

FORMAL DEFINITION OF BUSINESS RULES BY GRAMMAR SYSTEMS

LTA CONFERENCE

BRNO 2014

Author:

Ing. ROMAN HYPSKÝ

Trainer:

doc. RNDr. JITKA KRESLÍKOVÁ, CSc.

Content

2

- Theory
 - Introduction to Business Rules
 - Introduction to Grammar Systems
- Application of business rules by grammar systems
- Summary
- Future development
- Bibliography

Business Rules

3

- As the name suggests business rules is a set of rules that enable and partially automate and accelerate the company's decisions in certain processes.
- It works on the basis of the implication. If is accomplished anything, execute something (IF condition THEN action)
- The basic version has two parts: **Condition** and **Action**
- Usually are described by natural language, therefore we need a unified business dictionary.
- They are usually described using BPEL language in practice

Business Rules

4

- Typical is using in processes that contain easily automated actions that do not depend on human opinion.
- Typical use is for example in the banking sector, or for example for determination of the payroll.
- Examples:
 - If is the bank client more than 5 years, no commitments, and has a steady income greater than CZK 20 000, he can automatically get a loan without collateral to CZK 100 000.
 - If is the total price of the order min. CZK 1000, shipping is free.

Business Rules

5

□ Main advantages:

- Partial automation of processes and saving of relevant resources (time and people).
- It's not necessary to change processes, but we can simply add or modify rules.
- Used business rules can be formalized, and easily implemented using the information system and therefore can be automated.

Grammar Systems

6

- Grammar systems represent a set of grammars working together. If calculation is running on multiple stations, the stations are synchronized using the protocol to produce same language.
- Compilation at multiple locations is implemented for several reasons. For example computational complexity or increase of the generative power.
- There are two basic types of grammar systems:
 - CD grammar systems
 - PC grammar systems

Cooperating Distributed CD Grammar Systems

7

- Sequential grammar systems.
- All parts have a common **sentence form**, where works one of the grammars.
- The transfer of control between grammars is an important element.
- We have standard definition of grammar like
 $G = (N, \Sigma, P, S)$
- CD grammar system is defined as
 $\Gamma = (N, T, S, P_1, \dots, P_n)$ of degree n, where $n \geq 1$
- We can use that in several modes ($t, *, k\text{-step}$, etc.).

PC Grammar Systems

8

- Parallel grammar systems.
- Each grammar has own **sentence form**, so they can work parallel, each on own sentence form.
- There are query symbols for replacing sentence form.
- There is a master component that produces final language.
- $\Gamma = (N, K, T, (S_1, P_1), \dots, (S_n, P_n))$,
where K is a finite set of query symbols $K = \{Q_1, \dots, Q_n\}$.
- Centralized parallel systems are more important for us.

Basic types of Business Rules

9

□ Types of **condition-actions** business rules:

1) <condition>

then

<action 1

action 2

...

action N>

2) <condition>

then

<action 1

action 2

...

action N>

else

<action 1

action 2

...

action N>

3) <condition>

then

<action 1

action 2

...

action N>

else

<condition>

then

<action 1

action 2

...

action N>

else

...

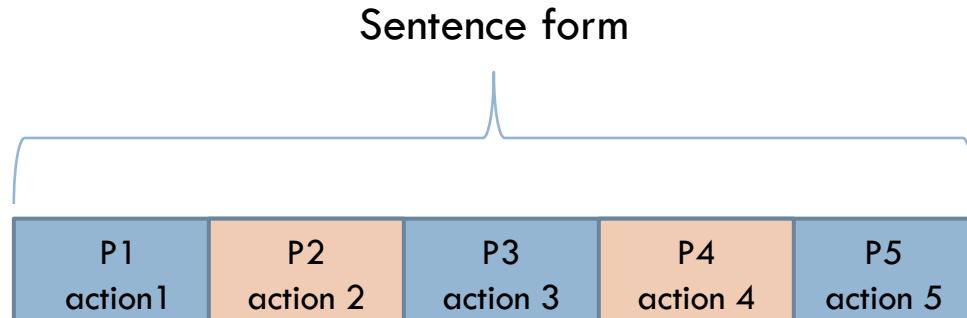
Idea of application business rules by grammar systems

10

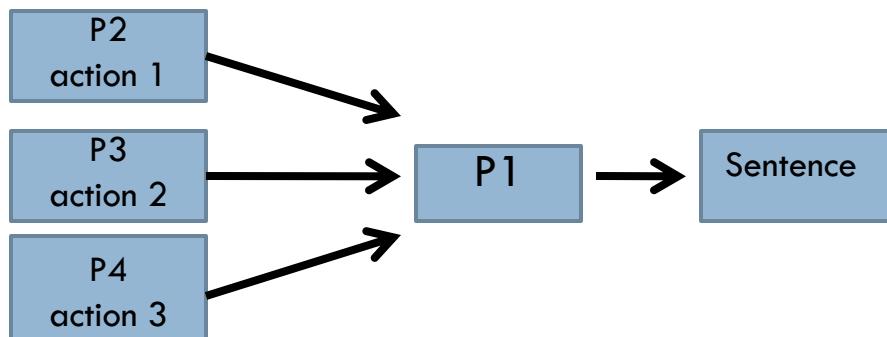
- Business rule
(simple)

```
<condition>
  then
    <action 1
      action 2
      ...
      action N>
```

- CD grammar system



- PC grammar system
(centralized)



Idea of application business rules by grammar systems

11

- There are two possible solutions:
 - A) Create business rules to fall into classes for which we have created a model of grammar system.
 - B) Create random rules as we need and for each create a special grammar system.
- I focused mainly on the solution A.

Example

12

- Consider the business rule: **Determination of salary**

if<working experience <= 1 year>

then

(salary = 5 000, variable = 3000, meal vouchers = 50xday, bonus = 2000)

else if<working experience are between 1 and 5 years>

then (salary = 10 000, variable = 6000, meal vouchers = 100xday,
bonus = 2000)

else if<working experience are between 5 and 10 years>

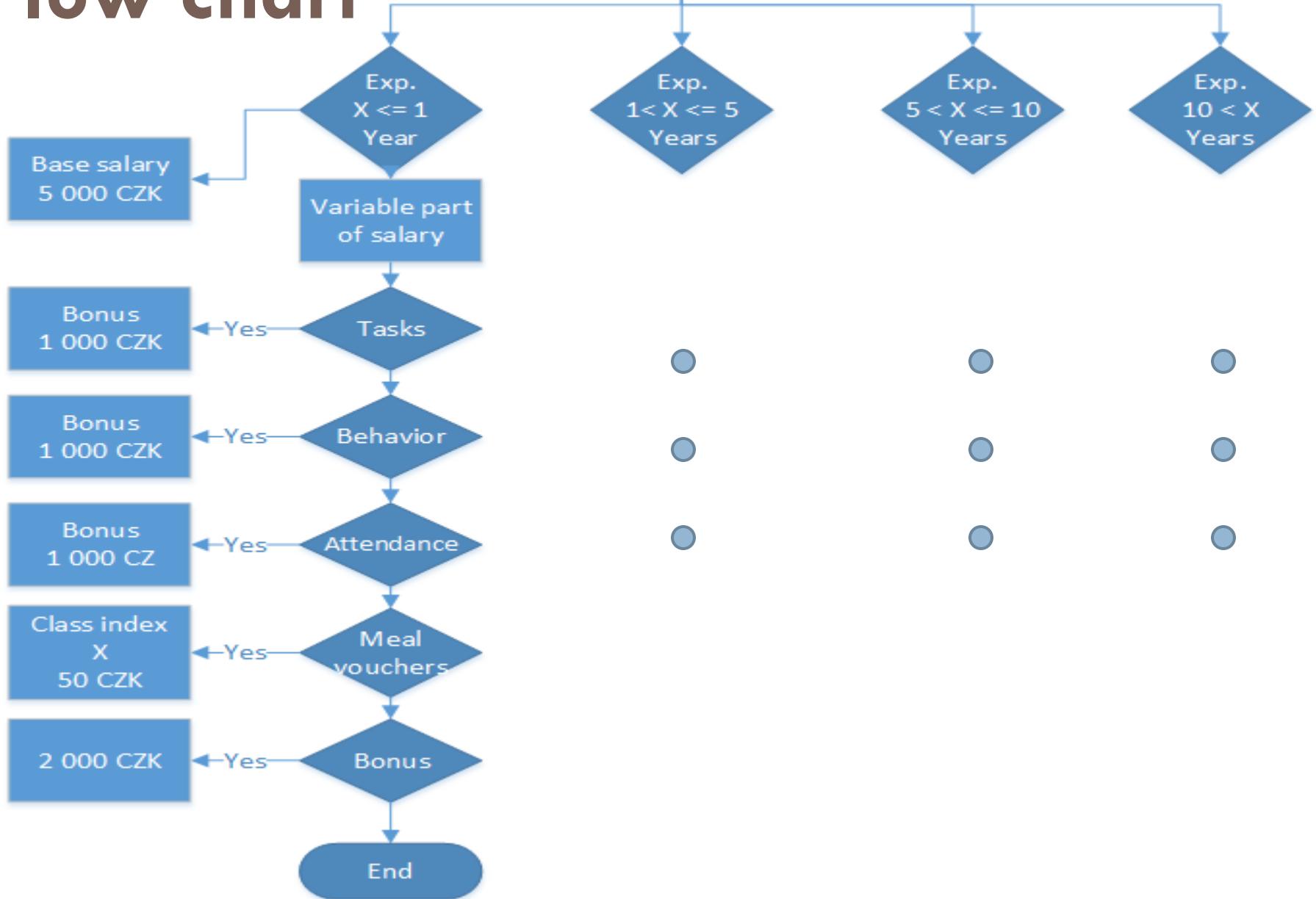
then (salary = 15 000, variable = 9000, meal vouchers = 150xday,
bonus = 2000)

else if<working experience more than 10 years>

then (salary = 20 000, variable = 12000, meal vouchers = 200xday,
bonus = 2000)

Flow chart

13



Using CD grammar system

14

- We use CD grammar system from lectures:
- Constants:
 - n = salary class
 - a = base of salary (typically CZK 5 000)
 - b = base of meal vouchers (CZK 50) * days in month
 - c = base of bonuses = tasks + behavior + attendance
 - d = bonus (CZK 2000)
- $\Gamma = (N, T, S, P_1, P_2)$
 - N = {S, A, B, C, D,}
 - T = {a, b, c, d}
 - S = {S}

Using CD grammar system

15

- $P_1 = \{S \rightarrow S, S \rightarrow ABd, C \rightarrow A, D \rightarrow B\}$
- $P_2 = \{A \rightarrow aCb, B \rightarrow cD, A \rightarrow ab, B \rightarrow c\}$

This CD grammar system generates language:

$$L_{=2} (a^n b^n c^n d : n \geq 1)$$

For example: Salary class 2

$$a^2 b^2 c^2 d =$$

$$\begin{aligned}5000 + 5000 + (50 \times 30) + (50 \times 30) + (1000 + 1000 + 1000) \\(1000 + 1000 + 1000) + 2000 = 21000 \text{ CZK}\end{aligned}$$

Example table

16

	Eperience	Base salary	Tasks	Behavior	Attendance	Meal vouchers	Bonus
n = 1	Less than 1 Year	5 000 CZK	1 000 CZK	1 000 CZK	1 000 CZK	50 CZK per day	2 000 CZK
n = 2	1 < X <= 5 Years	10 000 CZK	2 000 CZK	2 000 CZK	2 000 CZK	100 CZK per day	2 000 CZK
n = 3	5 < X <= 10 Years	15 000 CZK	3 000 CZK	3 000 CZK	3 000 CZK	150 CZK per day	2 000 CZK
n = 4	More than 10 Years	20 000 CZK	4 000 CZK	4 000 CZK	4 000 CZK	200 CZK per day	2 000 CZK



a^n

b^n

c^n

d

Using PC grammar system

17

- Now we use centralized PC grammar system (from lectures)
- $\Gamma = (N, K, T, (S_1, P_1), \dots, (S_n, P_n))$
 - $N = \{S, A, B, C\}$
 - $K = \{Q_2, Q_3\}$
 - $T = \{a, b, c\}$
 - $P_1 = \{S \rightarrow abc, S \rightarrow a^2b^2c^2, S \rightarrow aA, S \rightarrow a^3Q_2, A \rightarrow aA, A \rightarrow a^3Q_2, B \rightarrow b^2Q_3, C \rightarrow c\}$
 - $P_2 = \{B \rightarrow bB\}$
 - $P_3 = \{C \rightarrow cC\}$

Using PC Grammar system

18

- $(S, B, C) \Rightarrow (aA, bB, cC) \Rightarrow^* (a^n A, b^n B, c^n C)$
 $\Rightarrow (a^{n+3} Q_2, b^{n+1} B, c^{n+1} C) \Rightarrow (a^{n+3} b^{n+1} B, B, c^{n+1} C)$
 $\Rightarrow (a^{n+3} b^{n+3} Q_3, bB, c^{n+2} C) \Rightarrow (a^{n+3} b^{n+3} c^{n+2} C, bB, C)$
 $\Rightarrow (a^{n+3} b^{n+3} c^{n+3}, bbB, cC)$
- $L_r(\Gamma) = L_{nr}(\Gamma) = \{a^n b^n c^n : n \geq 1\}$

Summary

19

- We have more types of business rules than type (condition – action). For example limiting, etc.
- I studied only condition – action rules in this paper.
- I think that this issue could be a good use of hybrid CD grammar systems:
 - Every component (action) can have own mode.

Summary

20

- Using centralized PC grammar systems could also be useful:
 - Every action can be performed independently and component P_1 then create final string.
 - This system we can use for application a set of business rules.
- I thing better idea is to create business rules to fall into classes for which we have created a model of grammar system.

Future development

21

- We must create business dictionary with all expressions that we use in organization (set of terminal symbols).
- We make business rules from this terminal symbols by special rules (patterns).
- This patterns categorized business rules into the classes which we have model of grammar system for each of these classes.

Future development

22

- 1) We must create united dictionary of organization
 - $T_{people} = \{\text{client, employe, boss}\}$
 - $T_{subject} = \{\text{salary, meal_voucher, bonus, experience}\}$
 - $T_{period} = \{\text{year, month, week, day}\}$
 - $T_{money} = \{\text{thousand, voucher, 5thousands, bonus2k}\}$
- 2) We must create patterns of business rules
 - If ($<T_{people}><T_{subject}> \leq \text{number } <T_{period}>$)
(client experience \leq 1 year)
 $<5\text{thousand}>^n <\text{voucher}>^n <\text{bonus2k}>^n$
(where n is salary class identified from condition)
- 3) We must create grammar system for each pattern
 - This business rule is in class which has own model of grammar system. In this case is it grammar system from lectures which produces language L_2 ($a^n b^n c^n : n \geq 1$)

Bibliography

23

- Jiří Techet, Tomáš Masopust, Alexander Meduna
Parallel Communicating Grammar Systems, Cooperating Distributed Grammar Systems
<http://www.fit.vutbr.cz/~meduna/work/lib/exe/fetch.php?media=lectures:phd:tid:frvs:10-pcgspres.pdf>
- Luděk Dolíhal
Gramatické systémy – teorie a aplikace
<http://www.fit.vutbr.cz/study/courses/TJD/public/0910TJD-Dolihal.pdf>
- Ivana Šabatová
Podniková pravidla pod kontrolou
http://www.galeos.cz/uploads/Soubory/Studie/Podnikova_prawidla_pod_kontrolou.pdf
- VON HALLE, Barbara. The business rule revolution: running business the right way [online]. Cupertino, CA: Happy About, c2006 [cit. 2014-10-13]. xxviii, 291 s. ISBN 978-1-60005-013-8. FIT VUT Brno



**Thank you for your
attention**