

# Java Developers Day

## ***Silniki reguł biznesowych***

Mariusz Kaczor

[mariusz.kaczor@altkom.pl](mailto:mariusz.kaczor@altkom.pl)

Łukasz Szandecki

[lukasz.szandecki@altkom.pl](mailto:lukasz.szandecki@altkom.pl)

slide 1

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Agenda

- Mamy problem..
- Programowanie deklaratywne
- Drools – przykład implementacji JSR-94
- Algorytm RETE
- Tablica Decyzyjna
- Domain Specific Language
- Query Language

slide 2

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Co by było gdyby...

- Zlecenie

- Firma XYZ zamówiła program do sprzedaży produktu/usługi X różnego rodzaju klientom
- W ofercie znajduje się 'n' różnego rodzaju produktów, a w niedalekiej przyszłości planowane jest poszerzenie oferty
- W zależności od klienta mogą mu przysługiwać różne zniżki i promocje
- Promocje będą również zmienne w zależności od rodzaju sprzedawanego produktu/usługi

slide 3

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Co by było gdyby...

- W firmie programistycznej
  - Po 'n' miesiącach pracy zbliża się koniec projektu...
  - Aplikacja jest już prawie gotowa...
  - Zespół wymęczony wyczekuje wypoczynku...
- W tym samym czasie u klienta
  - Utwórzmy nowy rodzaj "klienta biznesowego"
  - Stwórzmy promocje "50% w dół" dla "stałych klientów"
  - itd...

Ponownie w firmie

Tworzymy oddzielny obiekt klienta biznesowego

Modyfikujemy logikę aplikacji

itd..

**Zastosujemy  
silniki reguł**

slide 4

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Programowanie deklaratywne

### – prolog

#### Deklarujemy fakty:

'It is sunny'.

'It is summer'.

'It is hot' :- 'It is summer', 'It is sunny'.

'It is cold' :- 'It is winter', 'It is snowing'.

#### Potem pytamy:

?- 'It is hot'.

Yes

slide 5

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Programowanie deklaratywne

### – prolog cd.

#### Deklarujemy fakty:

$\text{max}(A,B,B) :- A < B.$

$\text{max}(A,B,A).$

#### Pytamy:

?-  $\text{max}(3,4,M).$

$M = 4;$

slide 6

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Do czego przydają się silniki reguł?

- Gdzie stosować silniki reguł
  - problem jest zbyt "delikatny" dla tradycyjnego programowania
  - problem nie jest w pełni zrozumiały
  - logika w systemie zmienia się często
  - analiza biznesowa
  - systemy eksperckie
- Gdzie sprawdzają się lepiej inne języki
  - nie są odpowiednie do sterowania przepływem
  - w małej mierze sprawdzają się przy zarządzaniu procesami

slide 7

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Specyfikacja

- JSR-94 – silniki reguł
  - wymagania dla silników reguł
  - ograniczenia i limity stawiane przy kompilacji
  - mechanizm dostępu do reguł z poziomu programu
  - interfejs do komunikacji z silnikiem reguł
  - nie specyfikuje samego języka reguł

slide 8

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Implementacje

- **Drools – JBoss**
- ILog Rules
- JLisa
- JTP
- Mandarax
- OFBiz Rule Engine
- OpenLexicon
- TyRuBa

• • • •  
slide 9

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Czym jest i jak wygląda Drools?

- Reguła w Drools jest to dwuczęściowa struktura, w której wyróżniamy część Lewostronną (LHS) i Prawostronną (RHS)
- Reguły w Drools mogą posiadać dodatkowe atrybuty
- Podstawowa składnia:

```
rule "<name>"  
  <attribute> <value>  
  when  
    <LHS>  
  then  
    <RHS>  
end
```

slide 10

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Drools

- Element LHS – Left Hand Side, składa się z elementów warunkowych
- Kiedy fakty są umieszczane w pamięci roboczej, następuje próba dopasowania ich do wyrażeń elementów LHS
- W przypadku dopasowania faktu do wyrażenia LHS wykonywane są operacje zapisane w RHS
- Nie można wywoływać reguł
- Reguły uruchamiają się automatycznie w przypadku zmian pamięci roboczej

slide 11

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Drools przykład

```
/*dla każdego klienta 50% zniżki*/
```

```
rule "50% w dol"
```

```
when
```

```
    k : Klient()
```

```
then
```

```
    System.out.println("Super promocja - zniżka  
    50%");
```

```
    k.setDiscount(0.5);
```

```
end
```

slide 12

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Drools przykład

```
/* klientów biznesowych traktujemy  
inaczej */
```

```
rule "Klient biznesowy"
```

```
when
```

```
    k : Klient(biznesowy==true)
```

```
then
```

```
    System.out.println("Klient biznesowy");
```

```
    k.setVat(0);
```

```
    k.setDiscount(0.75);
```

```
end
```

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

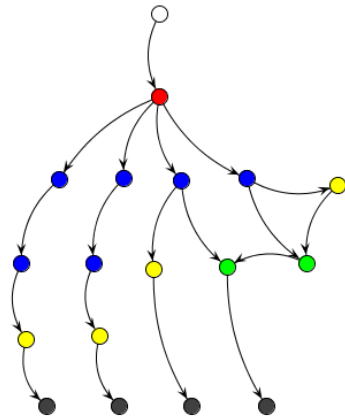
## DEMO

Przykład zapisu i programu  
wykorzystującego silnik reguł

slide 14

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Algorytm RETE



Drzewo RETE dla algorytmu wyliczania liczb Fibonacciego

slide 15

- Opracowany przez Dr. Charlesa Forgy w roku 1978-79
- Algorytm wykorzystywany przez Drools w celu dopasowania obiektów do odpowiednich reguł
- Można go podzielić na dwie części:
  - kompilacja reguł
  - wykonanie reguł
- Algorytm można przedstawić w postaci drzewa

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

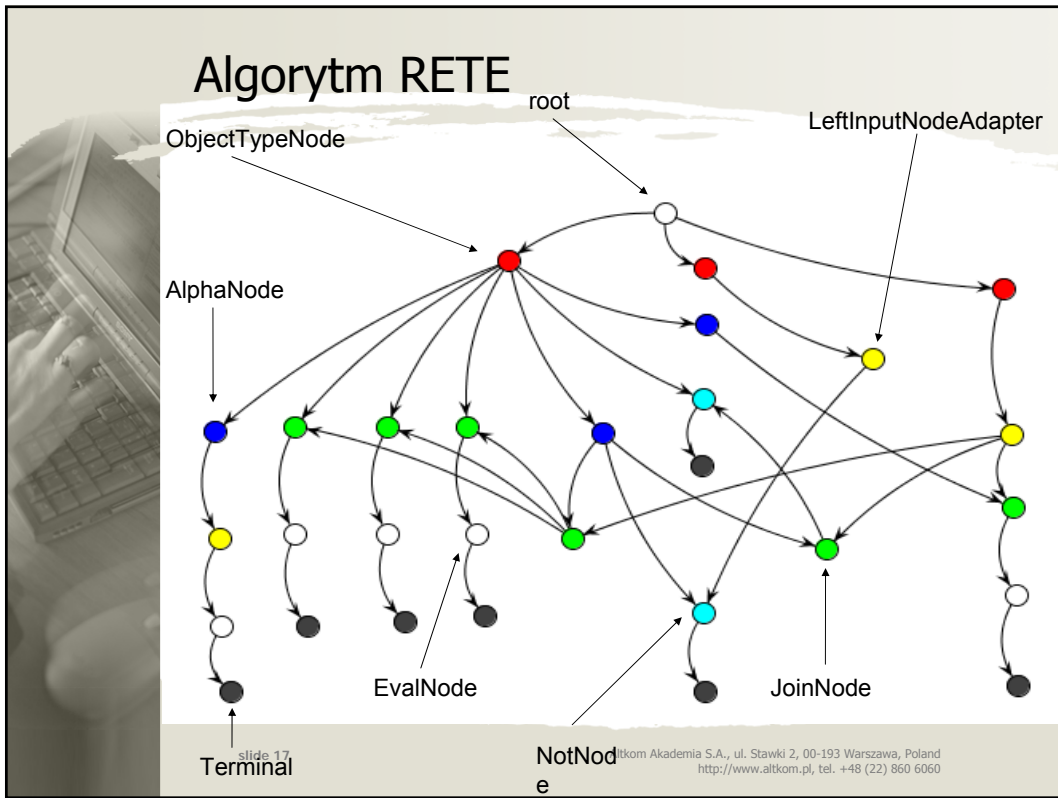
## Algorytm RETE

- W algorytmie RETE wyróżniamy cztery podstawowe typy węzłów: root, 1-input, 2-input i terminal
- Szczegółowe typy węzłów to:
  - root
  - ObjectTypeNode
  - AlphaNode
  - LeftInputAdapterNode
  - EvalNode
  - JoinNode, NotNode (BetaNodes)
  - TerminalNode

slide 16

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060





# DEMO

## Generacja drzewa RETE w programie Eclipse

slide 18

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
http://www.altkom.pl, tel. +48 (22) 860 6060

## Kolejna zagwostka...

- W firmie programistycznej
  - Program już oddany...
  - Drools sprawuje się znakomicie...
  - Teraz jedynie udzielamy supportu i wprowadzamy drobne poprawki...
- W tym samym czasie u klienta
  - Skoro mamy już wsparcie to może by je wykorzystać...
  - Wprowadźmy promocje z okazji kalendarzowego lata...
  - Zmieńmy całkowicie definicje "stałego klienta"...
  - itd...
- Ponownie w firmie
  - Od trzech tygodni wprowadzamy jedynie modyfikacje do reguł...
  - Przecież specjalista od marketingu z firmy XYZ znacznie lepiej zna swoje promocje...

slide 19

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Tablica decyzyjna

- "Dokładna lecz wciąż zwięzła" forma reprezentowania logiki warunkowej.
- Inny sposób wyrażania warunków logicznych
- Zapis reguł w arkuszach
- Wspierane formaty to: Excel i CSV
- W Drools na podstawie tabel generowane są zbiory reguł
- Nie polecane gdy:
  - mało reguł
  - reguły można opisać za pomocą zbioru wzorców

slide 20

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Tabele decyzyjne

- Tabela wymaga nagłówka i narzuca określone zasady
- Jeden wiersz w tabeli – jedna reguła
- Kolumny – warunki jakie obiekt ma spełnić lub akcje jakie należy podjąć
- Kolejność reguł nie ma znaczenia
- Arkusz może zawierać kilka tabel
- Na podstawie całej tabeli generowany jest jeden pakiet reguł

slide 21

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
http://www.altkom.pl, tel. +48 (22) 860 6060

“RuleSet” - słowo kluczowe informujące o nazwie pakietu jaki ma być użyty w wygenerowanym pliku reguł

“RuleTable” - słowo kluczowe informujące o początku tabeli decyzyjnej (pierwszy wiersz i pierwsza kolumna)

CONDITION, ACTION informacje o typie danych w komórkach (LHS lub RHS)

CONDITION	CONDITION	CONDITION	CONDITION	ACTION
Driver	locationRiskProfile	priorClaims	type	policy.setBasePrice(\$param)
Age Bracket	Location risk profile	Number of prior claims	Policy type applying for	Base \$ AUD
LOW		1	COMPREHENSIVE	450
			FIRE_THEFT	200
			COMPREHENSIVE	300
LOW			FIRE_THEFT	150
LOW				150

Opcjonalny element informujący o typie obiektu. Jeśli nie jest podawany należy zostawić pusty wiersz

W przypadku podania typu obiektu w poprzednim wierszu, dane w komórce określają pola składowe które będą porównywane

Wartości które będą porównywane

slide 22

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
http://www.altkom.pl, tel. +48 (22) 860 6060

## DEMO

### Tabela decyzyjna w działaniu

slide 23

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Łatwe podawanie reguł, czyli DSL Domain Specific Languages

- Zapis DSL – tworzenie własnego języka zapisu reguł i mapowanie tak zapisanych reguł na standardowy zapis
- DSL warstwa pośrednia pomiędzy twórcami reguł, a silnikiem reguł
- Zapis reguł z wykorzystaniem DSL pozwala na utrzymanie reguł przez personel “nie techniczny”
- Nie ma wpływu na kształt reguł
- Tekst DSL jest tłumaczony na etapie kompilacji w język zrozumiały dla silnika reguł

slide 24

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Język DSL

- Reguły DSL są zapisywane w zwykłych plikach tekstowych
- Na początku reguły występuje [when] lub [then], które określają typ reguły (LHS lub RHS)
- Przykład:  
[when] There is a Person with name of  
"{name}"=Person(name="{name}")  
[then] Log  
"{message}"=System.out.println("{message}");
- Może utrudniać proces debugowania dla osób początkujących

slide 25

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Język DSL – przykład

Plik DSL:

```
[when]There is a Person with=Person()  
[when]-age is less then {age}=age<{age}  
[when]-name is {name}=name=={name}  
[when]-lives in {country}=country=={country}
```

Wykorzystanie w pliku reguł:

```
There is a Person with  
-age is less then 20  
-lives in 'Poland'
```

```
There is a Person with  
-name is 'Jan'
```

slide 26

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Język DSL – przykład

Pliki DSL mogą również mapować wyrażenia warunkowe, przez co jeszcze bardziej ułatwiają zapisywanie reguł.

### Zapis reguły

```
There is a Person name Jan who lives in  
Poland  
And  
There is a Person name Maria which is 27  
years old  
Or  
Person has an Account
```

slide 27

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## DEMO

Przykład zapisu i programu  
wykorzystującego silnik reguł,  
wykorzystującego DSL

slide 28

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Query

- Query – zapytania składające się tylko ze struktury LHS
- Nie posiadają bloków "when" oraz "then"
- Służą do odpytania pamięci roboczej w poszukiwaniu obiektów spełniających podane kryteria
- Wykorzystywane są w kodzie programu i zwracają obiekty spełniające parametry zapytania

slide 29

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060

## Query - przykład

W pliku reguł:

```
query "car over the age of 10"  
  car : Car ( age > 10 )  
end
```

W kodzie programu:

```
QueryResult result = workingMemory  
.getQueryResult("car over the age of 10")  
  
for (Iterator iter = result.iterator;  
    iter.hasNext();  
    ...
```

slide 30

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060



DEMO

Query w działaniu

slide 31

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060



***Pytania***

slide 32

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060





***Dziękujemy za uwagę.***

Mariusz Kaczor

[mariusz.kaczor@altkom.pl](mailto:mariusz.kaczor@altkom.pl)

Łukasz Szandecki

[lukasz.szandecki@altkom.pl](mailto:lukasz.szandecki@altkom.pl)

slide 33

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland  
<http://www.altkom.pl>, tel. +48 (22) 860 6060