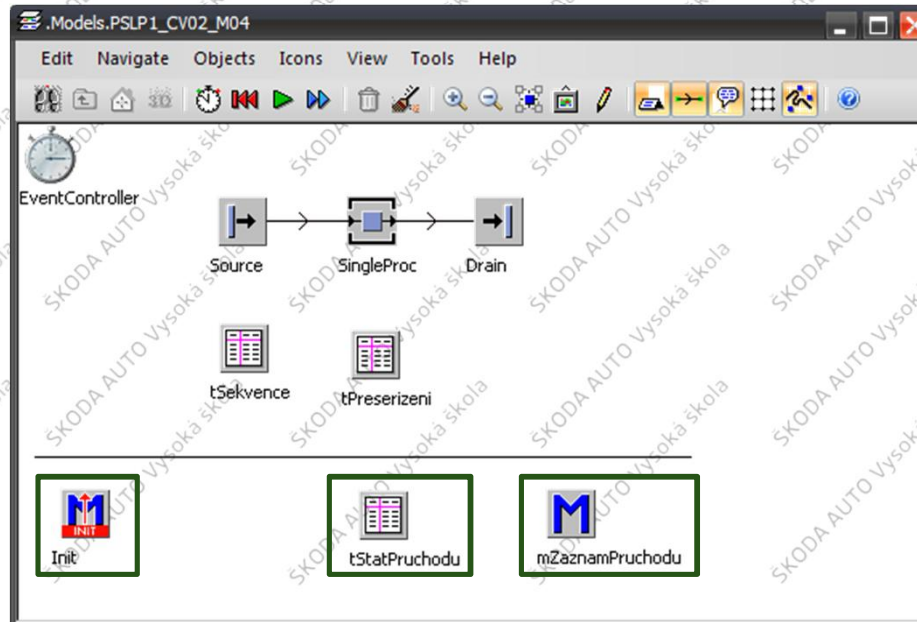
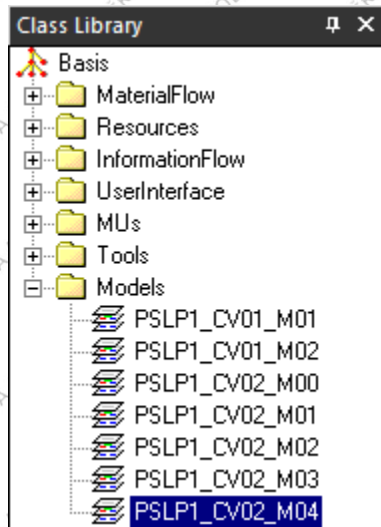




# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M03

- založení modelové sítě s názvem „PSLP1\_CV02\_M03 “
- vložení do sítě:
  - metody s názvem „Init“
  - tabulku s názvem „tStatPruchodu“
  - metodu s názvem „mZaznamPruchodu“





# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M03

- provedení naformátování tabulky
  - aktivování záhlaví sloupců
  - definování počtu sloupců a jejich datové typy
- připravení tabulky pro zápis dat (vytvoření záhlaví sloupců s názvy „AktualniCas“, „NazevMUs“, „CisloID“)

.Models.PSLP1\_CV02\_M04.tStatPruchodu

File Edit Format Navigate View Tools Help

AktualniCas

	string 1	string 2	integer 3
string	AktualniCas	NazevMUs	CisloID
1			
2			
3			
4			
5			

List Format

Settings Permissions Dimension Data Type

Number of rows:

Number of columns:

Column width:

Range:

OK

List Format

Settings Permissions Data Type

Alignment:

Font size:

Font color:

Background color:

Range:

OK Cancel Apply





## Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M03

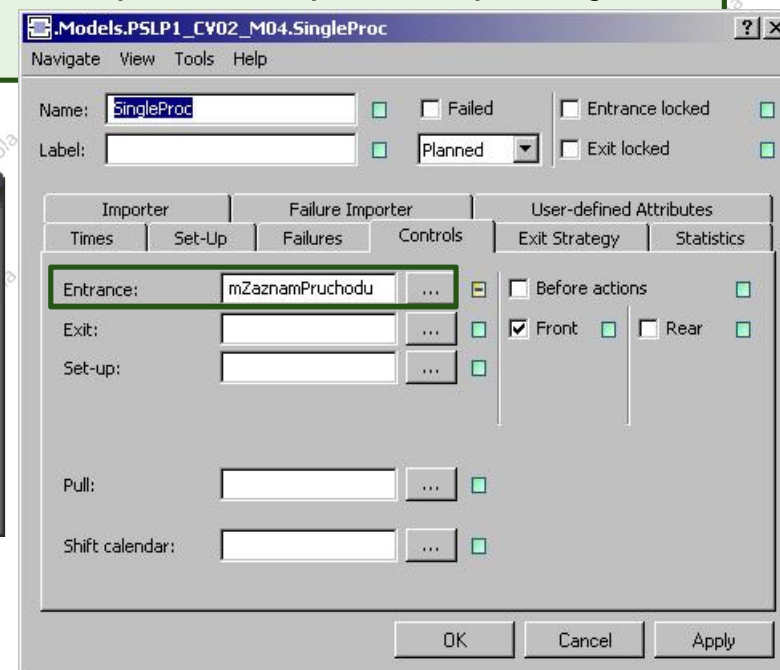
- programování metody „Init“ (při spuštění simulace jsou z tabulky „tStatPruchodu“ smazána veškerá předchozí data)
- programování metody „mZaznamPruchodu“
  - zjistí se poslední obsazený řádek v tabulce a do následujícího je provede zápis dat
  - definování zápisu dat pro jednotlivé sloupce
    - aktuální datum a čas
    - název MUs objektu
    - číslo MUs objektu
- metoda určená pro zápis „mZaznamPruchodu“ bude volána pokaždé při vstupu objektu na pracoviště „SingleProc“

```

M.Models.PSLP1_CV02_M04.Init
File Edit Navigate Run Template
View Tools Help
is
do
tStatPruchodu.delete();
end;

M.Models.PSLP1_CV02_M04.mZaznamPruchodu
File Edit Navigate Run Template View Tools Help
is
do
radek := integer;
radek := tStatPruchodu.YDim + 1;

tStatPruchodu["AktualniCas",radek] := to_str(EventController.AbsSimTime);
tStatPruchodu["NazevMUs",radek] := @.Name;
tStatPruchodu["CisloID",radek] := @.getNo;
end;
  
```





# Počítačová simulace logistických procesů I. - cvičení 02

Vzorový příklad: PSLP1\_CV02\_M03

- při spuštění simulace je možné vidět výsledek našeho snažení, do tabulky „tStatPruchodu“ jsou zaznamenávány požadovaná data prošlého objektu pracovištěm „SingleProc“

The screenshot displays the simulation software interface. The main window shows a process flow diagram with components: EventController, Dil\_03 Source, SingleProc, and Dil\_03 Dr. A tooltip for the 'Dil\_03' object shows the text: "Dil\_03: ,MUs.Entity:4552 on Drain". A green arrow points from this tooltip to a data table window titled ".Models.PSLP1\_CV02\_M04.tStatPruchodu".

The data table window shows the following data:

	string 1	string 2	integer 3
2013/01/01 00:00:00.0000			
	AktualniCas	NazevMUs	CisloID
4551	2013/01/03 17:14:25.8155	Dil_03	4551
4552	2013/01/03 17:15:08.8155	Dil_03	4552
4553			
4554			
4555			

On the right side, there is a control panel for the simulation, titled "...PSLP1\_CV02\_M04.EventController". It shows the current time as 2:17:15:51.8155 and includes playback controls (stop, play, next, previous) and a speed slider set to 10x real time.



