



Počítačová simulace logistických procesů II

7. přednáška – Struktura simulačního modelu

- › Jan Fábry
- › 23.10.2017



Počítačová simulace logistických procesů II

Obsah předmětu

- I. Úvod, organizace, semestrální projekty, projekty Škoda
- II. Vysvětlení témat semestrálního projektu
- III. Analýza dat
- IV. Analýza dat
- V. Plant Simulation
- VI. Plant Simulation, pojmový model
- VII. Struktura simulačního modelu**
- VIII. Tvorba simulačního modelu
- IX. Stochastické procesy a jejich zohlednění v modelu, optimalizace na bázi simulace
- X. Simulační experimentování
- XI. Důsledky na reálný systém, Process Designer
- XII. Rozhraní (ProcessDesigner, MALAGA, TriCAD)





7. přednáška - Struktura simulačního modelu

Cíl přednášky

- › Seznámit posluchače s komponenty diskrétní simulace z obecného hlediska.
- › Seznámit posluchače s různými hledisky dekompozice systému a jemu odpovídajícího simulačního modelu.
- › Na příkladu demonstrujícím standardizaci tvorby simulačních modelů vysvětlit následující pojmy a principy:
 - › Dílčí modely a jejich vzájemné propojení.
 - › Centrální parametrizace.
 - › Vkládání instancí tříd prvků do simulačních modelů.
 - › Uživatelská a expertní úroveň.
- › Seznámit posluchače se zásadami tvorby simulačních modelů.





7. přednáška - Struktura simulačního modelu

Obsah přednášky

- › Komponenty diskrétní simulace z obecného hlediska.
- › Možná hlediska dekompozice systému a simulačního modelu.
- › Dílčí modely a jejich vzájemné propojení.
- › Centrální parametrizace.
- › Vkládání instancí tříd prvků do simulačních modelů.
- › Uživatelská vs. expertní úroveň.
- › Zásady tvorby simulačních modelů.





7. přednáška - Struktura simulačního modelu

Komponenty diskrétní simulace z obecného hlediska

- › Teorie diskrétní simulace pracuje s následujícími pojmy:
 - › **System** \equiv „účelově definovaná množina prvků a vazeb mezi nimi, jež vykazují jako celek určité vlastnosti, resp. chování“.
 - › **Prvek** \equiv „objekt, který nelze dále dělit na jednotlivé části“.
 - › **Model** \equiv „zjednodušený obraz systému“.
 - › **Stav systému** \equiv „množina proměnných a jejich hodnot potřebných k jednoznačnému popisu systému v daný okamžik“.
 - › **Entita** \equiv „objekt v systému, který je nutné explicitně reprezentovat v modelu“
(např. díl, zakázka, dopravní prostředek,...).
 - › **Atribut** \equiv „vlastnost dané entity (např. prioritizace čekajících zákazníků)“.
 - › **Parametr** \sim atribut, jehož hodnoty lze (uživatelsky) nastavit.
 - › **Aktivita** \equiv „časové trvání specifikované délky se známým počátkem“
(např. výrobní čas) – může být určeno také jako stochastická veličina“.





7. přednáška - Struktura simulačního modelu

Komponenty diskrétní simulace z obecného hlediska

- › Teorie diskrétní simulace pracuje s následujícími pojmy:
 - › **Událost** \equiv „okamžité nastoupení změny některých/všech stavových proměnných, kterým dochází ke změně stavu systému“.
 - › **Hlášení události** \equiv „záznam události, která nastane v současný či budoucí simulační čas doplněná souvisejícími daty nutnými pro vykonání této události – minimálně typ události a čas události“.
 - › **Seznam událostí** \equiv „seznam hlášení událostí seřazený vzestupně podle času události“.





7. přednáška - Struktura simulačního modelu

Komponenty diskrétní simulace z obecného hlediska

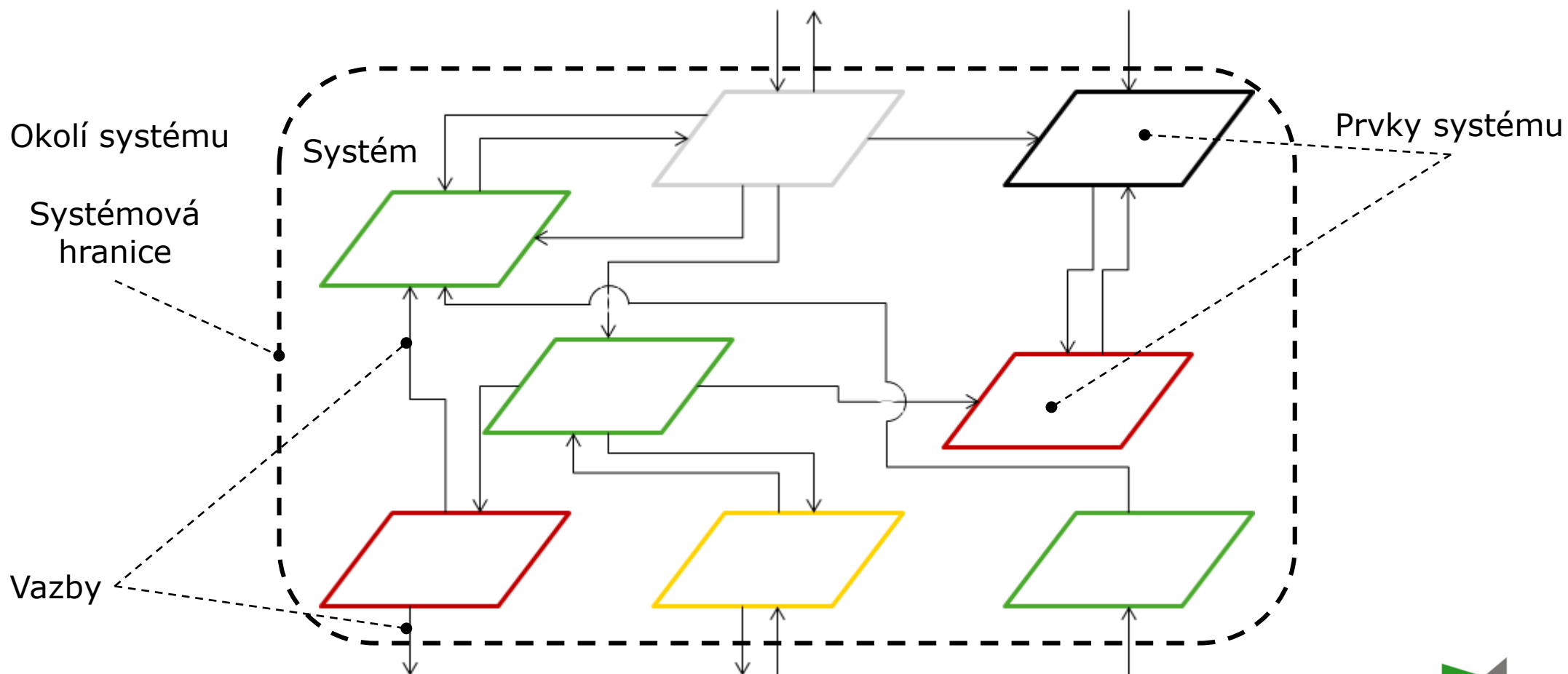
- › Systém lze charakterizovat hodnotami tzv. **stavových proměnných**.
 - › Hodnoty všech stavových proměnných v daný okamžik určují **stav celého systému**.
 - › **Dynamický systém** se v čase mění – mění se v čase hodnoty (některých/všech) jeho stavových proměnných (opakem je **statický systém**).
 - › **Diskrétní simulace** \equiv uvažuje takový systém, ve kterém ke změnám (některých/všech) stavových proměnných dochází skokově v oddělených (diskrétních) časových okamžicích.
 - › Diskrétní simulace se typicky zabývá systémy, jejichž (některé/všechny) stavové proměnné vykazují následující vlastnosti:
 - › **Stochastické (náhodné)** – hodnota proměnných je výsledkem náhodného pokusu.
 - › **Dynamické** – hodnoty stavových proměnných se v čase mění.
 - › **Diskrétní změny** – hodnoty stavových proměnných se mění v diskrétní časové okamžiky.
- Pozor!** Samotné hodnoty proměnných nemusí být diskrétní – může jít o atributy vyjádřené reálnými hodnotami (rychlost, hmotnost,...) ze spojitých intervalů, kvalitativní atributy (barva,...) apod.





7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

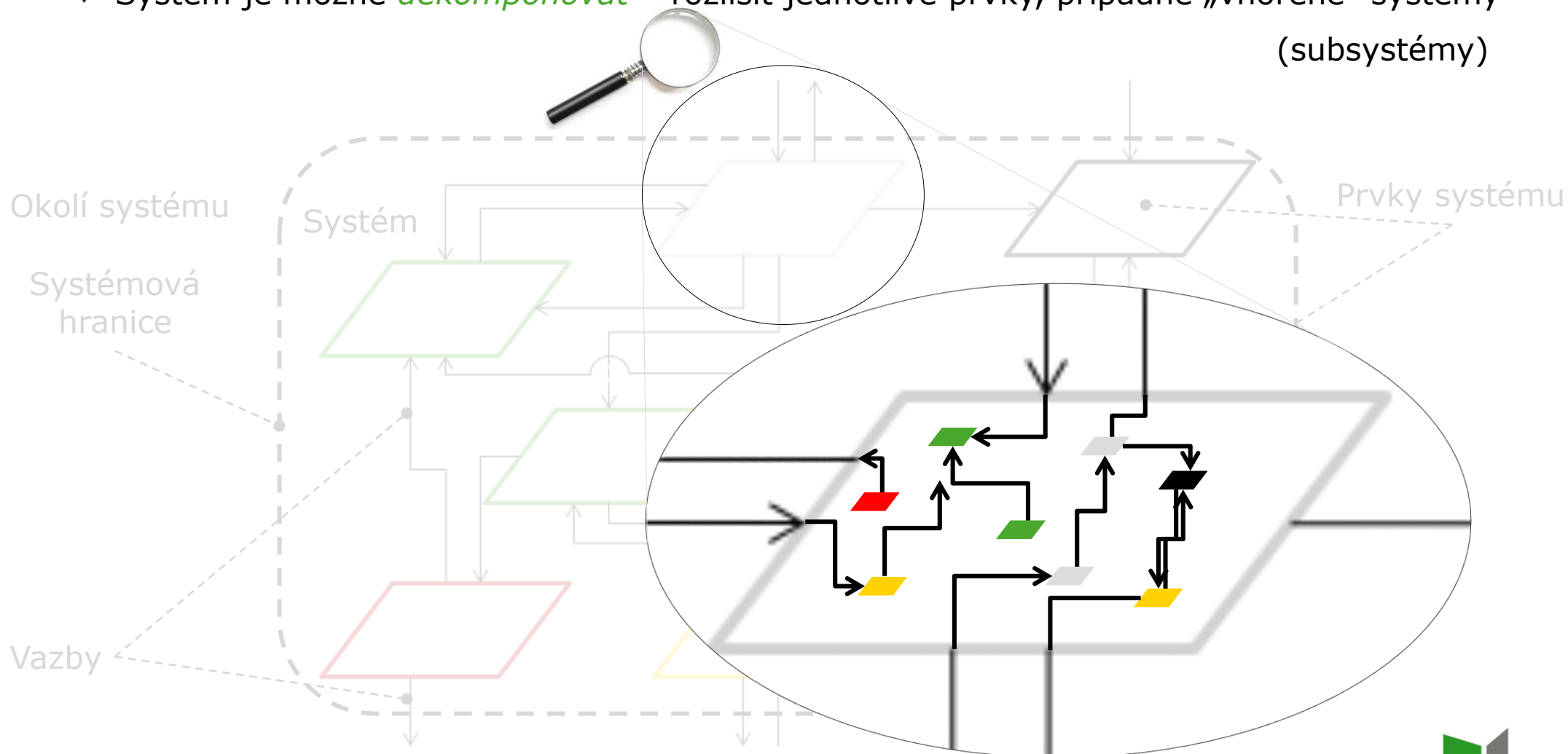




7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

- › System je možné *dekomponovat* – rozlišit jednotlivé prvky, případně „vnořené“ systémy (subsystémy)

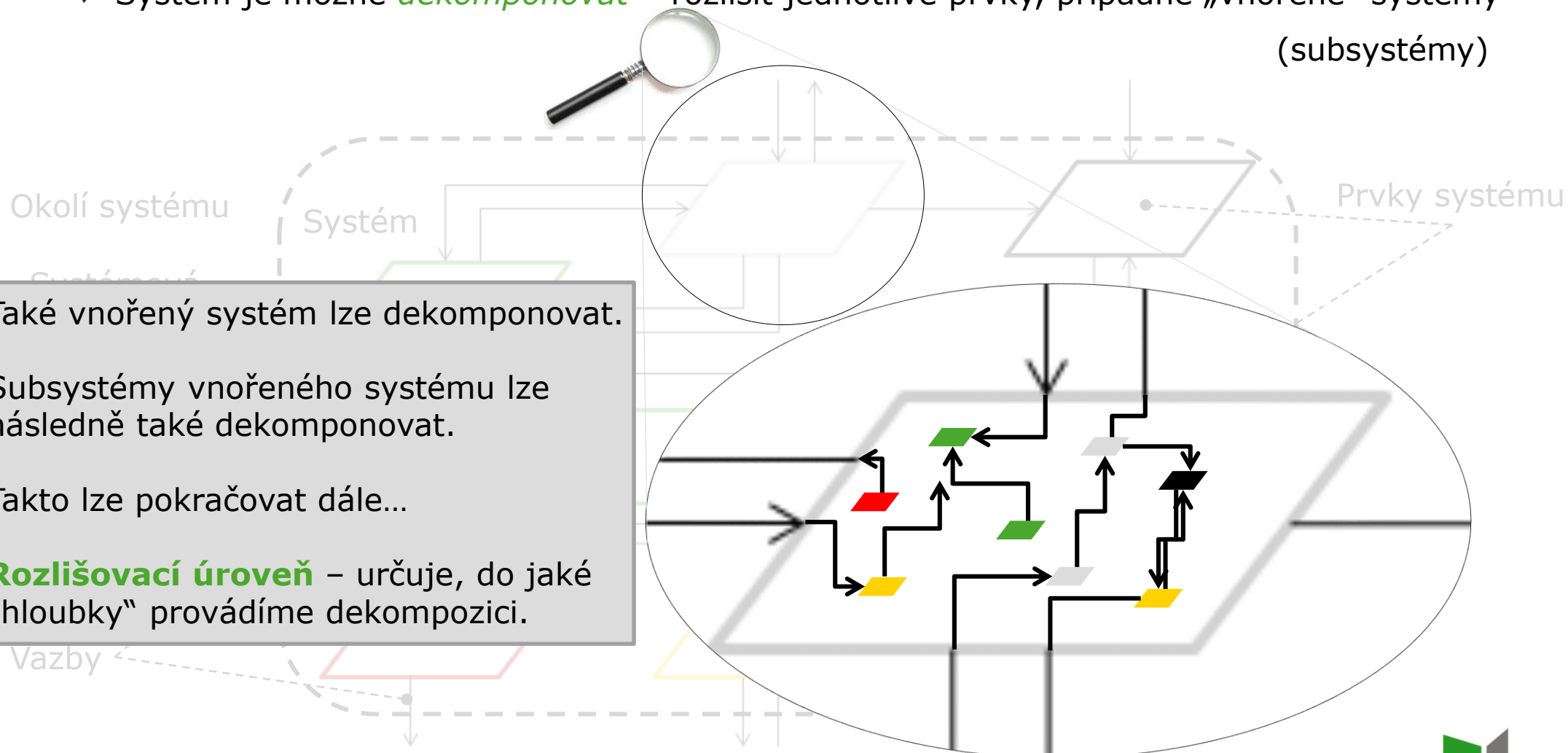




7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

- › System je možné *dekomponovat* – rozlišit jednotlivé prvky, případně „vnořené“ systémy (subsystémy)



Také vnořený systém lze dekomponovat.

Subsystémy vnořeného systému lze následně také dekomponovat.

Takto lze pokračovat dále...

Rozlišovací úroveň – určuje, do jaké „hloubky“ provádíme dekompozici.





7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

› **Povaha vazeb mezi prvky**

- › Materiálový tok, informační tok, ...
- › Telefonní síť, internet, LAN, telegraf, ...
- › ...

→ vaše vlastní příklady?

› **Topologická (geometrická)**

- › Přízemí, 1. patro, 2. patro, ...
- › Sektor „A“, Sektor „B“, ...
- › ...

→ vaše vlastní příklady?

› **Funkcionální**

- › Zásobování, skladování, manipulace, zpracování, balení, expedice, ...
- › Útok, záloha, obrana, ...
- › ...

→ vaše vlastní příklady?

› **Vlastní**

- › Dle specifických potřeb projektu → vaše vlastní příklady?



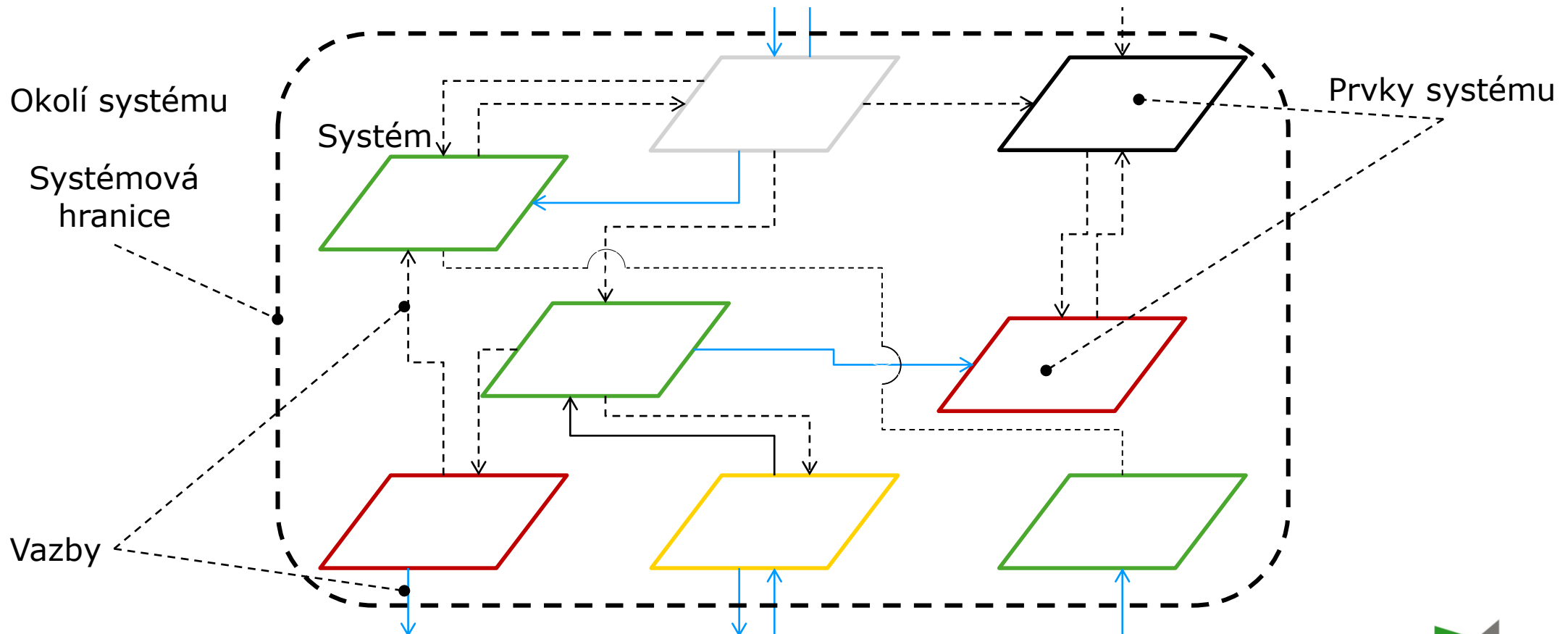


7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

› Povaha vazeb mezi prvky

- > Informační tok ...včetně dotčených prvků!
 —————> Materiálový tok dekompozice by měla vést k rozpadu na disjunktní množiny!

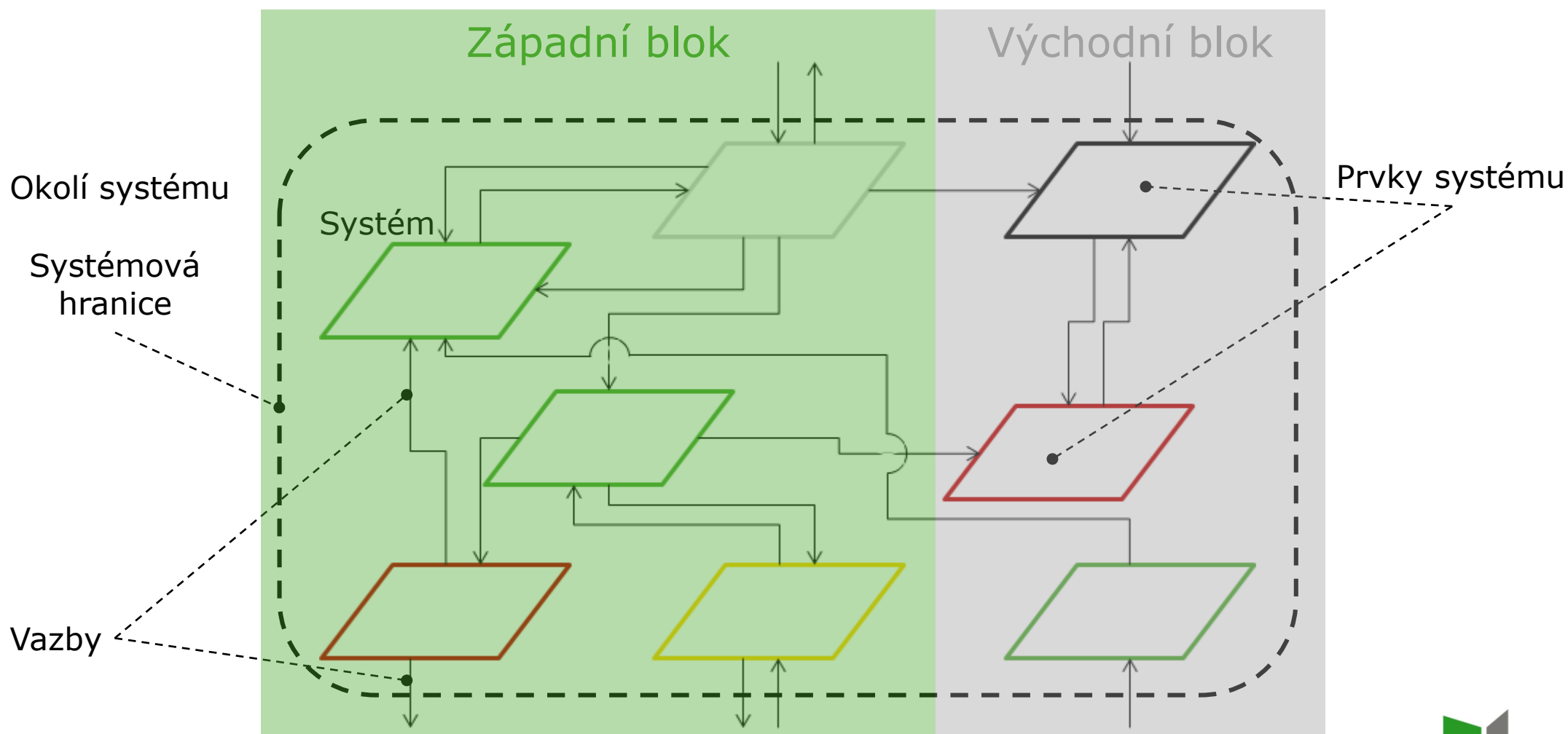




7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

- › **Topologická (geometrická)** dekompozice by měla vést k rozpadu na disjunktní množiny!



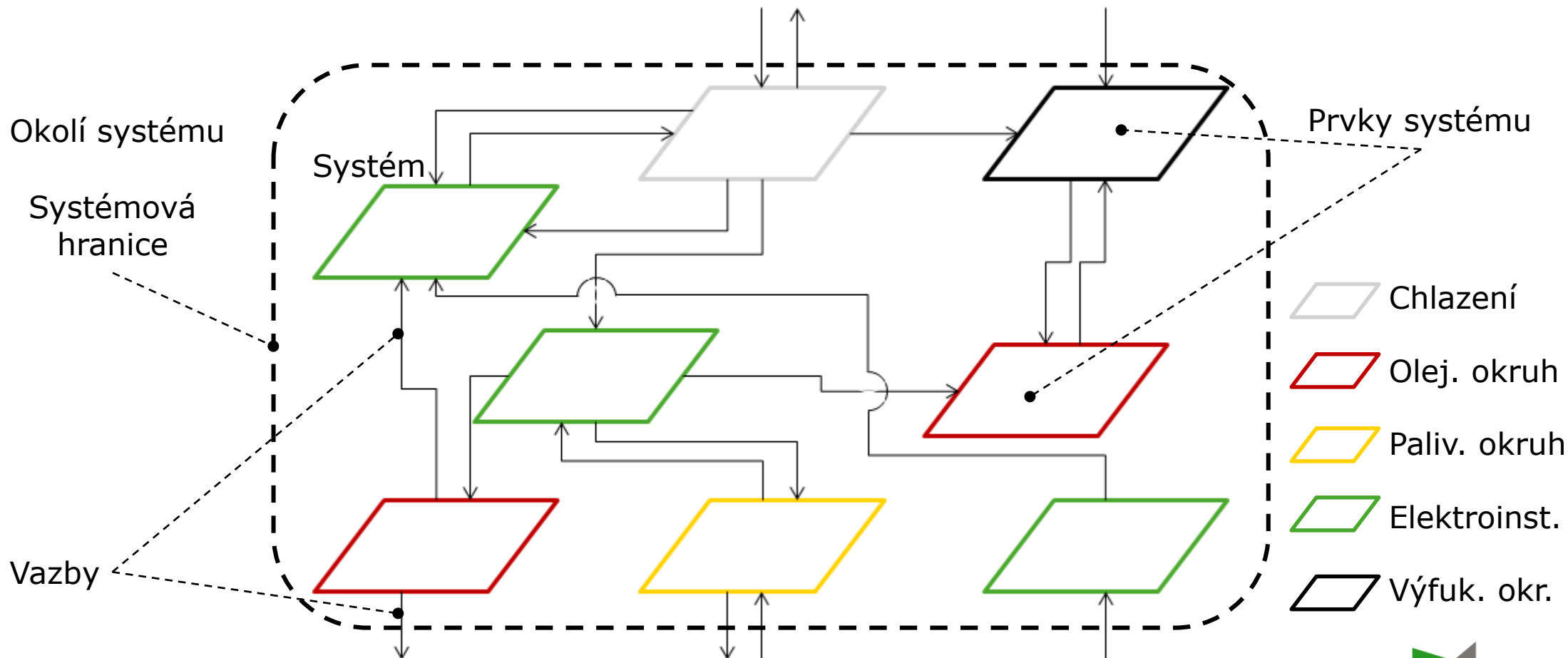


7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

› Funkcionální

dekompozice by měla vést k rozpadu na disjunktní množiny!




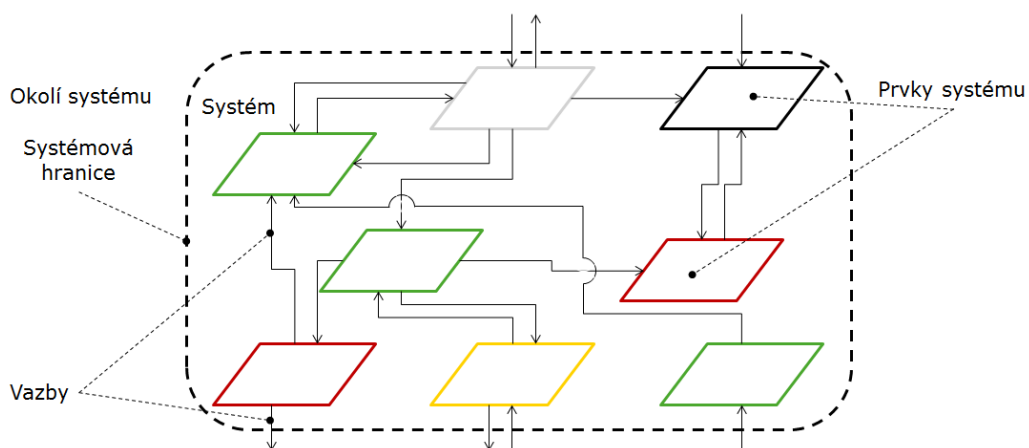


7. přednáška - Struktura simulačního modelu

Možná hlediska dekompozice systému a simulačního modelu

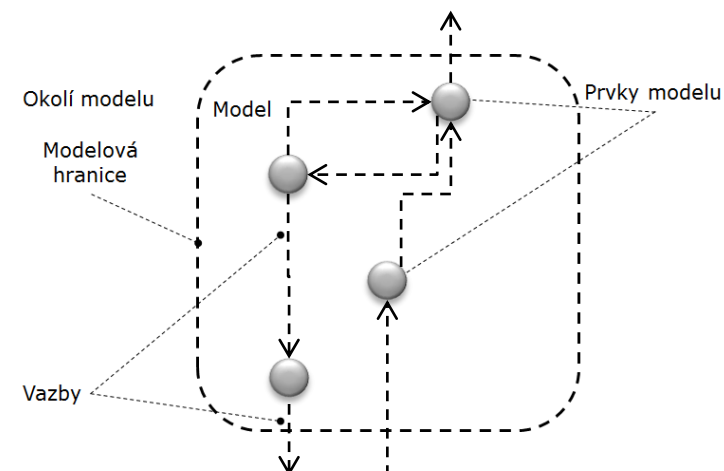
- › **Model** \equiv „zjednodušený obraz systému“.
- › **Prvky modelu** \sim prvky systému (včetně konceptu dekompozice prvků).
...dále nedekomponovatelné (na dané rozlišovací úrovni).

Systém  **Model**
zjednodušení, abstrakce, analogie,...



současná/budoucí realita

\sim



zobrazení reality





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Při vytváření simulačních modelů (výrobních/logistických/dopravních...) systémů je vhodné postupovat v následujících krocích:
 1. Stanovení **hranic systému** ~ ty určí **hranice modelu**.
 2. Vytvoření přehledu (podstatných) **prvků systémů** ~ přehled **prvků modelu**.
přehled vytvoříme provedením (myšlenkové) **dekompozice systému**
 3. Vytvoření popisu **vazeb mezi prvky systému** ~ popis **vazeb mezi prvky modelu**.
 4. Postupné modelování – obvykle v těchto krocích:
 - 1) Jednotlivé prvky.
 - 2) Vazby mezi prvky + naprogramování chování prvků.
 - 3) Propojení (sloučení) prvků v submodely + naprogramování chování submodelů.
 - 4) Vazby mezi submodely.
 - 5) Propojení (sloučení) submodelů ve výsledný simulační model + naprogramování chování.
 - 6) Parametrizace modelu.





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

› Při vytváření simulačních modelů (výrobních/logistických/dopravních...) systémů je třeba postupovat v následujících krocích:

1. Stanovení **hranic systému** ~ ty určí **hranice modelu**.
2. Vytvoření přehledu (podstatných) **prvků systémů** ~ přehled **entit modelu**.
přehled vytvoříme provedením (myšlenkové) **dekompozice systému**.
3. Vytvoření popisu **vazeb mezi prvky systému** ~ popis **vazeb mezi entitami modelu**.

Kroky 1.-3. jsou předmětem **systemové analýzy**, jejímž cílem (výstupem) je **pojmový model**. Pojmový model specifikuje systém pomocí verbálního a/nebo matematického aparátu:

- c) popisný (vysvětlující) text,
- d) vývojový diagram,
- e) Sankeyův diagram, špagetový diagram, Ganttův diagramm, aj.,
- f) číselné údaje.



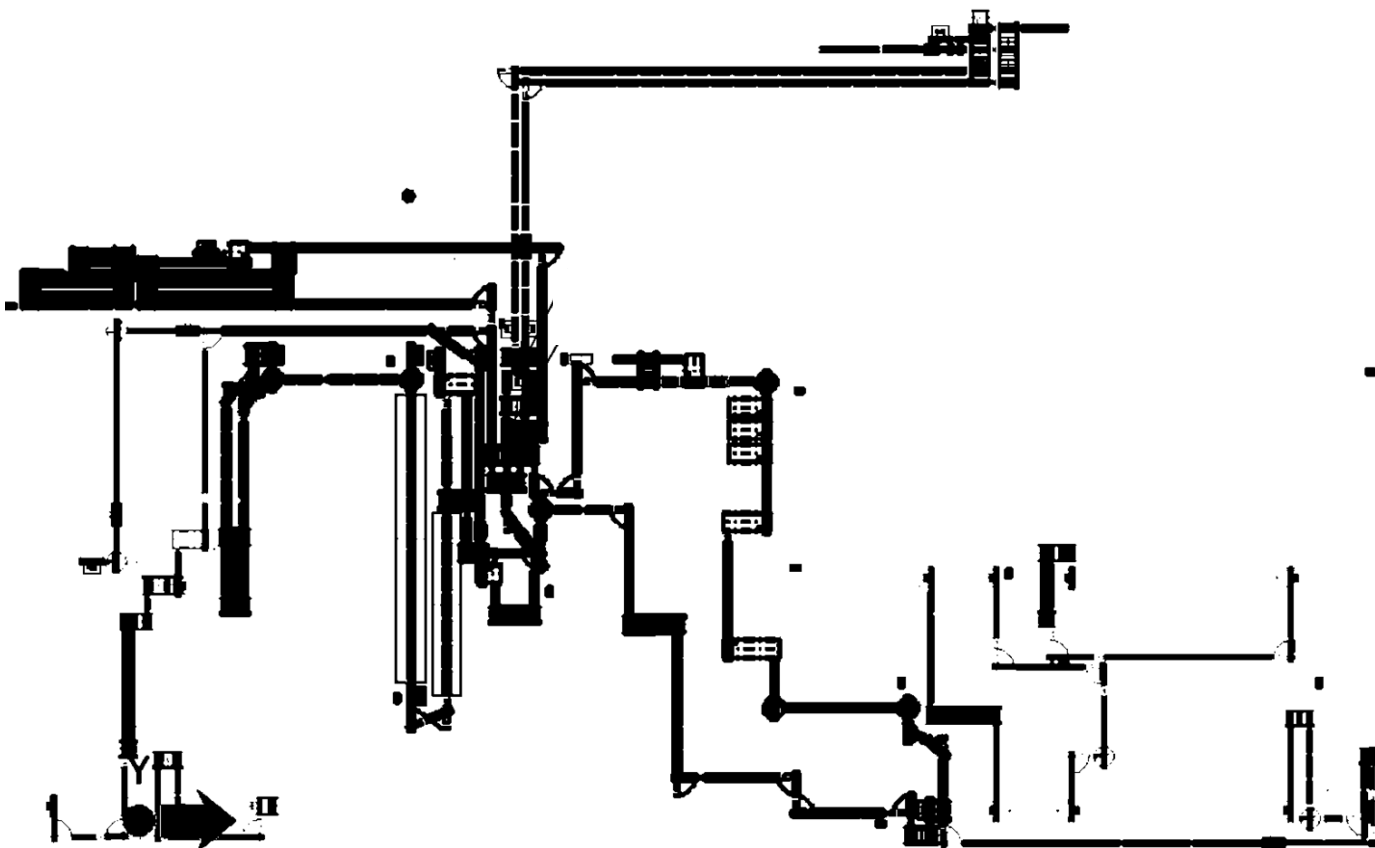


7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Výše popsaný postup tvorby simulačních modelů bude vysvětlen na příkladu simulační studie dopravníkového systému (viz layout) – primární úkol:

Vytvořte simulační model tohoto systému.





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

› Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.

› **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – **parametrů.**
4. Modelování prvků.
5. Sestavení submodelů – obrazů subsystémů pomocí prvků.
6. Propojení submodelů v celkový model – obraz celého systému.
7. Parametrizace jednotlivých prvků.
8. Naprogramování chování modelu (shoda se skutečností/plánem).

Systemová analýza

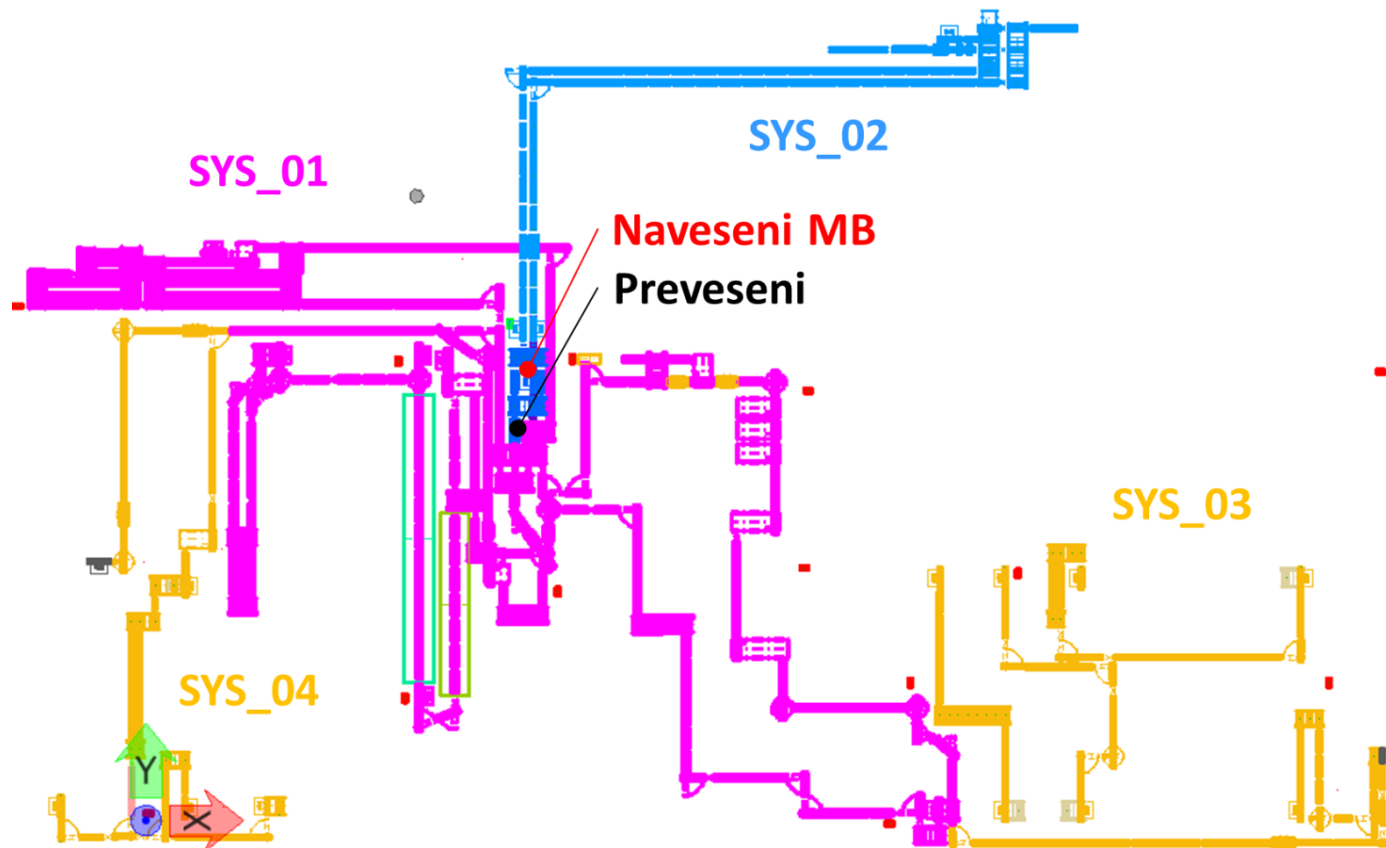




7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**
 1. Dekompozice systému na dílčí subsystémy.

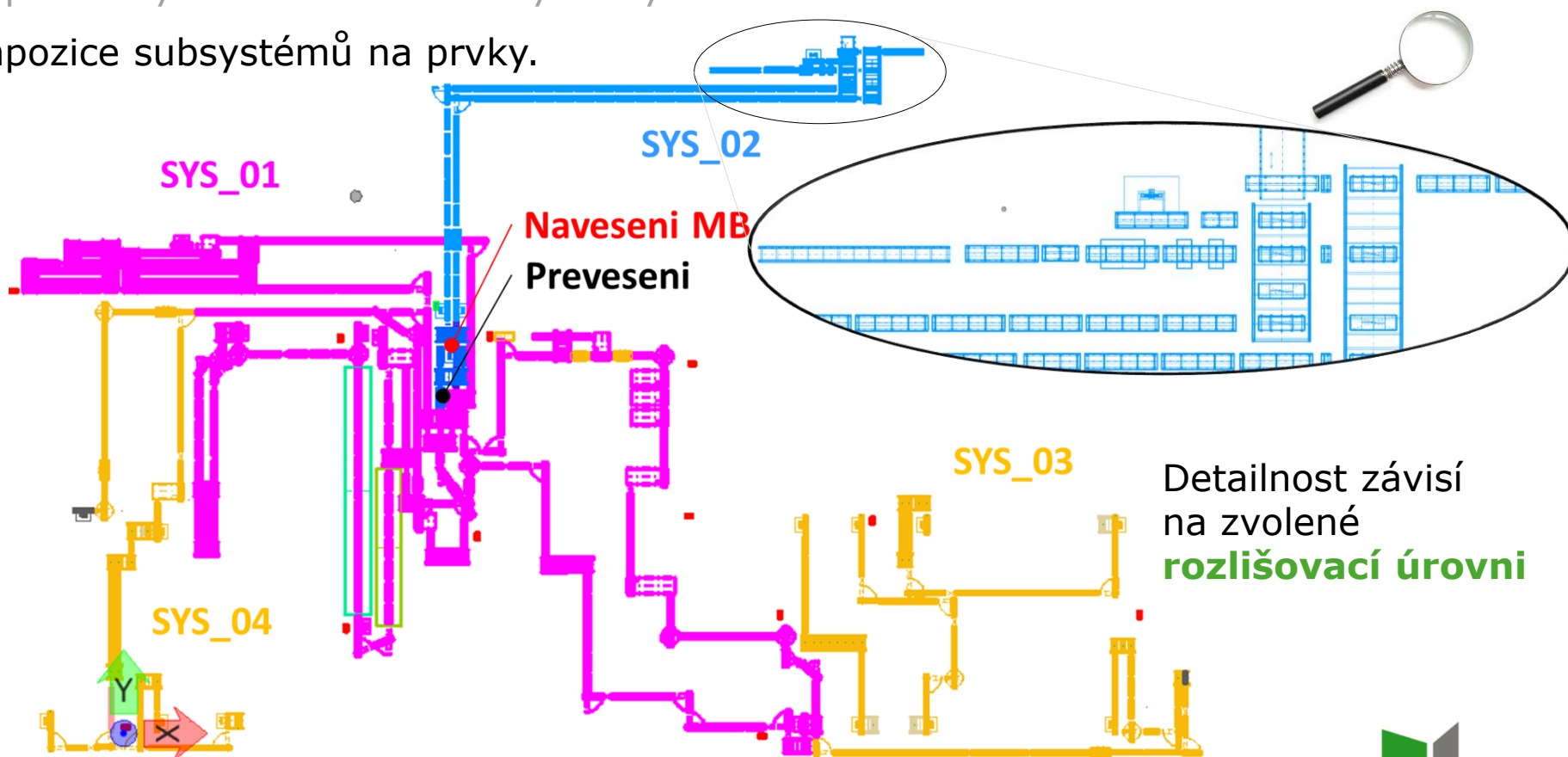




7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**
 1. Dekompozice systému na dílčí subsystémy.
 2. Dekompozice subsystémů na prvky.





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – **parametrů.**

Systemová analýza

- › Vlastnosti těchto prvků analyzujeme (včetně hodnot!).
 - › Vytvoříme (simulační) *modely těchto prvků.*
 - › Pomocí modelů těchto prvků vytvoříme *dílčí modely.*
 - › Dílčí (simulační) modely propojíme v (simulační) *model celku.*
 - › Model celku (reálného systému) *parametrizujeme.*
 - › Naprogramujeme plánované (současné) *chování systému.*
- ...*kvalitativní vs. kvantitativní*





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – **parametrů.**

Systemová analýza

- › Vlastnosti těchto prvků analyzujeme (včetně hodnot!)

...kvalitativní vs. kvantitativní

Příklad: pro dopravníkovou techniku jsou typickými parametry:

- Rozměry (délka, šířka, výška) [m]
- Rychlosti (podélný posuv, příčný posuv, zdvih, pokles,...) [$m \cdot s^{-1}$]
- Zrychlení (analogicky) [$m \cdot s^{-2}$] ...někdy zanedbáváme
- Kapacita [-] ...max. počet elementů mat. toku v/na prvku
- Způsob manipulace ...současný nájezd/výjezd, shuttle,...





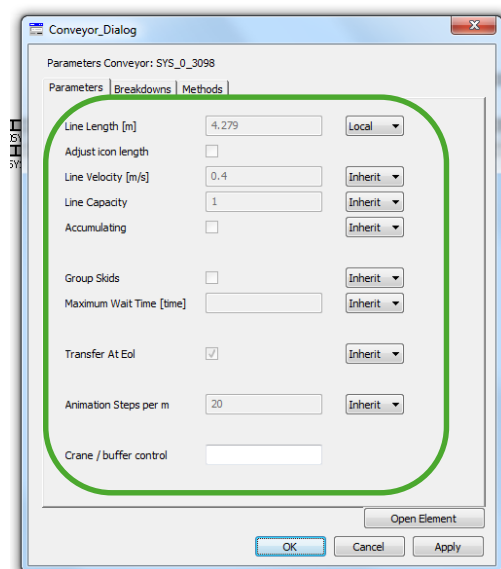
7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – **parametrů.**

Systemová analýza



- Rozměry [m]
- Rychlosti [$\text{m}\cdot\text{s}^{-1}$]
- Zrychlení [$\text{m}\cdot\text{s}^{-2}$]
- Kapacita [-]
- Způsob manipulace
- další...





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – parametrů.
4. Modelování prvků.

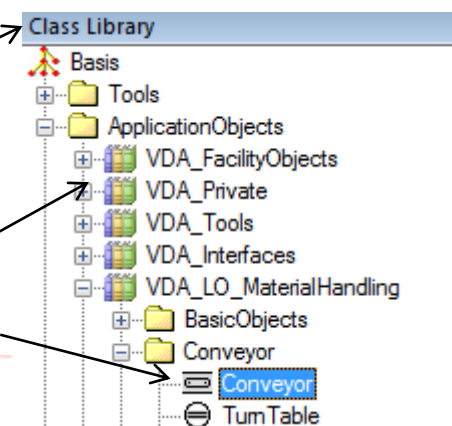
Simulační SW obvykle obsahuje tzv. **knihovny prvků**, příp. umožňuje prvky načíst (databáze, www,...) nebo vytvářet vlastní prvky.

Prvky mají řadu **atributů** (viz výše). Uživatelsky nastavitelné atributy jsou **parametry**.

Parametrizací je ovlivněno chování prvků, resp. chování celého simulačního modelu!

Příklad: Plant Simulation

- Knihovna tříd (Class Library)
 - Class Library obsahuje knihovny prvků
 - Knihovny obsahují prvky (třídy prvků)





7. přednáška - Struktura simulačního modelu

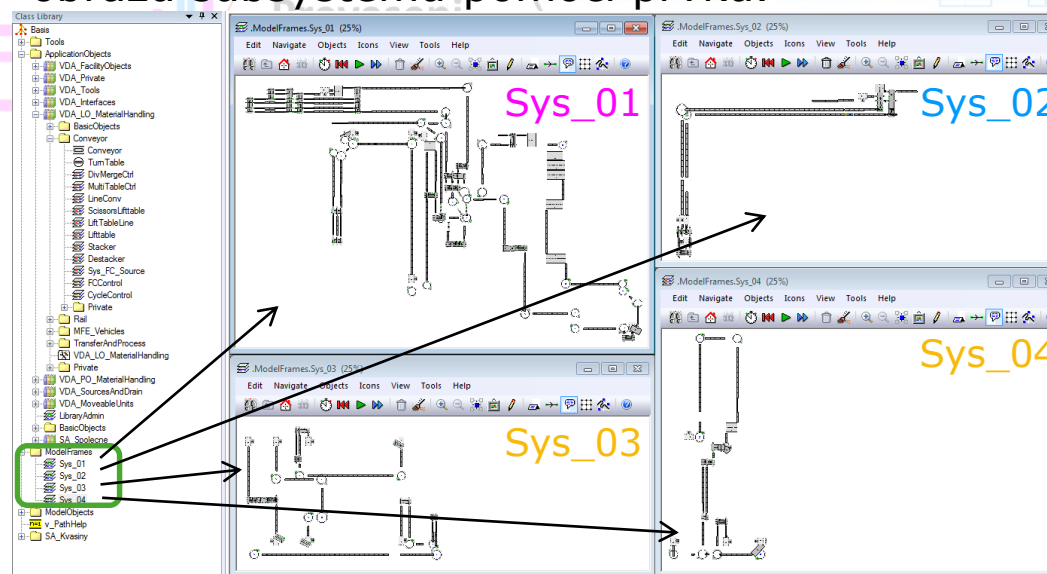
Dílčí modely a jejich vzájemné propojení

› Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.

› **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – parametrů.
4. Modelování prvků.
5. Sestavení submodelů – obrazů subsystémů pomocí prvků.

› Pomocí modelů prvků vytvoříme *dílčí (simulační) modely*



Při průchodu elementů materiálového toku více submodely je třeba zajistit, aby ve všech submodelech byly atributy těchto elementů platné!



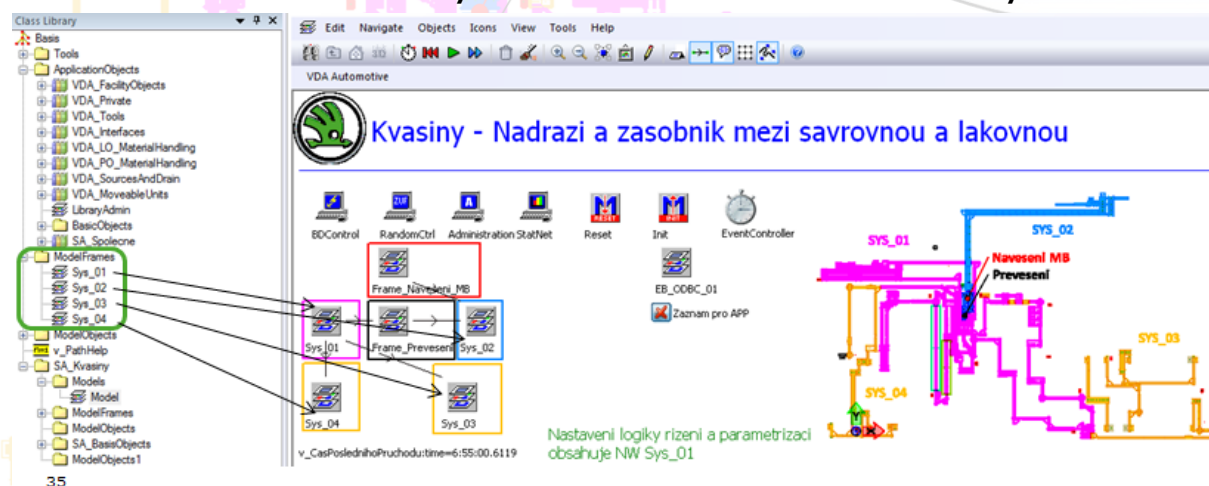


7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – parametrů.
4. Modelování prvků.
5. Sestavení submodelů – obrazů subsystémů pomocí prvků.
6. Propojení submodelů v celkový model – model celého systému.



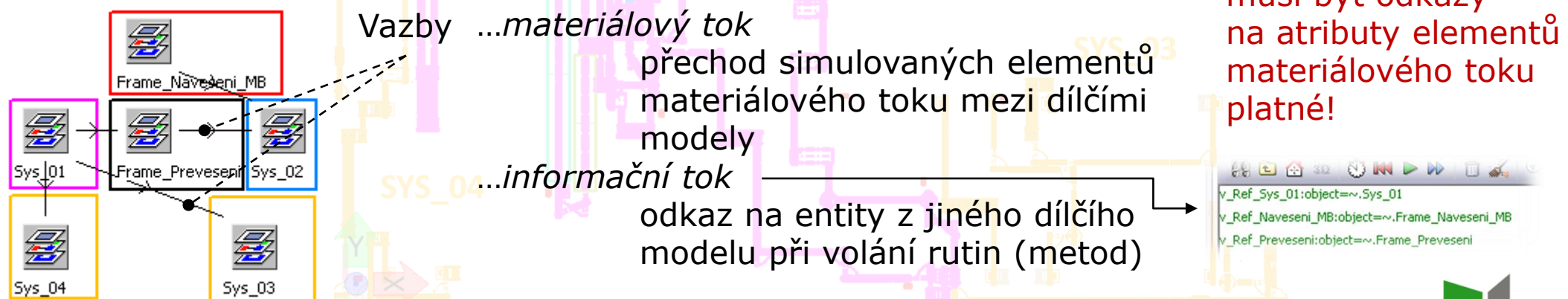


7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**

1. Dekompozice systému na dílčí subsystémy.
2. Dekompozice subsystémů na prvky.
3. Analýza vlastností prvků – parametrů.
4. Modelování prvků.
5. Sestavení submodelů – obrazů subsystémů pomocí prvků.
6. Propojení submodelů v celkový model – model celého systému.





7. přednáška - Struktura simulačního modelu

Dílčí modely a jejich vzájemné propojení

- › Reálný (navržený) systém – dopravníková technika propojující několik výrobních oblastí.
- › **Řešení úkolu „vytvořte simulační model tohoto systému!“**
 1. Dekompozice systému na dílčí subsystémy.
 2. Dekompozice subsystémů na prvky.
 3. Analýza vlastností prvků – parametrů.
 4. Modelování prvků.
 5. Sestavení submodelů – obrazů subsystémů pomocí prvků.
 6. Propojení submodelů v celkový model – model celého systému.
 7. Parametrizace jednotlivých prvků.
- › Lze rozlišit parametrizaci.
 - › **Lokální** – nastavení hodnot parametrů jednotlivých prvků.
 - › **Centrální** – využití agregovaných přehledů a metod (rutin) pro načtení a zápis parametrů (včetně možnosti kontroly konzistence).

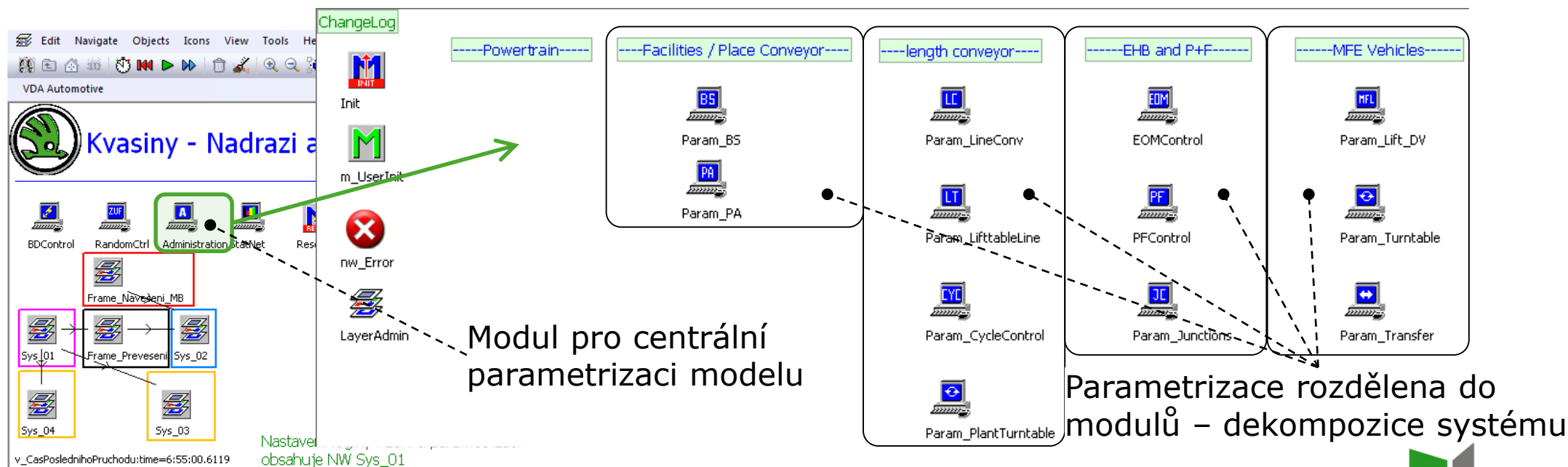




7. přednáška - Struktura simulačního modelu

Centrální parametrizace

- › Každý prvek modelu je třeba parametrizovat – nastavit hodnoty jeho atributů.
- › Výhody a nevýhody výše uvedených způsobů parametrizace.
 - › **Lokální** – neefektivní, nepřehledné při vyšším počtu elementů modelu a složité hierarchii.
 - › **Centrální** – přehlednější, zejména s využitím rutin pro načtení/zápis hodnot snadné (včetně možnosti vzájemné konzistence).





7. přednáška - Struktura simulačního modelu

Centrální parametrizace

- › Každý prvek modelu je třeba parametrizovat – nastavit hodnoty jeho atributů.
- › Výhody a nevýhody výše uvedených způsobů parametrizace.
 - › **Lokální** – neefektivní, nepřehledné při vyšším počtu elementů modelu a složité hierarchii.
 - › **Centrální** – přehlednější, zejména s využitím rutin pro načtení/zápis hodnot snadné (včetně možnosti vzájemné konzistence).

Online Help Automotive Library

Parameter Read_Parameter Write_Parameter

ChangeLog

Init

m_UserInit

nw_Error

LayerAdmin

Powertrain

Facilities / Place Conveyor

length conveyor

Param_BS

Param_PA

Param_LineConv

Param_LifttableLine

Param_CycleControl

Param_PlantTurntable

Each of the „modules“ for central parameterization of model elements contains (besides possible additional elements):

- Table with overview of all parameters
- Routine (method) for loading parameters of model elements into the table
- Routine (method) for saving (modified) parameters from the table into the elements in the model





7. přednáška - Struktura simulačního modelu

Centrální parametrizace

- › Každý prvek modelu je třeba parametrizovat – nastavit hodnoty jeho atributů.
- › Výhody a nevýhody výše uvedených způsobů parametrizace.
 - › **Lokální** – neefektivní, nepřehledné při vyšším počtu elementů modelu a složité hierarchii.
 - › **Centrální** – přehlednější, zejména s využitím rutin pro načtení/zápis hodnot snadné (včetně možnosti vzájemné konzistence).

The screenshot displays a simulation software interface with a central parameter table. The table is titled "Param_PlatTurntable" and lists various conveyor and lifttable objects with their parameters. A green box highlights the "Parameter" icon in the top right, and a green arrow points from it to the table. Another green box highlights the "Param_LineConv" icon in the top center, and a green arrow points from it to the table. The table contains the following data:

object	string	length[m]	speed[m/s]	integer	boolean	boolean	time	boolean	time	string	boolean	integer	integer	integer	integer	
string	Object	Type	Length	Velocity	Capacity	Accumulating	MaxWaitTime	TransATEOL	LiftingTime	DefaultTablePos	Statistics	Stack	Height	StackingTime	DestackingTime	AnimationSteps
1	~.~.Frame_Naveseni_MB.Conv	Conveyor	10	0.4	1	false	0.0000	true			false					20
2	~.~.Frame_Naveseni_MB.Conv	Conveyor	10	0.4	1	false	0.0000	true			false					20
3	~.~.Frame_Preveseni.SYS_0_P	Conveyor	4.9	0.4	1	false	0.0000	true			false					20
4	~.~.Frame_Preveseni.SYS_0_P	Conveyor	4.9	0.4	1	false	0.0000	true			false					20
5	~.~.Sys_04.SYS_0_260	Conveyor	3.318	0.6	1	false	0.0000	true			false					20
6	~.~.Sys_04.SYS_0_2741	Conveyor	3.314	0.6	1	false	0.0000	true			false					20
7	~.~.Sys_04.SYS_0_2742	Conveyor	7.379	0.6	1	false	0.0000	true			false					20
8	~.~.Sys_04.SYS_0_2743	ScissorLifttable	3.6	0.6	1	false	0.0000		15.0000	Entrance	false					20
9	~.~.Sys_04.SYS_0_411	Conveyor	5.429	0.6	1	false	0.0000	true			false					20
10	~.~.Sys_04.SYS_0_412	Conveyor	5.4	0.6	1	false	0.0000	true			false					20

The interface also shows various icons for model components like "Param_BS", "EOMControl", "Param_Lift_DV", and "Param_LineConv". A text box on the right states: "Tabulka centrální parametrizace obsahuje přehled všech prvků dané funkce v modelu a jejich parametrů".





7. přednáška - Struktura simulačního modelu

Centrální parametrizace

- › Každý prvek modelu je třeba parametrizovat – nastavit hodnoty jeho atributů.
- › Výhody a nevýhody výše uvedených způsobů parametrizace
 - › **Lokální** – neefektivní, nepřehledné při vyšším počtu elementů modelu a složité hierarchii.
 - › **Centrální** – přehlednější, zejména s využitím rutin pro načtení/zápis hodnot snadné (včetně možnosti vzájemné konzistence).

object	string	length[m]	speed[m/s]	integer	boolean	boolean	time	boolean
0	1	2	3	4	5	6	7	8
string	Object	Type	Length	Velocity	Capacity	Accumulating	GroupSkids	MaxWaitTime
1	~.~.~.Frame_Naveseni_MB.Conv	Conveyor	10	0.4	1	False	False	0.0000
2	~.~.~.Frame_Naveseni_MB.Conv	Conveyor	10	0.4	1	False	False	0.0000
3	~.~.~.Frame_Preveseni.SYS_0_P	Conveyor	4.9	0.4	1	False	False	0.0000
4	~.~.~.Frame_Preveseni.SYS_0_P	Conveyor	4.9	0.4	1	False	False	0.0000
5	~.~.~.Sys_04.SYS_0_260	Conveyor	3.318	0.6	1	False	False	0.0000
6	~.~.~.Sys_04.SYS_0_2741	Conveyor	3.314	0.6	1	False	False	0.0000
7	~.~.~.Sys_04.SYS_0_2742	Conveyor	7.379	0.6	1	False	False	0.0000
8	~.~.~.Sys_04.SYS_0_2743	ScissorLifttable	3.6	0.6	1	False	False	0.0000
9	~.~.~.Sys_04.SYS_0_411	Conveyor	5.429	0.6	1	False	False	0.0000
10	~.~.~.Sys_04.SYS_0_412	Conveyor	5.4	0.6	1	False	False	0.0000

Sys_04



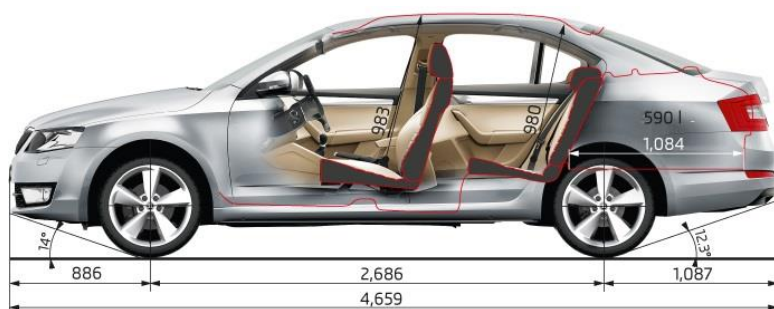


7. přednáška - Struktura simulačního modelu

Vkládání instancí tříd prvků do simulačních modelů

- › **Třída prvku** \equiv originál nacházející se v knihovně tříd; tento originál lze mnohonásobně vkládat (instanciovat) do modelu – všechny tyto instance budou mít (dědit) vlastnosti (hodnoty atributů) originálu.
- › **Instance prvku** \equiv konkrétní prvek vložený do modelu, přičemž tento prvek má svůj originál, jehož vlastnosti (hodnoty atributů) dědí (hodnoty lze změnit vlastní parametrizací).

Třída



Základní atributy: rozměry, výrobce,...

Parametry: motorizace, typ karoserie, barva, výbava,...

Instance



Základní atributy **zděděny** (stejně hodnoty jako třída)

Některé parametry zděděny (např. typ karoserie), jiné změněny (např. barva)





7. přednáška - Struktura simulačního modelu

Vkládání instancí tříd prvků do simulačních modelů

- › U každého prvku vloženého/vygenerovaného v (sub)modelu, tj. u každé instance lze jednoznačně identifikovat jeho třídu.

The screenshot illustrates the structure of a simulation model. On the left, a 'Class Library' pane shows a hierarchy of classes, with 'Entity' highlighted in green and labeled 'třída'. The main workspace shows a model diagram with elements: 'Source', 'SingleProc_1', 'SingleProc_2', and 'Entity: .MUs.Entity:80173 on SingleProc_2'. A callout box points to the 'Entity' element, labeled 'instance entity ~ konkrétní element mat. toku'. A 'model' label points to the main workspace area. A configuration dialog for '.Models.Frame.Source' is open, showing various settings. A callout box points to the 'MU' field, which is set to '*.MUs.Entity', labeled 'odkaz na třídu generovaných entit ~ elementů mat. toku'.





7. přednáška - Struktura simulačního modelu

Vkládání instancí tříd prvků do simulačních modelů

- › Instance **dědí hodnoty** atributů své třídy, dědičnost lze zrušit **vlastní parametrizací**.
- › **Příklad 1:** instance entit ~ elementů materiálového toku.

The screenshot illustrates the configuration of simulation model elements. It shows a 'Class Library' on the left with a tree structure including 'Entity' under 'MUs'. The main workspace, labeled 'model', displays a flow diagram with components: Source, SingleProc_1, SingleProc_2, and Drain. Two dialog windows are open:

- Třída (dialogové okno):** This window is for the class '#.MUs.Entity'. It shows the 'Name' as 'Entity' and 'Conveying direction' as '0 (forward)'. Under the 'Attributes' tab, values are: MU length: 0.8 m, MU width: 0.8 m, MU height: 1 m, Booking point length: 0.4 m, Booking point width: 0.4 m, and Booking point height: 0 m.
- Instance (dialogové okno):** This window is for the instance '#.MUs.Entity:80172'. It shows the 'Name' as 'Entity' and 'Number' as '80172'. The 'Label' is 'Zmenena hodnota'. Under the 'Attributes' tab, values are: MU length: 1.6 m, MU width: 0.8 m, MU height: 1 m, Booking point length: 0.4 m, Booking point width: 0.4 m, and Booking point height: 0 m. The 'Label' and 'MU length' fields are highlighted in yellow, indicating they have been customized.





7. přednáška - Struktura simulačního modelu

Vkládání instancí tříd prvků do simulačních modelů

- › Instance **dědí hodnoty** atributů své třídy, dědičnost lze zrušit **vlastní parametrizací**.
- › **Příklad 2:** instance prvků představujících další prvky systému (stroje, dopravníky,...).

The screenshot illustrates the configuration of a simulation model. On the left, a 'Class Library' tree shows the hierarchy: MaterialFlow > SingleProc. The main workspace shows a model diagram with a 'Source' connected to 'SingleProc_1', which is connected to 'SingleProc_2', and finally to a 'Drain'. Below the diagram, two dialog windows are shown:

- Třída (dialogové okno):** This window is for the 'SingleProc' class. It shows various attributes such as 'Processing time' (set to 1:00), 'Set-up time' (0), and 'Recovery time' (0). The 'Planned' checkbox is checked.
- Instance (dialogové okno):** This window is for the 'SingleProc_1' instance. It shows the same attributes but with 'Processing time' set to 2:00. The 'Entrance locked' checkbox is checked, and the 'Planned' checkbox is unchecked.

Arrows indicate the relationship between the class library, the model diagram, and the two dialog windows.

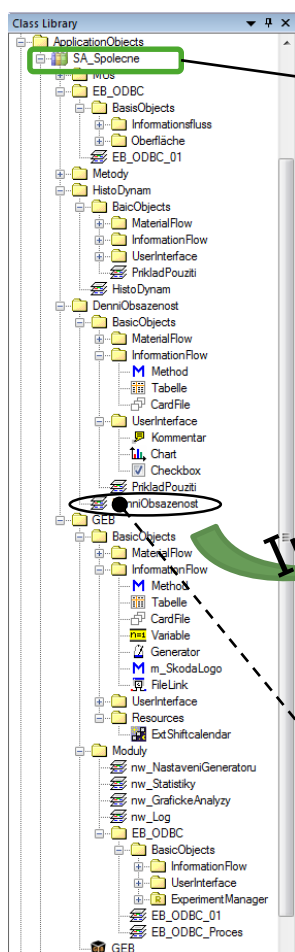




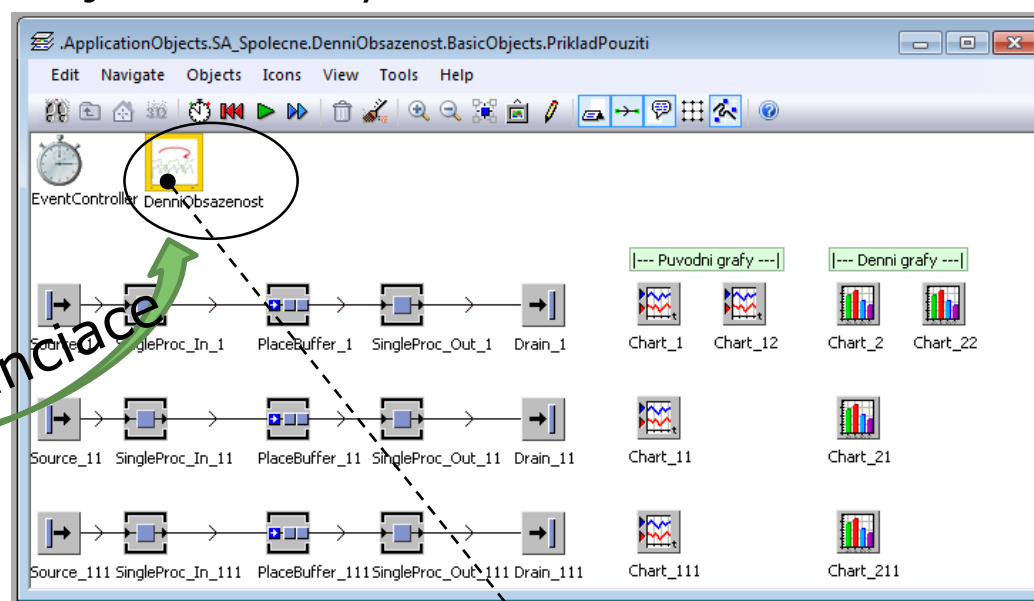
7. přednáška - Struktura simulačního modelu

Uživatelská vs. expertní úroveň

- › Simulační SW Plant Simulation umožňuje vytvářet **vlastní knihovny prvků**.



Příklad: knihovna „SA_Spolecne“ obsahující složky s třídami prvků, které jsou vkládány často do modelů v rámci různých projektů.



Třída
(v knihovně prvků)

Instance
(v modelu)

Model „DenniObsazenost“ lze z knihovny tříd (Class Library) vložit (instanciovat) vícekrát do libovolného simulačního modelu.

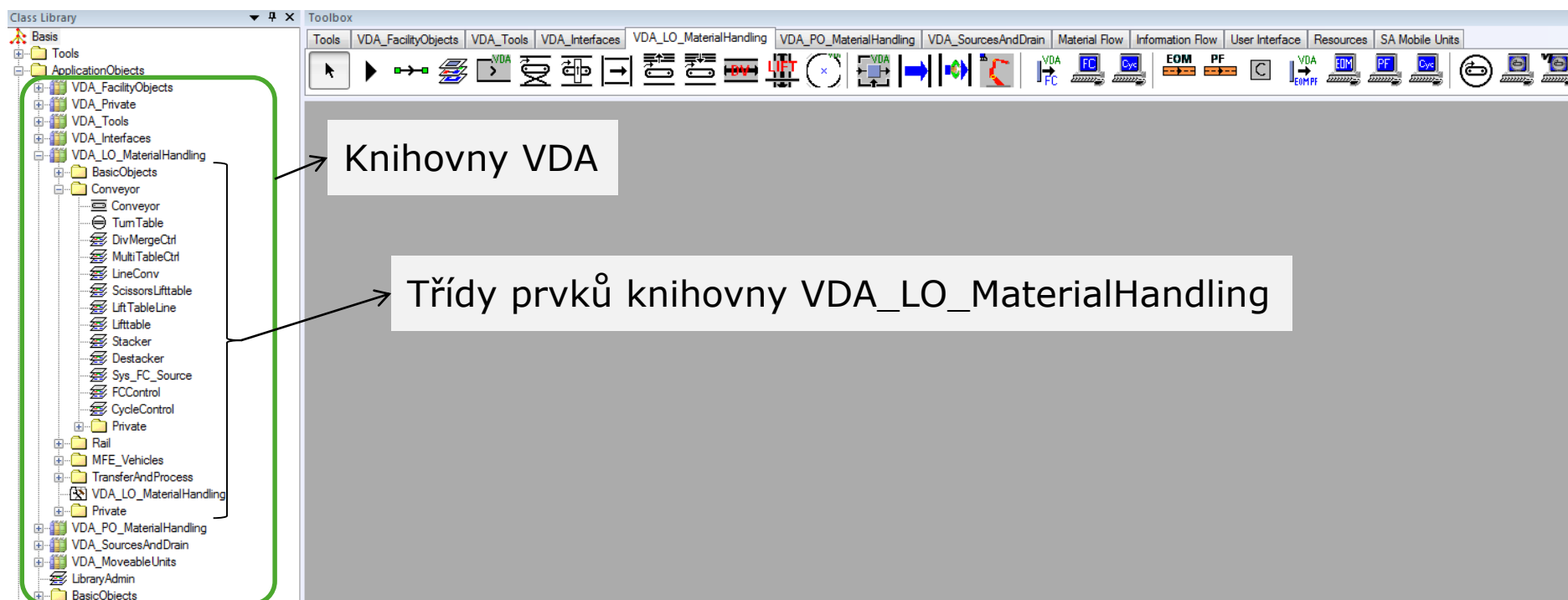




7. přednáška - Struktura simulačního modelu

Uživatelská vs. expertní úroveň

- › V oboru automotive existuje pro tvorbu simulačních modelů **standard „VDA“** – Verband der Automobilindustrie; podporovaný simulační software je Plant Simulation.
- › VDA vyvíjí knihovny obsahující prvky (třídy prvků) – tyto prvky jsou vhodné pro modelování systémů typu lakovna, svařovna, montáž, logistických systémů,...





7. přednáška - Struktura simulačního modelu

Uživatelská vs. expertní úroveň

- › Většina prvků knihoven VDA má složitější strukturu, kterou lze *dekomponovat* na dvě úrovně (subsystémy):
 - › **Uživatelská úroveň** – lokální parametrizace (nastavení *hodnot parametrů* daného prvku).
 - › **Expertní úroveň** – zajišťuje funkci a komunikaci instance v rámci simulačního modelu
...označovaná jako „nw_Private“ je vnořena do (vrstvy) uživatelské úrovně.

Příklad: prvek „ScissorsLifttable“ pro modelování nůžkového zvedáku.

Uživatelská úroveň („station level“)

Expertní úroveň (nw_Private)

Parameter prvku





7. přednáška - Struktura simulačního modelu

Zásady tvorby simulačních modelů

- › **Orientace na účel simulační studie (cíl projektu)** – cílem projektu není vytvořit model.
- › **Model zahrnuje pouze nezbytný detail** – přístup „KISS“ (Keep It Simple, Stupid).
- › **Model je vyvíjen v průběhu času** – metoda postupně se zpřesňujících modelů.
- › **Model popisuje všechny kritické aktivity/události ve vhodném detailu** – model nezahrnuje nepodstatné jevy (procesy z okolí systému).
- › **Model je flexibilní** – model umožňuje pozdější úpravy a snadnou parametrizaci.
- › **Model je robustní** – model je možné využít jako součást (submodel) „většího modelu“, propojit s jiným modelem apod.
- › **Model zřetelně zobrazuje výsledky** – výsledky jsou ucelené, jednoznačné (uvedeny jednotky, okrajové a počáteční podmínky), přehledné.





7. přednáška - Struktura simulačního modelu

Otázky z dané problematiky

- › Jak lze vnímat **system** a co je podstatou **systemového přístupu**?
- › Vysvětlete princip **diskrétní simulace** ve vztahu ke stavům modelovaného systému.
- › Jak rozumíte pojmům **entita**, **událost**, **seznam událostí** v kontextu diskrétních simulací?
- › Jak lze **system** dekomponovat a proč se dekompozice systému provádí?
- › Co je to **prvek** systému a co popisují **atributy entit**?
- › Jaké mohou být **vazby** mezi prvky a mezi subsystemy obecně a v simulačním modelu?
- › Jakým způsobem lze **nastavovat parametry prvků** v simulačním modelu?
- › Jaký je rozdíl mezi **třídou** a **instancí** prvku?
- › Vysvětlete princip **dědičnosti** atributů.





Děkuji.

