

Salomon  
System przetwarzania wiedzy

*Iteration Description for Construction – 1.0*  
*Wersja 1.1*

Data	Wersja	Opis	Autor
20-12-2005	1.0	Wersja początkowa	Dominik Seweryn
27-12-2005	1.1	Drobne poprawki	Dominik Seweryn

## 1 Zadania dla iteracji

### 1.1 Implementacja algorytmu budowy drzew decyzyjnych

Drugim algorytmem o jaki rozbudowany ma zostać plugin odpowiadający za konstrukcje drzew decyzyjnych jest C4.5. Jest to ulepszona wersja algorytmu ID3. Między innymi atrybuty ciągle są dyskretyzowane (zamiana na przedziały wartości),

### 1.2 Implementacja pluginu wnioskującego

Plugin ten na podstawie stworzonego drzewa, będzie przeprowadzał klasyfikacje nowych danych i przypisywał je do już istniejących grup.

### 1.3 Implementacja wizualizacji w postaci graficznej w pluginie wizualizacyjnym

Plugin odpowiadający za wizualizację w obecnej iteracji będzie prezentował wynik działania algorytmu (utworzone drzewo) z wykorzystaniem JTree umożliwiając obserwowanie drzewa na różnych poziomach szczegółowości.

## 2 Produkt iteracji

Cykl prowadzący do wytworzenia wersji 1.0 rozwijanego rozszerzenia do platformy Salomon także i w tym przypadku nie przewiduje wystąpienia w poszczególnych fazach więcej, niż jednej iteracji. W wyniku fazy Transition powinien powstać zupełnie nowy plugin, umożliwiający wnioskowanie na nowych danych z wykorzystaniem istniejącego drzewa decyzyjnego i decydowanie gdzie zakwalifikować właśnie analizowane dane. Dodatkowo podczas iteracji zostaną rozbudowane dwa pluginy. Plugin budujący drzewa decyzyjne zostanie rozszerzony o nowy znacznie bardziej zaawansowany algorytm - C4.5. Plugin wizualizacyjny będzie prezentował stworzone drzewa w formie graficznej (dotychczas była tylko forma tekstowa) umożliwiając, dzięki rozwijanym węzłom, przeglądanie na różnych stopniach szczegółowości. Niewielkiej modyfikacji ulegnie także plugin wczytujący dane - będzie umożliwiał wybór przedziału wierszy z danymi na których będzie budowane drzewo. Pozostałe rekordy będą mogły być wykorzystane przy wnioskowaniu i właśnie na jego potrzeby jest ta modyfikacja.

## 3 Plan testów

Podobnie jak w poprzednim cyklu przy testowaniu będziemy korzystali z technologii JUnit. Niektóre testy, np. testy działania pluginu odpowiadającego za wizualizację,

będą też przeprowadzane ręcznie.

Testy będą obejmowały sprawdzenie poprawności:

- zaimplementowania algorytmów budowy drzew decyzyjnych.
- działania poszczególnych klas, interfejsów, w tym także interfejsy szkieletowe (np. realizujące operacje działania na bazie danych).
- działania pluginu realizującego budowę drzew decyzyjnych.
- działania pluginu wizualizacyjnego.
- działania pluginu wnioskującego.
- działania całości zrealizowanej funkcjonalności.

## 4 Efekt wykonania iteracji

W omawianej iteracji udało się zrealizować zgodnie z planem wszystkie zakładane rozszerzenia istniejących pluginów, natomiast nie udało się stworzyć pluginu wnioskującego. W związku z tym jego realizacja została przeniesiona do cyklu trzeciego.