Schlussbericht JValue-PML

Gemeinschaftliche Modellierung von Offene-Daten-Pipelines

Projektleitung: Prof. Dr. Dirk Riehle

Förderkennzeichen: 19F1133A

Laufzeit: 2022-11-01 bis 2023-10-31

Vorwort

Dieser Abschlussbericht berichtet über die Ergebnisse des JValue-PML Projekts.

I. Projektüberblick

1. Aufgabenstellung

Aufgabe des JValue-PML Projekts ist es, eine domänenspezifische Sprache zur Beschreibung von Daten-Pipelines zu definieren und zu implementieren, so dass Open-Source-Zusammenarbeit an Mobilitätsdaten unter Nutzung öffentlicher Open-Source-Infrastruktur (z.B. GitHub) möglich wird. Die Erwartung ist, dass diese für das Daten-Engineering geschaffene Sprache und Ausführungsumgebung die Erstellung von Daten-Pipelines einfacher, schneller, und weniger fehlerbehaftet macht.

2. Voraussetzungen

Das Projekt ist motiviert dadurch, dass wir die Erfolge und Methoden der Open-Source-Welt in die Offene-Mobilitätsdaten-Welt übertragen wollen. In diesem Projekt vereinfachen wir die Zusammenarbeit durch die Entwicklung einer domänenspezifischen Sprache (DSL) mit textueller Syntax.

Besondere organisatorische Voraussetzungen gab es keine. Das JValue-PML Projekt ist das erste seiner Art; es sollen weitere Projekte folgen.

3. Planung und Ablauf des Vorhabens

Das JValue-PML Projekt wurde als Teil der Antragsvorbereitung geplant und die Ausführung begann mit Projektstart am 2022-11-01. Das Projekt besteht aus zwei wesentlichen Phasen:

1. Konzeption und Implementierung von Jayvee, der JValue-PML Sprache (60%)
2. Demonstration der Nützlichkeit von Jayvee im Daten-Engineering (40%)

4. Wissenschaftliche und technische Ausgangslage

Uns ist keine domänenspezifische Sprache (DSL) für das Daten-Engineering bekannt gewesen und es gibt unseres Wissens auch heute keine (jenseits unserer Arbeit). Die wesentliche alternative Möglichkeit, Daten-Pipelines zu erstellen, besteht in der Verwendung einer allgemeinen (“general purpose”) Programmiersprache (GPL) zusammen mit unterstützenden Bibliotheken.

Ein bekanntes Beispiel für eine GPL ist Python, und die vermutlich bekannteste unterstützende Bibliothek für

Python im Daten-Engineering ist Pandas.

Andere Möglichkeiten wären z.B. die Verwendung von Java (GPL) zusammen mit einer der vielen Apache Bibliotheken wie z.B. Kafka, Spark, Flink, etc.

5. Zusammenarbeit mit anderen Stellen

In der technischen Umsetzung haben wir für dieses vergleichsweise kleine Projekt nicht mit anderen Stellen zusammengearbeitet. In der Öffentlichkeitsarbeit haben wir auf weitere Dienste zurückgegriffen (primär Social-Media-Kanäle).

II. Durchführung und Ergebnisse

1. Verwendung der Zuwendung und erzielte Ergebnisse

Die erwarteten Ergebnisse wurden erreicht und planmäßig umgesetzt..

1. Es wurde eine erste Version der domänenspezifischen Sprache (DSL) für das Daten-Engineering, Jayvee genannt, konzipiert und umgesetzt.
2. Jayvee wird seitdem durchgängig in der Lehre eingesetzt. Der Einsatz in der Lehre dient sowohl der Demonstration der Sprache wie auch dem Aufspüren von Fehlern, welche dann vom über das Projekt bezahlten Entwickler behoben werden.
3. Die Bewertung der erhofften Effekte begann mit Einsatz der Lehre und wird kontinuierlich auch über die Projektlaufzeit hinaus fortgesetzt. Zum Zeitpunkt der Niederschrift dieses Dokuments können wir einen vollen Erfolg der erhofften Wirkung bestätigen.

Die Projektmittel werden auf das Gehalt des Entwicklers angewendet.

1.1 Prüfliste für Vorhabensziele gemäß Zuwendungsbescheid

Festgehaltenes Ziel	**Zielerreichung**
1. Optimierung des Daten-Engineering-Prozess, in dem die Daten zur Nutzung von Open Data extrahiert, bereinigt (transformiert) und in ein Zielformat geladen werden (sog. ETL-Prozess).	Zu 100% erreicht, im Rahmen der versprochenen Funktionalität. Wir können offene Daten besser als zuvor (dank Jayvee / JValue-PML) extrahieren und bereinigen.
2. Entwicklung einer Open-Source-Software zur Datenverarbeitung (einer Modellierungssprache, eines den Daten-Pipelines zugehörigen Compiler und eines Laufzeitsystems für kompilierte Modelle, welche als Daten-Pipeline ausgeführt werden kann).	Zu 100% erreicht. Die versprochene Software steht unter Open-Source-Lizenz auf GitHub der interessierten Öffentlichkeit zur Verfügung, siehe https://github.com/jvalue/jayvee
3. Demonstration der zu entwickelnden Software, die aufzeigt, dass das gemeinschaftliche Erschließen von Open Data mit der neuen Technologie möglich ist.	Zu 100% erreicht. Unsere Demonstration im Rahmen der Lehre hat gezeigt, dass einzelne Personen (die Studierenden) offene Daten mit Jayvee besser als zuvor erschließen können. In zwei Semestern (bisher) konnten unsere Studierenden spannende Analysen auf Basis offener Mobilitätsdaten entwickeln. Ausgewählte (publikumswirksame!) Beispiele finden Sie auf unserem Blog unter

	https://oss.cs.fau.de/tag/made-projects/
4. Qualitative Evaluation der zu entwickelnden Software.	Zu 100% erreicht. Die Studierenden können mit Jayvee alle Aufgaben in hoher Qualität und ähnlich schnell wie mit Python und Pandas erledigen. Wissenschaftliche Publikationen sind in Arbeit.

1.2 Prüfliste für Projektmeilensteine gemäß Zuwendungsbescheid

Meilensteine	**Meilensteinerreichung**
Meilenstein 1 (Ende Juni 2023). Einfache Version des Compilers steht zur Verfügung. Einsatz in Studierendenprojekten zwecks Evaluation ist zuverlässig möglich. Demonstration an einem Beispiel.	100% erreicht, siehe die beeindruckenden Projektbeispiele unter https://oss.cs.fau.de/tag/made-projects/
Meilenstein 2 (Ende Oktober 2023). Durchgeführter Einsatz der zu entwickelnden Software in Studierendenprojekten zwecks Evaluation. Bewertung (Evaluation) der Zielerreichung einer Verbesserung gemeinschaftlicher Arbeit.	Zu 100% erreicht, Die qualitative Evaluation im Rahmen dieses Projekts wurde abgeschlossen, wissenschaftliche Publikationen werden erstellt.

2. Wichtigste Positionen des zahlenmäßigen Nachweises

Aufgrund des späten Projektstarts, und um für die Lehre ab Mai 2023 rechtzeitig fertig zu werden, haben wir einen Teil der Projektmittel nach vorn in die Konzeption und Implementierung von Jayvee gezogen. Durch diesen initialen Kraftakt konnten wir den Projektplan einhalten und im Mai mit der Demonstration beginnen.

Die folgende Tabelle zeigt die prozentuale Verwendung der Projektmittel. Der Gesamtzuwendungsbetrag des Projekts beträgt 83.333 Euro zzgl. Projektpauschale.

2022		**2023**									
Nov	Dez	Jan	Feb	Mar	Apr	Mai	Jun	Jul	Aug	Sep	Okt
3 PM = 18,75%	3 PM = 18,75%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%	1 PM = 6,25%

3. Notwendigkeit und Angemessenheit der geleisteten Arbeit

Es wird die Jayvee domänenspezifische Sprache für das Daten-Engineering von insb. Mobilitätsdaten entwickelt. Die Sprache wird bereits in der Lehre eingesetzt; Fehler werden laufend behoben, um den Lehrbetrieb zu unterstützen.

Wir demonstrieren Jayvee in unserem Kurs MADE. Im Sommersemester 2023 haben 68 Studierende der Informatik und verwandter Disziplinen Jayvee genutzt, um offene Mobilitätsdaten in ein für eine einfache Data-Science Aufgabe brauchbares Format zu überführen. Jede/r der 68 Studierenden löst ausgewählte Aufgaben am Beispiel von Datensätzen aus der Mobilithek mit Jayvee und Python + Pandas, was einen Vergleich von Jayvee mit dem “Gold-Standard” des Data Engineering ermöglicht. Darüber hinaus betreibt jeder Student ein eigenes Data Science Projekt in Verbindung mit der Mobilithek, so dass aktuell 68 Projekte zu offenen Mobilitätsdaten und deren Bereinigung betrieben werden.

Jayvee ist Open-Source-Software und steht auf GitHub unter https://github.com/jvalue/jayvee zur Verfügung. Die von unseren Studierenden entwickelten Daten-Pipelines für offene Mobilitätsdaten stehen ebenfalls auf

GitHub in den jeweiligen studentischen Repositories zur Verfügung.

1	https://github.com/quicktus/2023-amse
2	https://github.com/jackDS008/2023-amse-template
3	https://github.com/uj73yxun/2023-amse-template
4	https://github.com/leoreinmann/2023-amse-template
5	https://github.com/Lavicola/2023-amse-lavicola
6	https://github.com/luccalb/2023-amse-template
7	https://github.com/jnskoe/2023-amse-template.git
8	https://github.com/Radler77/amse
9	https://github.com/MichaelSeyboldt/2023-amse
10	https://github.com/prebbe/2023-amse
11	https://github.com/nmarkert/amse
12	https://github.com/iheziqi/amse-project.git
13	https://github.com/lifeoffelixt/2023-amse-he46pusa
14	https://github.com/EmreR7/2023-amse-template
15	https://github.com/StealWonders/amse
16	https://github.com/stefandnfr/2023-amse
17	https://github.com/Ariffazeel99/2023-amse-template
18	https://github.com/MaeenBadea/2023-amse-template
19	https://github.com/nicolasbandel/2023-amse-nb
20	https://github.com/CarstenSchmotz/2023-AMSE-cs
21	https://github.com/martinreimer/FAU-SS23-DataEngineering/
22	https://github.com/patriotic/SAKI
23	https://github.com/bilalasgharaziz/2023-amse-template
24	https://github.com/mapleprice/2023-AMSE
25	https://github.com/motschel123/2023-amse
26	https://github.com/helenakohl/2023-amse-template
27	https://github.com/shaonsani/shaonsani-amse-template
28	https://github.com/rohitpotdukhe01/2023-amse-yx49uxym
29	https://github.com/Chitra23Ahuja/2023-amse-template
30	https://github.com/mdhasanai/2023-amse-template
31	https://github.com/theanindya/data-engineering
32	https://github.com/thesagni/2023-AMSE-Sagni
33	https://github.com/lnrd96/2023-amse-template
34	https://github.com/dominic0df/2023-amse
35	https://github.com/maanex/2023-amse
36	https://github.com/mujeeb-netizen/2023-amse-template
37	https://github.com/leondaniel22/2023-amse-template
38	https://github.com/tritthart/SAKI-FAU
39	https://github.com/MG-98/DS-project-SAKI

40	https://github.com/derwehr/2023-saki
41	https://github.com/OmarFourati/2023-amse-template
42	https://github.com/fabalex7/2023-amse
43	https://github.com/Magnus-schn/2023-amse-template_magnus
44	https://github.com/janinepa/2023-amse-template
45	https://github.com/kreisligaspieler/2023-amse
46	https://github.com/JobstHanna/2023-amse-template
47	https://github.com/Guenni-Kologe/2023-amse-template_LSc
48	https://github.com/MaxSkaw/2023-amse-template
49	https://github.com/Shagun-Shah/2023-amse-template
50	https://github.com/diganto-deb/2023-AMSE
51	https://github.com/CAgcoder/2023-amse-template
52	https://github.com/um59ipuh/2023-SAKI-um59ipuh-sarker
53	https://github.com/AshnaC/saki-project
54	https://github.com/ChrisMastersoo7/2023-amse.git
55	https://github.com/sujitdebnath/fau-data-engineering-ss23
56	https://github.com/myarmatov/2023-amse-template
57	https://github.com/rskakumanu/2023-amse-template
58	https://github.com/rafia6/2023-amse-template.git
59	https://github.com/Waldleufer/2023-amse-data-engineering
60	https://github.com/tanagha/2023-amse-AnaghaTamhankar
61	https://github.com/agar03yj/2023-amse-template
62	https://github.com/EugBe/2023-amse-template
63	https://github.com/kreuz1995/2023-amse-template
64	https://github.com/notmuchnerdy/2023-amse-template
65	https://github.com/aleemfau/2023-amse-template-23150759-
66	https://github.com/gmMustafa/2023-amse-template
67	https://github.com/HassanRady/2023-amse-template
68	https://github.com/AliAsghar01/2023-amse-template

Speziell möchten wir die folgenden Daten-Engineering und Data-Science-Projekte hervorheben:

- “Weather conditions and music preference” (Link)
- “How to improve police checks for parking violations” (Link)
- “Relationship between charging point infrastructure and electromobility in Germany” (Link)
- “Does weather have a significant impact on the number of highway traffic accidents?” (Link)
- “Effects of weather on renewable energy production” (Link)
- “Bicycle traffic on rainy days” (Link)
- “Train delays and the weather” (Link)
- “Road accidents and their correlation with traffic signs in Berlin in 2022” (Link)

Im Wintersemester 2023/24, also über die Projektlaufzeit hinaus, haben wir wiederum MADE gelehrt, dieses Mal mit 165 Studierenden. Die Ergebnisse finden sich auszugsweise unter https://oss.cs.fau.de/tag/made-projects/ wieder.

Im Sommersemester 2024, also weit über die Projektlaufzeit hinaus, haben jetzt 170 Studierende eine weitere Runde von Jayvee-basierten Projekten mit offenen Daten aus der Mobilithek begonnen. Ergebnisse werden ebenfalls unter https://oss.cs.fau.de/tag/made-projects/ der gezeigt werden.

4. Voraussichtlicher Nutzen

Das JValue-PML Projekt legt die Basis für eine wissenschaftliche Bewertung in Folgeprojekten auf Basis separater finanzieller Mittel. Die in diesem Projekt durchgeführte Demonstration dient dem Beleg der gewünschten Ergebnisse, ist aber nicht für eine ernstzunehmende wissenschaftliche Validierung von Hypothesen ausreichend. Dazu benötigt es weitere Softwareentwicklung sowie kontrollierte Experimente, für welche wir weitere Anträge einreichen werden.

Mit dem Legen dieser Basis für wissenschaftliche Folgearbeiten hat das JValue-Projekt seine wissenschaftlichen Ziele voll erreicht.

Das JValue-PML Projekt konzipiert und implementiert die Jayvee DSL für das Daten-Engineering. Mit Bereitstellung von Jayvee auf GitHub als Open-Source-Software und der Demonstration der Nutzbarkeit und Robustheit von Javyee in unserer Lehre haben wir auch die technischen Ziele des Projekts voll erreicht.

Die im Antrag erläuterte Verstetigung der Ergebnisse durch ein Startup ist weiterhin unser Ziel; für eine Bewertung dessen ist es aber zu früh.

5. Fortschritt bei anderen Stellen

Es gab keine anderen Stellen.

6. Erfolgte oder geplante Veröffentlichungen des Ergebnisses

Es erfolgten und erfolgen zwei Formen der technischen Veröffentlichungen:

- Das Jayvee Projekt auf GitHub, mit dem wir die erarbeiteten Programme der Welt als Open-Source-Software zur weiteren Nutzung bereitstellen.
- Die 68 studentischen Projekte, welche auf GitHub öffentlich bereit stehen, und welche unterschiedliche offene Datenquellen für Data-Science in Form bringen.
- Die 165 Folgeprojekte (Winter 2023/24) und die erwarteten 170 Folgeprojekte (Sommer 2024) werden sich ebenfalls auf GitHub finden lassen.

Die studentischen Projekte enthalten jeweils fünf Übungen. Der konkrete Code beschreibt jeweils eine Pipeline, welche Mobilitätsdaten aus der Mobilithek aufbereitet. Die Pipelines sind zum Vergleich jeweils in Python und

in Jayvee programmiert. Der Code steht auf GitHub bereit und kann von weiteren Parteien aufgegriffen und verbessert werden.

Es laufen mehrere studentische Abschlussarbeiten zu Jayvee und öffentlichen Mobilitätsdaten. Zu erwähnen wären die folgenden Arbeiten:

- Daniel Langbein. Evaluation of Open Data Using the Example of a Public Transport Navigation Website. Bachelor Thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg: 2023.
- Maximilian Lattka. Requirements for an Open Mobility Data Processing Language. Master Thesis. Univ. Erlangen: 2023.
- Johannes Noah Schilling. Processing Open Transport Data: Design and Implementation of an Extension for a Data Pipeline Modelling Language. Master Thesis. Univ. Erlangen: 2023.
- Maximilian Ackermann. Design and Implementation of a Web-based Editor for Data Pipelines. Bachelor Thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg: 2023.
- Elias Pfann, Jayvee Data Wrangler: A UI Tool to Generate Jayvee Pipelines. Master Thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg: 2024

Es gab noch keine wissenschaftlichen Veröffentlichungen, da die Projektlaufzeit dafür zu kurz war und wissenschaftliche Arbeit im engeren Sinne nicht umfasste. Wir haben die Evaluation in qualitativer Form sowie in quantitativer Form (kontrollierte Experimente) fortgesetzt und erste Papier zur Veröffentlichung eingereicht. Wie immer in der Wissenschaft wird es noch seine Zeit dauern.