



Fakultät II – Informatik, Wirtschafts- und Rechtswissenschaften
Department für Informatik

Projektgruppe StreamCars

Sommersemester 2010 - Wintersemester 2010/11

wöchentliche Arbeitsberichte

vorgelegt von
Benjamin Grünebast

Gutachter:
André Bolles

21. Juni 2010

Inhaltsverzeichnis

Kalenderwoche 20	3
Kalenderwoche 21	4
Kalenderwoche 22	5
Kalenderwoche 23	6
Kalenderwoche 24	7

Kalenderwoche 20

Vom 17. Mai 2010 bis zum 23. Mai 2010.

Erledigte Aufgaben

- Präsentation für Seminar fertig gemacht.
- Ausarbeitung für Seminar fertig gemacht.
- Einarbeitung in Objektmodell und Operatoren in Odysseus
- Einarbeitung in Odysseus
 - Thomas Seminararbeit durchgearbeitet
 - Die Odysseus-Klassen `RealationalTupel` und andere zur Schemadefinition benötigte Klassen „angeschaut“ (verstanden)

Aufgetretenen Probleme

- Objektmodell kann nicht einfach definiert werden.
 - Es leitet sich aus den PAF-Funktionen und den zu implementierenden AAS ab.
 - Die Modelle werden also durch Input- und Outputschemas der Operatoren definiert.
 - Die Schemas werden durch „Create Stream Attribute...“ erzeugt.

Aufgaben für Kalenderwoche 21

- Im allgemeinen, neue Aufgaben suchen.
- Schemadefinitionen der Daten im Datenfluss bzgl. der einzelnen Operatoren herausfinden und besprechen
- Erste Quelle in Odysseus erstellen, um UDP-Verbindung und Daten in Odysseus zu testen

Kalenderwoche 21

Vom 24. Mai 2010 bis zum 30. Mai 2010.

Erledigte Aufgaben

- Erster Klassenentwurf für physische Quelle in Odysseus erstellt.
- Plugin-Projekt `de.uniol.inf.is.odysseus.scars.base` erstellt
- Analogien und Hilfen in dem Odysseus-Projekt `de.uniol.inf.is.odysseus.objecttracking` gefunden. (Programm Code nachvollzogen)

Aufgetretenen Probleme

- Konnten die Quelle leider nicht mit den UDP Daten aus jDVE testen, da Hauke und Sven nicht da waren, und wir das Ding einfach nicht zum laufen bekommen haben, obwohl sie sagten es wäre fertig.

Aufgaben für Kalenderwoche 22

- ...

Kalenderwoche 22

Vom 31. Mai 2010 bis zum 6. Juni 2010.

Erledigte Aufgaben

- Auslesen der richtigen Attribute aus den relationalen Tupeln (Klasse: `RelationalTuple` und `MVRelationalTuple`) versucht nachzuvollziehen, um die Daten für die Algorithmen bereitzustellen.
- Freitag: Exkurs Braunschweig DLR
- Mit Volker die Trennung von Odysseus-Operator und Algorithmus diskutiert und analysiert.

Aufgetretenen Probleme

- Leider immer noch mangelnde Klarheit über das Auslesen der richtigen Attribute in den Operatoren. Die Klasse `SDFExpression` beschreibt die Funktion die in PQL angegeben wird und enthält die Namen der benötigten Attribute. Wie kommt sie in den Operator?

Aufgaben für Kalenderwoche 23

- Bericht zur Initialisierung des Kontextmodells schreiben.

Kalenderwoche 23

Vom 7. Juni 2010 bis zum 13. Juni 2010.

Erledigte Aufgaben

- Am Montag mit Volker Umsetzung von Operatoren eingearbeitet.
- Den Bericht für die Initialisierung des Kontextmodells fertig geschrieben.

Aufgetretenen Probleme

- Volker war Donnerstag leider krank, daher konnten wir nicht zusammen die Operatoren implementieren.

Aufgaben für Kalenderwoche 24

- Operatoren implementieren, jedenfalls einen, damit alle einmal gesehen haben wie das geht.

Kalenderwoche 24

Vom 14. Juni 2010 bis zum 20. Juni 2010.

Erledigte Aufgaben

- Graphische Zeichnung zur Architektur erstellt, um einen Überblick über die zu erstellenden Operatoren und deren Ein-/Ausgabedaten zu erhalten. (Jetzt aber schon wieder veraltet)
- Wichtige Aspekte bei der Implementierung des Filter-Operators geklärt. (z.B. Prädiktionsfunktionen werden dort für die Matrixberechnungen auch benötigt (mit Daniel))
- Die „hoffentlich“ letzten Lücken bei der allgemeinen Implementierung von Operatoren für objektrelationale Daten geschlossen. Genauer: wie komme ich an das Tupel heran, das die liste der detektierten Objekte enthält? Lösung: Name der Liste wird und anderer wichtiger Attribute wie der Zeitstempel werden in der Anfragesprache definiert, und dem logischen Operator als Strings übergeben, anschließend erfolgt mit hilfe des entsprechenden Eingabeschemas die Auflösung in indices der entsprechenden Attribute (nun ist jedoch die Auflösung das problem!).
- Speziellen Container-Klasse erstellt, die als Metadatum im Shema die PredictionFunctions enthält (kann vll später für die Filterung mit Hilfsmethoden ausgestattet werden, die bei der Matrixberechnung nützlich sind).
- Mit Volker und den anderen über die Parametrisierung und aufteilung des assoziationsoperators diskutiert. (Volkers Idee: Assoziationsoperator besteht aus einer Reihe von Operatoren oder Funktionen die die Verknüpfungen zwischen neu und alt mithilfe einer Korrelationsmatrix bewerten, und am ende der Reihe werden jeweils die besten bewertungen gewählt. Somit, gute Parametrisierung mit Hilfe verschiedener Bewertungsoperatoren (nicht nur gating).
- Interface für die Bewertungsfunktion (als Parameter für die Bewertungsoperatoren) mit den anderen erstellt.
- Meinen Senf zum Product backlog dazugegeben.

Aufgetretenen Probleme

- Am Montag und Donnerstag war leider die Lücke für die allgemeine Implementierung von Operatoren für objektrelationale Daten noch nicht geschlossen.
- die Speziellen Container-Klasse die als Metadatum im Shema die PredictionFunctions enthält muss noch mit den anderen besprochen werden.
- Auflösung des Attributnames, bei einem objektrelationalen Schema. (vll. neuen datentyp: „or“ oder „complex“???) woher weiß ich, wo ich im schema nach dem attributnamen suchen muss? Können wir das festlegen? normalerweise sollte das etwas generischer sein. Und was ist, wenn zwei attribute den gleichen namen haben? (kann warscheinlich nicht passieren) Gesetzt den fall man hat den namen der liste, in der lediglich die messobjekte enthalten sind, kann man dann das schema, welches eigentlich ein baum ist, komplett durchsuchen?

Dann reicht nicht einfach nur ein index an welcher stelle das attribute(die liste) steht, sondern es muss ein index pfad angegeben werden. ???

Aufgaben für Kalenderwoche 25

- die Speziellen Container-Klasse die als Metadatum im Shema die PredictionFunctions enthält muss noch mit den anderen besprochen werden.
- die prediction-Op und predicionAssignOp müssen endlich fertig werden, und mit Timo und den anderen zur Erstellung der Grammatik besprochen werden.
- den anderen die neuen erkenntnisse über die implementierung der operatoren (bzgl. der lücken, siehe workDone) erklären, so das alle anfangen können die operatoren zu implementieren.