



netidee

PROJEKTE

PrototAlp

Endbericht | Call 16 | Projekt ID 5899

Lizenz CC-BY

Inhalt

1.	Einleitung	3
2.	Projektbeschreibung.....	3
1.1	Beschreibung der Projektziele/Zielgruppe	4
1.2	Inhaltlicher Überblick über das Projektergebnis	4
1.3	Die System-Architektur von PrototAlp	5
1.4	Besonderheiten	9
3.	Verlauf der Arbeitspakete.....	9
3.1	Arbeitspaket 1 - Detailplanung und Formales am Projektstart.....	9
3.2	Arbeitspaket 2 - Use-case-Evaluierung.....	10
3.3	Arbeitspaket 3 – Setup und Management der Infrastruktur.....	11
3.4	Arbeitspaket 4 – Software-Setup	11
3.5	Arbeitspaket 5 – Dokumentation und Formales am Projektende.....	12
4.	Umsetzung Förderauflagen	12
5.	Liste Projektergebnisse	13
6.	Verwertung der Projektergebnisse in der Praxis.....	13
7.	Öffentlichkeitsarbeit/ Vernetzung	14
8.	Eigene Projektwebsite	14
9.	Geplante Aktivitäten nach netidee-Projektende	14
10.	Anregungen für Weiterentwicklungen durch Dritte	14

1. Einleitung

Dieser Bericht fasst die Arbeit im Projekt „PrototAlp“ zusammen. Wir geben darin eine Beschreibung der Projektidee sowie der Projektziele, beschreiben die Zielgruppe, für die das Projekt vorgesehen ist, und gehen auf weitere Aspekte des Projekts ein, wie die Arbeitspakete, die Projektergebnisse. Die Verwertung der Projektergebnisse sowie der weiterführenden Aktivitäten.

2. Projektbeschreibung

Die Idee zum Projekt „PrototAlp“ entstand aus persönlicher Erfahrung durch durchgeführte Lehre in diversen Kursen an Universitäten. Die Kurse hatten immer einen Zusammenhang mit den Themen Big-Data, Artificial-Intelligence, Machine-Learning, und Data-Science. Man kann den Student*innen leicht einen Überblick über die aktuellen Themen geben und Beispiele oder Gefahren besprechen. Es ist jedoch oft schwer, die komplexen Algorithmen des Machine-Learning und Methoden von Data-Science in der Praxis zu vermitteln, weil dazu Programmierkenntnisse notwendig sind. Anschaulicher wird es jedoch, wenn Student*innen, auch wenn sie selbst noch keine Erfahrung im Programmieren haben, mit den Werkzeugen experimentieren können. Dies erfordert jedoch in der Regel das Aufsetzen und Installieren komplexer Programmierumgebungen, was in Kursen ohne vorgesehenen Programmierbezug in der Regel aus Zeitgründen einfach nicht möglich ist. Dadurch entstand die Idee, eine Plattform zu entwickeln, in die sich die Student*innen einfach nur online einloggen. Sobald sie eingeloggt sind, haben sie eine fertige Programmierumgebung zur Verfügung, wo alle wichtigen Software-Pakete bereits vorinstalliert sind und auch schon Datensätze zum Experimentieren bereitstehen. Der lästige Aufwand des Einrichtens einer Programmierumgebung mit Installation von notwendigen Software-Paketen entfällt, man kann gleich „loslegen“. Weiters benötigen Experimente in den erwähnten Anwendungsfeldern oft spezielle Hardware, damit die Berechnungen zeitnah durchgeführt

werden können. Speziell starke Grafikkarten sind hier von großem Vorteil. Die Plattform soll diese Ressourcen ebenfalls bereits zur Verfügung stellen, wodurch den Student*innen keine hohen Kosten für den Erwerb ebendieser entstehen.

Diese Idee wurde im Projekt „PrototAlp“ umgesetzt. Im folgenden Kapitel beschreiben wir im Detail, welche Ziele gesetzt wurden und welche Zielgruppen PrototAlp anspricht.

1.1 Beschreibung der Projektziele/Zielgruppe

Ziel des Projekts war es, eine Plattform als „Platform-as-a-Service“ zu errichten. Die Plattform stellt eine interaktive Programmierumgebung zur Verfügung, in der eigener Programm-Code ausgeführt und getestet werden kann. Diese Plattform ist auf entsprechend leistungsfähiger Hardware gehostet, sodass Experimente im Bereich Data-Science, Artificial-Intelligence und Machine-Learning zeitgerecht durchgeführt werden können, weil diese in der Regel sehr berechnungsintensiv sind.

Zielgruppe. PrototAlp gewinnt durch die gestiegene Relevanz und auch Sichtbarkeit von künstlicher Intelligenz an Bedeutung. Speziell große Sprachmodelle, die selbstständig Texte schreiben oder Bilder generieren können, erfreuen sich immer größerer Beliebtheit. Immer mehr Personen möchten in die Thematik einsteigen, wodurch auch Bildungseinrichtungen nachgezogen haben, und vermehrte dementsprechende Schwerpunkte anbieten. Folgende Zielgruppen sind aus unserer Sicht relevant:

- **Schüler*innen und Student*innen**, die eine dementsprechende Ausbildung erhalten und PrototAlp für Aufgaben, Übungen oder eigene Anwendungen verwenden können.
- **Lehrende**, die in Kursen mit Bezug zu KI und Machine-Learning unterrichten und eine schnell einsatzfähige Programmierplattform im Unterricht einsetzen wollen.
- **Bildungseinrichtungen**, die dementsprechende Kurse und Lehrgänge anbieten. Diese können PrototAlp einsetzen, um damit die technische Infrastruktur zur Verfügung zu stellen. Sie können PrototAlp entweder selbst hosten, wenn genügend Kapazitäten zur Systemadministration zur Verfügung stehen, oder als „Platform-as-a-Service“ anmieten, um vom eigenen Hosting entlastet zu werden (siehe auch Kapitel „Verwertung“).
- **Data-Scientist und Firmen oder Organisationen**, die in diesen Bereichen Produkte oder Services anbieten.

Auch Personen, die sich aus persönlichem Interesse in die Thematik einarbeiten wollen, können PrototAlp als leicht zu verwendende Plattform für eigene Experimente einsetzen.

1.2 Inhaltlicher Überblick über das Projektergebnis

Das Ergebnis des Projekts ist eine Plattform, die als Programmierumgebung speziell für Experimente und Prototypisierungen im Bereich Data-Science, Machine-Learning und Künstliche Intelligenz geeignet ist. Die für das Deployment notwendige Software steht unter einer offenen Lizenz zur Verfügung und kann von Interessierten auf eigenen Systemen gehostet werden.

Innerhalb der Research Studios Austria Forschungsgesellschaft ist eine weitere Verwertung in eine Weiterentwicklung als kommerzielle „Platform-as-a-Service“ angedacht, die von Kund*innen angemietet werden kann.

1.3 Die System-Architektur von PrototAlp

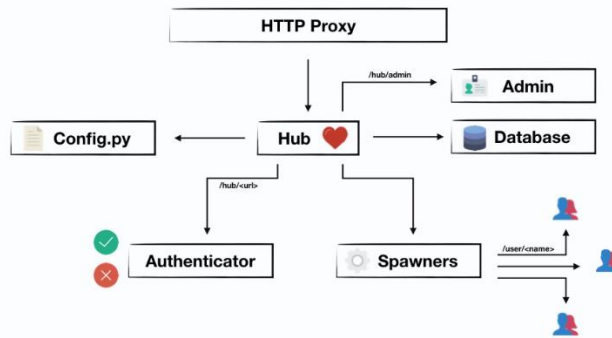
PrototAlp basiert auf der Open-Source-Software „JupyterHub“¹. Ursprünglich war die Verwendung von Jupyter Notebooks² vorgesehen, die gerne von Data-Scientists verwendet werden und weit verbreitet sind. Dadurch reduziert sich der Einarbeitungsaufwand, da die künftigen Anwender*innen bereits mit den Grundprinzipien des Systems vertraut sind. Nach intensiver Recherche (siehe Arbeitspaket 2) wurde jedoch schnell klar, dass JupyterHub die geeignetere Variante ist. Jupyter Notebook ist sehr gut für den Einsatz auf individuellen Systemen, beispielsweise auf dem eigenen Laptop oder Desktop-Rechner geeignet. Es eignet sich jedoch nur bedingt für den Einsatz als Cloud-System, auf das mehrere Anwender*innen gleichzeitig zugreifen sollen. Mit JupyterHub ist dieses Problem jedoch gelöst. Es kann eine Cloud-Umgebung aufgesetzt werden, wo mehrere Anwender*innen gleichzeitig auf ihre Notebooks, und damit auf ihren Code zugreifen können. Weiters löst JupyterHub bereits Probleme, die bei Cloud-Systemen auftreten, z.B. das User-Management oder die Authorization. Zudem kann JupyterHub, je nach ausgewähltem Deployment, auf eine sehr hohe Anzahl an Anwender*innen hochskaliert werden, was für einen kommerziellen Einsatz hochrelevant ist. Möglich wird dies durch die Verwendung eines Kubernetes³-Clusters im Hintergrund, mit dem zusätzliche Pods für weitere Anwender*innen einfach generiert werden können. **Abbildung 1** zeigt die Systemarchitektur von JupyterHub. Wie erwähnt gibt es bereits einen Authenticator. Ein Admin-Bereich erlaubt die einfache User*innen-Verwaltung. Mit Hilfe der Spawner können neue Instanzen generiert werden, auf denen für alle Anwender*innen Jupyter Notebooks zur Verfügung stehen.

¹ JupyterHub: <https://jupyter.org/hub>, Zugriff am 10.05.2023.

² Jupyter Notebook: <https://jupyter.org/>, Zugriff am 10.05.2023.

³ Kubernetes: <https://kubernetes.io/>, Zugriff am 10.05.2023.

JupyterHub



All icons where obtain on Flaticon (<https://www.flaticon.com/packs/essential-collection>)

Abbildung 1 Die Systemarchitektur von Jupyterhub⁴.

JupyterHub selbst stellt rein nur die gemanagte Programmierumgebung zur Verfügung. Für PrototAlp waren daher eine Reihe von Weiterentwicklung notwendig. In JupyterHub sind keine Software-Libraries vorinstalliert. Wir entschieden uns dazu, eine große Reihe von Libraries bereits vorinstalliert zur Verfügung zu stellen, um den Anwender*innen die lästige Selbstinstallation zu ersparen. Zur Ermittlung der notwendigen Libraries führten wir eine Kombination aus Umfrage und Desk-Research durch (siehe Arbeitspaket 2). Die ermittelten Libraries wurden dann in einem Docker⁵-Image zusammengefasst und definiert. Das Docker-Image kann in JupyterHub mit Hilfe von Command-Line-Befehlen geladen werden und können sofort von den Anwender*innen zur Verfügung. Es gelang uns, hier einen effizienten Prozess für das Re-Deployment auszuarbeiten, wodurch Updates mit neu entstandenen oder entdeckten Libraries einfach umgesetzt werden können (Details dazu in Arbeitspaket 4). Dieses Docker-Image hat den Namen „prototAlp-Notebook. Es basiert auf existierenden Notebooks (siehe Abbildung 2), z.B. dem „datascience-notebook“⁶, wodurch schon viele gängige Bibliotheken inkludiert sind, und erweitert dieses um eine große Anzahl neuer Bibliotheken. Das prototAlp-Notebook ist öffentlich zugänglich und auf Github⁷ und Dockerhub⁸ verfügbar.

⁴ Subsystem von JupyterHub: <https://jupyterhub.readthedocs.io/en/stable/>, Zugriff am 10.05.2023.

⁵ Docker: <https://www.docker.com/>, Zugriff am 10.05.2023.

⁶ Datascience-notebook: <https://github.com/jupyter/docker-stacks/tree/main/datascience-notebook>, Zugriff am 10.05.2023.

⁷ PrototAlp-notebook auf Github: <https://github.com/Researchstudio-DSc/prototAlp-notebook>, Zugriff am 10.05.2023.

⁸ PrototAlp-notebook auf DockerHub: <https://hub.docker.com/r/michaelbochrsa/prototAlp-notebook>, Zugriff am 10.05.2023.

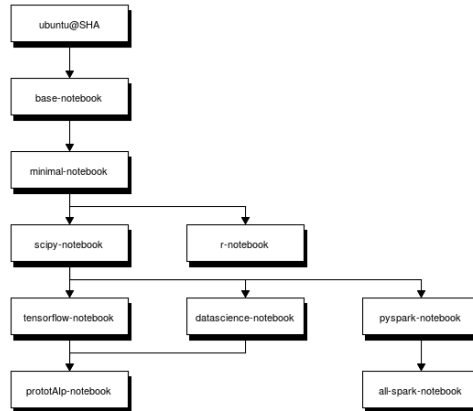


Abbildung 2 Das prototAlp-Notebook kombiniert unterschiedliche Vorgänger miteinander und erweitert diese.

Sobald PrototAlp auf einem Host-System deployed wurde, kann darauf über einen herkömmlichen Web-Browser zugegriffen werden. Der erste Schritt für Anwender*innen besteht daraus, sich in der Plattform zu registrieren und somit einen Account anzulegen. Sobald dies geschehen ist, können sie sich auf der Plattform einloggen (siehe Abbildung 3).

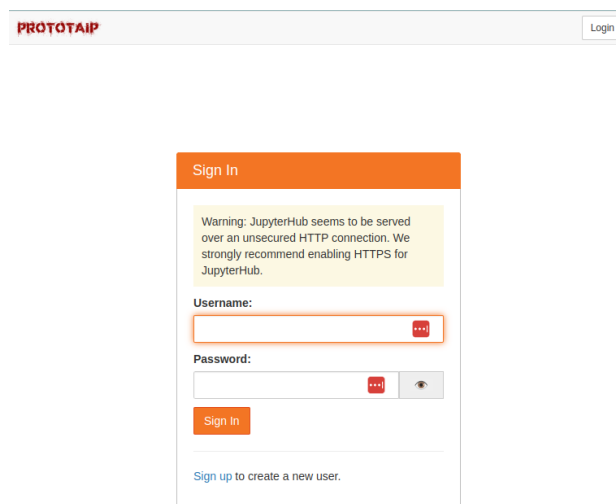


Abbildung 3 Der Login-Bereich der Plattform.

Sobald die Anwender*innen eingeloggt sind, können sie mit ihrer Arbeit beginnen. Abbildung 4 zeigt die Arbeitsoberfläche von PrototAlp. Wie erwähnt basiert PrototAlp auf JupyterHub, einer hochskalierbaren Plattform für Jupyter Notebooks. Die Arbeitsoberfläche besteht daher aus den weithin bekannten Jupyter Notebooks. Die Oberfläche bietet eine Dateiverwaltung (linke Seite), sowie das Hauptfeld für interaktives Programmieren.

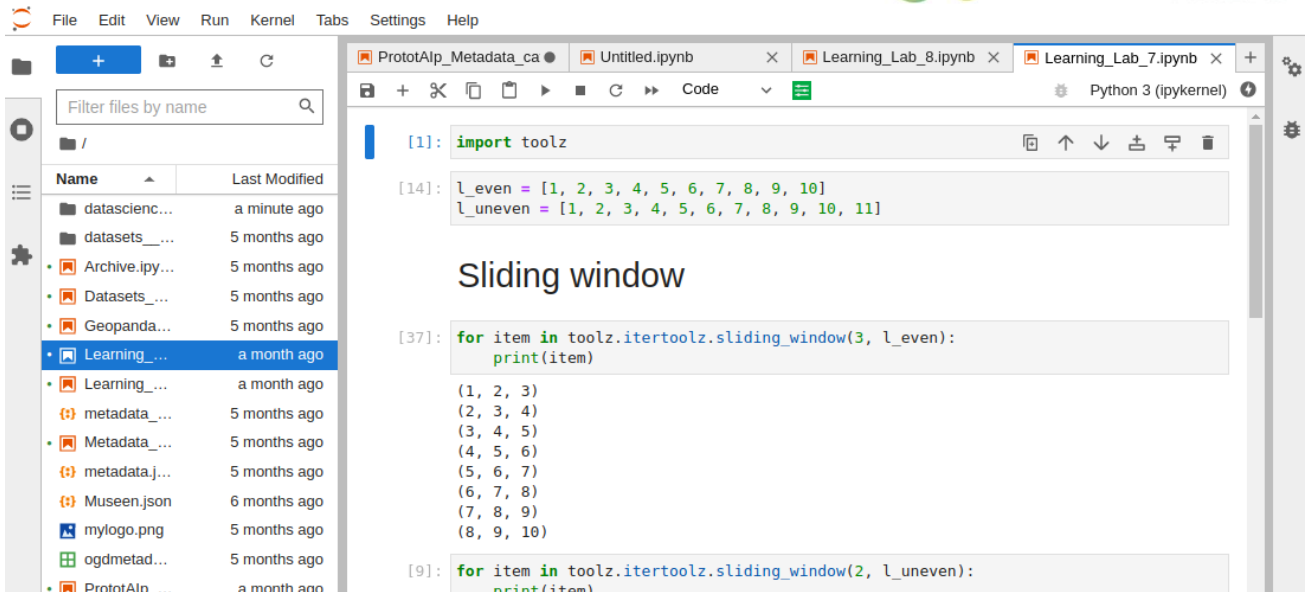
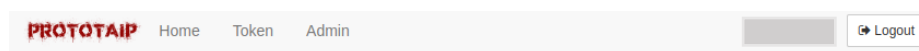


Abbildung 4 Die Arbeitsoberfläche in PrototAlp besteht aus den weit verbreiteten Jupyter Notebooks.

Weiters besitzt PrototAlp eine Benutzerverwaltung, die ebenfalls von JupyterHub bereitgestellt wird (siehe Abbildung 5). Damit kann typisches User-Management durchgeführt werden, wie beispielsweise Autorisierungen zurückzunehmen oder Passwörter zu ändern. Insgesamt steht damit eine ausgeklügelte Benutzerverwaltung zur Verfügung, mit der auch Zwei-Faktor-Authentifizierung möglich ist.



Authorization area

Username	Has 2FA?	Is authorized?		
saturn	False	Yes	Unauthorize	Change password
[REDACTED]	False	Yes	Unauthorize	Change password
[REDACTED]	False	Yes	Unauthorize	Change password
[REDACTED]	False	Yes	Unauthorize	Change password
[REDACTED]	False	Yes	Unauthorize	Change password

Abbildung 5 Die Benutzerverwaltung von PrototAlp.

Durch den geschickten Einsatz robuster Technologien könnte in PrototAlp eine Plattform geschaffen werden, die ideal für den Einsatz im Bereich Data-Science, Machine-Learning, und künstliche Intelligenz ist. Bei steigenden Nutzer*innenzahlen kann PrototAlp verhältnismäßig einfach hochskaliert werden. Weiters wurde die Plattform so entworfen, dass sie leicht auf Neuerungen adaptiert werden kann. Die Welt von Data-Science, Machine-Learning und KI verändert sich rapide, es gibt ständig Weiterentwicklungen bestehender Tools oder völlig neuartige Tools. PrototAlp hat die notwendige technologische Basis, um mit diesen Herausforderungen umgehen zu können.

1.4 Besonderheiten

Im Antrag war der Kauf einer leistungsfähigen Grafikkarte beabsichtigt. Diese wurde mittlerweile von einem externen Sponsor für die Laufzeit des Projekts zur Verfügung gestellt und eine Strategie für eine Verwertung nach Projektende wird ausgearbeitet. Es handelt sich um eine Nvidia A40⁹. Diese Grafikkarte wurde speziell für Datenzentren entwickelt und bietet die Möglichkeit zur Virtualisierung. Wenn mehr als ein/e Anwender*in mit der Grafikkarte arbeitet, dann kann diese virtuell in kleinere Grafikkarten „zerlegt“ werden, ohne dass sich die Berechnung unterschiedlicher Anwender*innen in die Quere kommen. Dieser Aspekt ist besonders wichtig für eine spätere Hochskalierung von PrototAlp. Die technische Spezifikation der Grafikkarte ist wie folgt:

- GPU-Architektur: NVIDIA Ampere-Architektur
- Interner Speicher: 48 GB GDDR6 mit ECC
- Bandbreite des Speichers: 696 GB/s
- Anzahl CUDA Cores: 10,752

Weitere Details finden sich im Datenblatt¹⁰ der Grafikkarte. Die Kosten für die Grafikkarte waren € ... Da das veranschlagte Budget nicht fuer den Kauf einer Grafikkarte verbraucht wurde, wurde es in Personalkosten umgewandelt.

3. Verlauf der Arbeitspakete

Hinweis: Sofern sich seit dem Zwischenbericht bei diesem Arbeitspaket keine Veränderungen ergeben haben, kann der betreffende Text unverändert aus dem Zwischenbericht übernommen werden.

3.1 Arbeitspaket 1 - Detailplanung und Formales am Projektstart

(Keine Veränderungen seit Zwischenbericht, daher unverändert übernommen.)

Kurzbeschreibung der Haupttätigkeiten. Die Haupttätigkeiten in diesem Arbeitspaket bestanden darin, organisatorische Arbeiten am Beginn des Projekts durchzuführen und die weitere Vorgehensweise ordnungsgemäß zu planen.

Erkenntnisse zur Vorgangsweise. Es wurden die formellen Aufgaben zum Projektstart durchgeführt (z.B. Abruf der ersten Förderrate). Weiters wurde die Webseite des Projekts auf der Plattform von netidee aufgesetzt sowie erste Ideen für die Blog-Post-Gestaltung gesammelt. Weiters wurde ein internes Projekt-Kickoff geplant, das am 04.03.2022 stattfand.

⁹ Nvidia A40: <https://www.nvidia.com/en-us/data-center/a40/>, Zugriff am 11.05.2023.

¹⁰ Datenblatt Nvidia A40: <https://images.nvidia.com/content/Solutions/data-center/a40/nvidia-a40-datasheet.pdf>, Zugriff am 11.05.2023.

Kurzbeschreibung der erreichten Ergebnisse. Die Einrichtung der Webseite und der Abruf der ersten Förderrate wurden durchgeführt. Es wurde ein vorläufiger Blog-Post-Plan erstellt.

Besondere Erfolge/ Probleme. Der Erfolg entsprach den Erwartungen, es gab keine Probleme.

Gab es große Abweichungen zum Plan? Warum? Die erwarteten Ergebnisse wurden erreicht, es gab keine Abweichungen zum Plan.

3.2 Arbeitspaket 2 - Use-case-Evaluierung

(Keine Veränderungen seit Zwischenbericht, daher unverändert übernommen.)

Kurzbeschreibung der Haupttätigkeiten. Dieses Arbeitspaket hatte das Ziel, die Use-Cases und Bedarfe von späteren Nutzer*innen genauer kennenzulernen.

Erkenntnisse zur Vorgangsweise. Wir konzipierten eine Umfrage, die uns genauere Einblicke in das Nutzungsverhalten späterer Anwender*innen liefern sollte. Diese Umfrage umfasste Fragen zur Anwendungsdomaene (z.B.: Artificial Intelligence, Computer-Human Interfaces, NLP, ...), die Arten von Datensätzen, mit denen unsere Respondent*innen schon gearbeitet haben, ihre Hardware-Bedürfnisse, sowie bekannte und bevorzugte Software-Bibliotheken.

Kurzbeschreibung der erreichten Ergebnisse. Der Fragebogen wurde mit Google-Forms erstellt und innerhalb unseres eigenen Netzwerks von Data-Scientists verschickt. Es gab 7 Respondenten, die den Fragebogen ausgefüllt haben. Diese Zahl ist zugegebenermaßen gering, liefert aber dennoch wichtige Erkenntnisse für die weitere Gestaltung von PrototAlp, die zusammen mit unseren eigenen Erkenntnissen und einer Desk-Research ergänzt werden.

Besondere Erfolge/ Probleme. Das Umfrageergebnis hat bestätigt, dass Anwender*innen von der Verfügbarkeit von leistungsfähigen Grafikkarten profitieren können. Abbildung 6 zeigt das Resultat auf die Frage, ob eine starke GPU einen positiven Effekt auf die Arbeit der Respondenten hätte, die mit großer Mehrheit mit „Ja“ beantwortet wurde.

9) If you had access to a platform with a powerful GPU and preloaded libraries and datasets, do you think it would help you advance projects faster?

7 responses

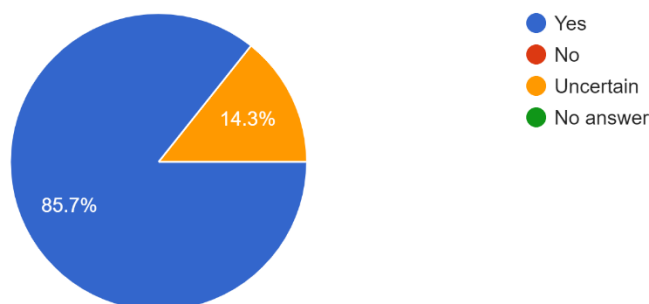


Abbildung 6 Laut Umfrage stellen GPUs eine signifikante Verbesserung bei der Durchführung von Data-Science-Experimenten dar.

In diesem Arbeitspaket gab es keine Abweichung vom Plan.

3.3 Arbeitspaket 3 – Setup und Management der Infrastruktur

(Der folgende Teil wurde vom Zwischenbericht auf Grund der starken Ähnlichkeit übernommen. Weitere Arbeiten nach dem Zeitpunkt des Zwischenberichts sind unten beschrieben.)

Kurzbeschreibung der Haupttätigkeiten. In diesem Arbeitspaket wurde die technische Infrastruktur von PrototAlp aufgesetzt. Dafür wurde ein leistungsfähiger Rechner eingesetzt, der mit einer speziellen, für den Einsatz für Data-Science-Projekte optimierten, Grafikkarte ausgestattet wurde. Als Betriebssystem wurde Ubuntu Linux ausgewählt, da es weit verbreitet in dieser Domäne ist. Die Virtualisierung der Grafikkarte geschieht über das von Nvidia, dem Hersteller der verwendeten Grafikkarte, bereitgestellte Toolkit.

Erkenntnisse zur Vorgangsweise. Das Arbeitspaket umfasste Tätigkeiten wie die Auswahl einer für die Skalierbarkeit geeigneten Grafikkarte, das Einrichten des Betriebssystems auf dem Host-Rechner, und die Installation notwendiger Software-Pakete für das Management der einzelnen Komponenten von PrototAlp.

Kurzbeschreibung der erreichten Ergebnisse. Ein Server wurde als Host fuer PrototAlp eingerichtet und die notwendigen Hard- und Software-Infrastruktur wurde aufgesetzt. Es wurde Docker als geeignete Lösung für die Orchestrierung von Software-Komponenten ausgewählt.

Besondere Erfolge/ Probleme. Der besondere Erfolg bestand darin, dass Dockerhub effizient als CI/CD-Umgebung genutzt werden konnte. Einerseits wurde dadurch uns Docker-Image öffentlich zugänglich gemacht, andererseits konnten wir so eine Lösung schaffen, mit der Nachinstallationen von Software einfach durchgeführt werden können.

Gab es große Abweichungen zum Plan? Warum? In diesem Arbeitspaket gab es eine Abweichung vom Plan: im Antrag wurde die Anschaffung einer leistungsfähigen Grafikkarte budgetiert. Diese Anschaffung wurde jedoch nicht benötigt, da ein Sponsor für diese Grafikkarte gefunden wurde. Das dafür vorgesehene Budget wurde in Personalkosten umgewandelt und so die Entwicklung von PrototAlp vorangetrieben. Das freigewordene Budget wurde in Personalkosten umgewandelt, wodurch die weitere Integration von Libraries sowie die Entwicklung weiterer Connectoren zu existierenden Open-Source-Datensätzen möglich wurde.

Weitere Entwicklungen seit dem Zwischenbericht. Nach dem Aufsetzen der Hauptarchitektur auf dem Server wurden weitere Optimierungen durchgeführt, wie beispielsweise das Optimieren des Deployment-Prozesses. Dies macht es künftigen Betreibern von PrototAlp leichter, die Plattform on-premise aufzusetzen. Weiters wurden Benchmarking-Experimente durchgeführt, um die Performance der Grafikkarte zu evaluieren.

3.4 Arbeitspaket 4 – Software-Setup

Kurzbeschreibung der Haupttätigkeiten (aus Zwischenbericht übernommen). Das Software-Setup umfasste die Auswahl einer geeigneten Open-Source-Plattform, die für den Einsatz als gehostete Plattform (Client-Server-Architektur) verwendet werden kann, sowie die Auswahl und das Deployment von Software-Libraries, die für Anwendungen im Bereich Data-Science, Machine-Learning und künstliche Intelligenz

geeignet sind. Die Architektur wurde als Micro-Services gestaltet und dadurch leicht containerisierbar. (*Neu seit dem Zwischenbericht*)

Besondere Erfolge/ Probleme. Ein schon im Zwischenbericht beschriebener besonderer Erfolg war derjenige, dass es gelungen ist, eine effiziente Pipeline für die Neuintegration von Software-Libraries einzurichten. Dadurch wurde das Hinzufügen neuer Bibliotheken auf effiziente Weise möglich. Ein weiterer Erfolg ist derjenige, dass in PrototAlp jetzt eine große Anzahl an Software-Bibliotheken zur Verfügung stehen, die ohne Setup-Aufwand von den User*innen eingesetzt werden können.

Gab es große Abweichungen zum Plan? Warum? Es gab keine Abweichungen vom Plan.

Erkenntnisse zur Vorgangsweise

Den “effizienten Re-Deployment”-Prozess beschreiben, wo wir das Docker-Image lokal bearbeiten, dann in Docker-Hub hochladen und von dort wieder in JupyterHub ziehen.

3.5 Arbeitspaket 5 – Dokumentation und Formales am Projektende

Kurzbeschreibung der Haupttätigkeiten. Diese Tätigkeiten umfassten einerseits die Vorgaben wie das Bereitstellen der Projektergebnisse und die Verlinkung der Software-Repositories, aber auch die interne

Besondere Erfolge/ Probleme. Das Problem war das Festhalten an den regelmäßigen Blogposts, speziell wenn man keinen dementsprechenden Hintergrund hat. Dadurch waren unsere Posts nicht so regelmäßig, wie wir es uns gewünscht haben. Um dieses Problem bei zukünftigen Vorhaben zu bewältigen, planen wir die Verwendung eines Post-Kalenders (siehe „Erkenntnisse zur Vorgangsweise“ weiter unten).

Gab es große Abweichungen zum Plan? Warum? Die Abweichungen bestanden darin, dass die Posts nicht immer genau alle zwei Monate erfolgten. Die Posts wurden aber nachgeholt.

Erkenntnisse zur Vorgangsweise Man kann leicht auf die geforderten regelmäßigen Blogposts vergessen. Es empfiehlt sich, bereits bei Projektbeginn die Blogposts zu planen und diese in einem Post-Kalender festzuhalten.

4. Umsetzung Förderauflagen

Dieses Kapitel ist nur relevant, wenn in der Fördervereinbarung spezielle Förderauflagen festgelegt wurden. In diesem Fall soll in diesem Kapitel dargestellt werden, wie diese berücksichtigt werden.

Es gab keine speziellen Förderauflagen.

5. Liste Projektergebnisse

Kurzbeschreibung der erreichten Projektergebnisse jeweils mit Open Source Lizenz und Webadresse (netidee Vorgaben beachten!)

1	<i>PrototAlp setup: Dieses Repository enthält Software-Code für das Aufsetzen und den täglichen Umgang mit einer eigenen on-premise PrototAlp-Instanz. Interessierte Nutzer*innen können damit ihre eigene PrototAlp-Plattform aufsetzen und für ihre eigenen Endanwender*innen betreiben.</i>	<i>MIT License</i>	<i>https://github.com/Researchstudio-DSc/prototAlp</i>
2	<i>PrototAlp notebook: Diese Notebook ist eine Weiterentwicklung des weit verbreiteten Datascience-Notebook. Es enthält eine große Anzahl an vorinstallierten Software-Libraries, die im Bereich künstliche Intelligenz, Machine-Learning, und Data-Science weit verbreitet sind.</i>	<i>MIT License</i>	<i>https://github.com/Researchstudio-DSc/prototAlp-notebook</i>
3	<i>PrototAlp image: Dies ist ein Docker-Image, das eine bequeme Installation des PrototAlp-Notebooks erlaubt. Auf diese Weise können Updates des PrototAlp-Notebooks schnell und effizient die PrototAlp-Plattform integriert werden.</i>	<i>MIT License</i>	<i>https://hub.docker.com/r/rsafgdatascience/prototAlp-notebook</i>

6. Verwertung der Projektergebnisse in der Praxis

Angaben zur Verwertung der Projektergebnisse in der Praxis

Die im Moment geplante Aktivität ist die Einführung und Bewerbung von PrototAlp in Bildungseinrichtungen wie Universitäten und Fachhochschulen. Idealerweise sollten dies Bildungseinrichtungen sein, die Lehrgänge in einem für PrototAlp geeigneten Gebiet anbieten, wie beispielsweise Data-Science, Machine-Learning, Natural-Language-Processing, etc. Diese stellen oftmals keine oder nur ungeeignete Infrastruktur für die Bearbeitung von dementsprechenden Experimenten zur Verfügung, wie beispielsweise Barebone-Linux-Server. Gründe dafür sind die hohen Entwicklungs- und Wartungskosten entsprechender Lösungen. PrototAlp kann hier Abhilfe schaffen: mit PrototAlp können diese Bildungseinrichtung auf einfache Weise eine geeignete Plattform mieten, z.B. für die Laufzeit eines Lehrgangs, und diese dann den Studierenden zur Verfügung stellen.

7. Öffentlichkeitsarbeit/ Vernetzung

Beschreibung der im Rahmen Ihres netidee-Projektes bereits erfolgten bzw. noch geplanten Öffentlichkeitsarbeit oder Vernetzung

Die Öffentlichkeitsarbeit bestand aus den Blogposts, die ausserdem über die Webseite der RSA FG weiter gepostet wurden, sowie die Präsentation von PrototAlp in internen Roadshows (Jahresklausur). Dort wurde auch der erwähnte Sponsor der Grafikkarte gefunden.

8. Eigene Projektwebsite

Wird zusätzlich zur netidee-Projektwebsite noch eine eigene Website betrieben, so ist hier die Adresse anzugeben.

Es wird keine eigene Projektwebsite betrieben.

9. Geplante Aktivitäten nach netidee-Projektende

Sind weiterführende Aktivitäten nach dem netidee-Projektende geplant?

Die nächsten Aktivitäten umfassen die Entwicklung von Komponenten, die die kommerzielle Verwertbarkeit von PrototAlp erhöhen, wie beispielsweise die Integration eines Bezahlsystems, mit dem Nutzer*innen Kapazitäten von PrototAlp buchen können.

Ausserdem ist geplant, Visualisierungs- und Parametrisierungs-Tools für Data-Science-Experimente und Machine-Learning-Modelle zu entwickeln und dadurch die Durchführung von Experimenten effizienter zu gestalten (siehe auch nächste Kapitel „Anregungen für Weiterentwicklungen durch Dritte“).

10. Anregungen für Weiterentwicklungen durch Dritte

Welche Nutzungs- und Weiterentwicklungsmöglichkeiten für Dritte ergeben sich durch Ihr netidee-Projekt bzw. empfehlen Sie?

PrototAlp könnte von der Integration von fix integrierten Lösungen zur Visualisierung der Resultate von Data-Science-Experimenten oder der Parametrisierung von Machine-Learning-Modellen profitieren. Dies würde die Usability erhöhen und die Attraktivität von PrototAlp erhöhen. Im Rahmen dieses Projekts war eine dementsprechende Umsetzung und Entwicklung aber nicht möglich und auch nicht vorgesehen.