

OECON Products & Services GmbH



GAIA-X 4 Future Mobility



Abschlussbericht GAIA-X 4 AMS

OECON Products & Services GmbH

19A21004T

Juni 2025

Gefördert durch:



**Bundesministerium
für Wirtschaft
und Klimaschutz**



Dokumenteigenschaften

Titel	Abschlussbericht
Vorhabenbezeichnung	GAIA-X 4 AMS
Laufzeit des Vorhabens	01.12.2021 – 31.05.2025
Firma / Institut	OECON Products & Services GmbH
Erstellt von	Frank Brennecke
Beteiligte	Frank Brennecke, Jens Bläsche, Lars Martensen, Markus Stöbe
Geprüft von	Andreas Blesse
Freigabe von	Frank Brennecke
Datum	01.07.2025
Version	1.0

Inhaltsverzeichnis

1	Aufgabenstellung	5
2	Eingebrachte Dienste	5
3	ODD-konformes Routing	8
3.1	Kurzbeschreibung Operational Design Domain (ODD)	8
3.2	Beschreibung des OECON-Routingsystems	9
3.2.1	Routing-Dienst	9
3.2.2	POI Routing	11
3.2.3	Umkreissuche	11
3.2.4	Reichweitenanalyse	11
3.2.5	Attribute	12
3.3	Schnittstellen	13
3.3.1	Ortssuche	13
3.3.2	Routing	15
4	Zusammenspiel verschiedener Routing-Systeme	16
4.1	Informationen zum Aufbau der Datenbanken für Straßen und Routing	17
4.2	Spezialitäten der Datengrundlagen	18
4.3	Daten-Eingaben für das Routing	19
4.3.1	Start- und Zielkoordinaten	19
4.3.2	Zwischenstationen und „Travelling Salesman“	19
4.3.3	Korridore und „Über“-Stationen	20
4.4	Daten-Ausgabe des Routings	20
4.5	Übergabe von berechneten Routen	21
4.5.1	Unterschiedliche Projektionen	21
4.5.2	Deckung und Klassifizierung	21
4.5.3	Übergabe von Routen im Projekt	22
4.5.4	Navigation im Fahrzeug	22
4.6	Verkehrsinformationen und ODD	22
4.7	Modifikation zur Berechnung ODD-konformer Routen	23
4.8	Nutzung von C2C-Informationen	26
5	Im Projekt entwickeltes Beispielsystem	26
5.1	Routing-Services	26
5.2	Daten-Basis	27



5.3	Daten-Eingabe: Start-und Ziel-Koordinaten	28
5.4	Daten-Ausgabe: Die Fahrtroute	28
5.5	Einbindung in das GAIA-X-Umfeld	29
5.6	Demonstratoren	29
6	Mitarbeit bei anderen Projektteilen	32
6.1	TP1 und TP2 GAIA-X Framework und Infrastruktur-Ökosystem	32
6.2	TP3 Daten- und Dienste-Ökosystem	32
6.3	TP6 Projektmanagement	33



1 Aufgabenstellung

Das Projekt Gaia-X 4 Advanced Mobility Services (AMS) gehört zur Projektfamilie „Gaia-X 4 Future Mobility“ und fokussiert auf die Entwicklung zukünftiger Mobilitätsanwendungen anhand zweier Use-Cases und deren Erweiterung und Erprobung mit dem bereits im Aufbau befindlichen Daten- und Dienste-Ökosystem von Gaia-X. Das zentrale Element für die Umsetzung der beiden Use-Cases bei dem hier beschriebenen Vorhaben ist, dass sie konform zu dem am 28. Mai 2021 vom Bundesrat beschlossenen Gesetz zum autonomen Fahren umgesetzt werden. Mit Unterstützung einer Gaia-X-Infrastruktur und entsprechenden Datenservices wird diese Konformität in Echtzeit überwacht und der Mehrwert von Gaia-X hervorgehoben.

Kernthema für dieses Projekt ist „Advanced Mobility Services“, welches in beiden Use-Cases angesprochen wird und neben der Anwendung neuartiger Technologie und Technologiebausteine untersucht, ob ein Mehrwert Richtung sicherer, komfortabler und schnellerer Verkehr mithilfe Gaia-X Datenökosystem geschaffen werden kann. Hierfür wurden zwei anwendungsnahe Use Case ausgewählt, die beide autonomes Fahren gemäß L4 Gesetz untersuchen. Für die Umsetzung der Use Cases wird ein Gaia-X Framework mit passender Architektur als Einstieg in die Gaia-X Projektfamilie aufgebaut. Zudem müssen eine Dateninfrastruktur und ein Datenökosystem erstellt werden, welches unter anderem den Gaia-X Anforderungen als auch den Use-Case Anforderungen gerecht wird. Diese Arbeiten bilden die Voraussetzung für die Umsetzung der beiden Use Cases im Rahmen von Gaia-X. Der erste (für OECON relevante) Use-Case „*Sichere Koordination von autonomen Fahrzeugen*“ betrachtet einen nominalen Anwendungsfall. Der Fokus hierbei liegt auf die Koordination der spezifischen Betriebsbereiche automatisierter und vernetzter Fahrzeuge (ODD), bei denen eine automatisierte Funktion oder ein automatisiertes System für den ordnungsgemäßen Betrieb ausgelegt ist. Besonders in höheren Automatisierungsgraden, in denen Fahrzeugführende das System nicht dauerhaft überwachen und so anderen Tätigkeiten nachgehen können, spielt die Definition der ODD eine entscheidende Rolle.

OECON wurde zu einem späten Zeitpunkt Projektpartner, weil ein ursprünglich vorgesehener Partner seine Aufgaben nicht wie geplant wahrnehmen konnte und ausscheiden musste. Zum Zeitpunkt des Eintritts von OECON waren daher die Aufgabenstellungen bereits fixiert – unsere erste Aufgabe bestand darin, die bestehenden Anforderungen mit dem zu füllen, was wir mit unserer Technologie leisten konnten.

2 Eingebroughte Dienste

Im Projekt sollten Gaia-X-Dienste zur Einbettung automatisierter und vernetzter Straßenfahrzeuge betrachtet werden. Es setzt auf dem bereits im Aufbau befindlichen Daten und Dienste-Ökosystem von Gaia-X auf, erweitert dieses zur Validierung der beschriebenen Use-Cases (im OECON-Fall die „Sichere Koordination von autonomen Fahrzeugen“ im Rahmen des Arbeitspakets TP4). Basierend auf diesen Use-Cases werden Gestaltungsprinzipien von Gaia-X und entsprechender Standards und Technologien weiterentwickelt. Im Fokus des Projektes stehen dabei:

- Vernetzung und Kooperation von (automatisierten) Fahrzeugen durch Nutzung der Gaia-X Infrastruktur unter Berücksichtigung des Gesetzesentwurfs für automatisiertes Fahren in den zwei Use-Cases
- Beherrschung anspruchsvoller Fahrsituationen durch Nutzung der Gaia-X Ressourcen (z.B. Queren von Kreuzungen, Überholvorgänge, Bilden einer Rettungsgasse inkl. temporärer Übertretung von Verkehrsregeln)
- Schaffung eines Datenraums für sichere („safe“ und „secure“) Daten und Datendienste sowie authentifizierte Verteilung über eine Datenservice-Architektur

- Daten und Dienste aus hochkomplexen Fahrsituationen im Gaia-X Datenraum bereitstellen

Die Aufgabenstellung lässt bereits erkennen, dass es hier nicht um die komplette Neuentwicklung eines Systems zur Berechnung von Routen für automatisierte Fahrzeuge geht, sondern um die Erweiterung eines bereits bestehenden Systems. Der ursprüngliche Projektpartner hatte ein solches Routenplanungssystem im Angebot, und OECON als Ersatz hatte im Rahmen des von der Bundesregierung geförderten Großprojekts „Schaufenster Elektromobilität“ zwischen 2012 und 2015 ebenfalls ein Routenplanungssystem entwickelt, das seinerzeit speziell auf die Bedingungen von Elektrofahrzeugen abgestimmt war. Dazu war ein Kartendienst geschaffen worden, der folgende Optionen bietet:

Kartendarstellung

Es steht eine Weltkarte zur Verfügung stehen mit einem festlegbaren Standardausschnitt (hier: Bundesland Niedersachsen als erste Ansicht). Die Karte bietet die folgenden Funktionen:

- Vergrößern und Verkleinern des sichtbaren Ausschnitts anhand eines Maßstabs in vorgegebenen Auflösungsstufen
- Zoom durch Mousrad anhand eines Maßstabs in vorgegebenen Auflösungsstufen
- PAN-Funktion, also das Verschieben der Karte durch Klicken, Festhalten und Verschieben der Maus
- Center, also das Zentrieren eines Ausschnitts auf eine vorgegebene Koordinate

Darstellung der Daten

Die Darstellung der einzelnen Informationen ist in Ebenen, sogenannten „Layern“, serverseitig organisiert und kann über unterschiedliche Zoom-level visualisiert werden. Es werden folgende Standarddaten dargestellt:

- Orte (und Teilorte)
- Straßen
- Grenzen
- Gewässer
- Bahnlinien
- weitere topologische Daten
- Innerstädtische topologische Daten

Die einzelnen Standarddaten-Layer enthalten die Daten in priorisierter Form, so dass in einer Übersichtskarte von Deutschland nur die Landesgrenzen, das Autobahnnetz und die wichtigsten Orte zu sehen sind. Beim Hineinzoomen werden dann nach und nach die nachgeordneten Straßennetze, kleinere Orte, Straßenbeschriftungen und topologische Daten wie bspw. Flüsse, Seen und Waldflächen sichtbar.

Ortssuche und Gecodierung

Der Kartendienst nimmt Eingaben zur Ermittlung von Koordinaten entgegennehmen und liefert als Rückgabe entweder eine Vorschlagsliste oder bei Eindeutigkeit die exakten Koordinaten. Die Eingabe kann in verschiedenen Textformaten entgegengenommen werden. Falls mehr als eine vorgegebene Anzahl an Rückgabewerten möglich ist (bspw. x), werden nur die ersten x Werte zurückgegeben mit dem Hinweis, die Eingabe genauer zu spezifizieren.

Points of Interest

Es werden Zusatzdaten mit punktuellen geographischen Bezug (Points of Interest, POI) angezeigt. Insbesondere werden vorrangig POI mit Bezug zur Elektromobilität angezeigt, zum Beispiel:

- Ladesäulen,
- Werkstätten,
- Verleihfirmen (eCars, eBikes),
- Car-Sharingstandorte

Für die POI ist eine Datenbank vorhanden, in der relevante Informationen abgelegt werden, mit folgenden Inhalten: POI Zuordnung, Bezeichnung, Latitude, Longitude, weiterführende Textinformation (z. B. Ladesäulen DC oder AC, Ladesäule für eBike oder eCar, usw.).

Zusätzlich sind andere, frei verfügbare Datenbestände wie Tankstellen, Restaurants, Bars, Schnellimbisse, Haltestellen, öffentliche Gebäude, Museen, Theater, Kinos, Banken, Geldautomaten, Parkhäuser, E-Parkflächen etc. eingebunden.

Die Daten mit Bezug zur Elektromobilität werden bis heute aktiv aktuell gehalten. Andere Zusatzinformationen werden anhand der zur Verfügung gestellten Datenquellen aktuell gehalten und in einer Datenbank verwaltet. Der Status der Ladesäulen (besetzt/reserviert, frei & defekt) wird in Echtzeit angezeigt und findet in der Routenberechnung Berücksichtigung.

Umkreissuche

Durch die Übergabe einer Koordinate soll nach nächstgelegenen Objekten im Umkreis gesucht werden. Die Suche erfolgt nach Luftlinie oder Straßenkilometern, der Radius der Suche soll festlegbar sein. Als Objekte kommen Daten aus den POI-Tabellen infrage, beispielsweise Ladesäulen oder Restaurants.

Reverse Geocoding

Als „Reverse Geocoding“ wird die Möglichkeit bezeichnet, zu einer gegebenen Koordinate den nächsten Ort zu bestimmen. Eine solche Funktion ist vorzusehen und soll die bestmögliche Beschreibung liefern, also einen Ortsnamen mit Straßenbezeichnung und ggf. Hausnummer. Außerorts wird der nächste Ort als „nahe bei“ gekennzeichnet oder der nächste Straßename geliefert („Autobahn A2“).

Mapmatching (Optional)

Als „Mapmatching“ bezeichnet man die Zuordnung einer Koordinate zum nächsten Straßensegment. Sie nimmt wahlweise eine Koordinate mit Richtung oder eine Liste von Koordinaten (als zurückgelegten Weg, wobei die letzte Koordinate gematcht wird) entgegen. Zurückgeliefert wird die Koordinate auf dem nächstgelegenen und am besten passenden Segment zuzüglich einer Segment-ID und einer Beschreibung der Straße (Name).

Routenplanung

Auf Basis der vorhandenen Komponenten wurde ein Routenplaner umgesetzt, der auf Basis der Kartendaten Wege berechnen kann. Zur Berechnung wird ein Netz auf Basis von Kanten und Knoten genutzt, das auf dem gegebenen Straßennetz beruht. Die Routing-Funktion nutzt einen den A*-Algorithmus. Es sind damit zwei Ausprägungen möglich – ein 1:1-Routing (ggf. mit mehreren Stationen) und ein 1:n-Routing. Unter 1:n wird dabei ein Startpunkt mit n Zielen verstanden. Die Routenplanung und die erforderlichen Erweiterungen werden im nächsten Abschnitt beschrieben.

Die Routenplanung bot in der eingebrachten Version bereits einige Funktionen, die sich im Rahmen der Implementierung des ODD-Routings als wichtig erweisen sollten, weil sie sehr feine Spezialisierungen erlauben:



Wetterinformationen

Wetterdaten fließen in die Routenberechnung ein. Es werden folgende Typen unterschieden:

- Sonne – keine spezielle Berechnung nötig
- Regen – normale Berechnung, aber nachträgliche Begrenzung der Geschwindigkeit auf maximal 100 km/h, wodurch die Reisezeit steigt.
- Nebel – normale Berechnung, aber nachträgliche Begrenzung der Geschwindigkeit auf maximal 40 km/h, wodurch die Reisezeit erheblich steigt.

Verkehrsinformationen

Aktuelle Verkehrsdaten werden im Kartendienst angezeigt und bei der Routenplanung berücksichtigt. Dabei werden grundsätzlich nur Verkehrsdaten berücksichtigt, die in einem Umkreis von weniger als 100 km um den Startort liegen. Baustellen führen zu Verzögerungen.

Routen-Entfernungszeitberechnung

Eine Reise-Zeitberechnung ist eine Grundfunktionalität in jeder Routenberechnung. Die Erweiterung für diesen Dienst berücksichtigt die aktuell vorhandenen Reichweitenbeschränkungen von Elektrofahrzeugen in der Berechnung und über die vorhandene Ladesäuleninfrastruktur eine Routenempfehlung ausgeben. Durch Kenntnisse zur Leistung der Ladesäule und dem verbauten Akku-Typ im Fahrzeug werden zeitoptimierte Planungen erstellt. Dies bedeutet, dass Schnelladesäulen auch angefahren werden sollen, wenn die Akkuladung noch mehrere Kilometer reicht, um an dieser mit weniger Zeitverlust zu laden und so nachfolgend eine größere Strecke zurückzulegen.

Umkreissuche-Dienst

Es werden zu einem gegebenen Ausgangspunkt die jeweils nächsten Positionen einer geographischen Datenbank berechnet. Im speziellen kommt dieser Dienst zum Einsatz bei der Ermittlung der jeweils nächsten passenden Ladesäule.

Isochronen-Dienst

Die Umkreissuche allein liefert bei der Berücksichtigung von Ladesäulen kein gutes Ergebnis. Es müssen zusätzlich die zeitlichen Entfernungen berücksichtigt werden. Diese Reichweitenanalyse liefert der Isochronen-Dienst.

3 ODD-konformes Routing

3.1 Kurzbeschreibung Operational Design Domain (ODD)

Eine ODD steckt den Bereich genau ab, in dem automatisierte Fahrfunktionen zum Einsatz kommen dürfen. Somit stellen sie eine grundlegende Voraussetzung für sicheres automatisiertes Fahren dar.¹

Ein automatisiertes Fahrsystem (Automated Driving System, ADS) ermöglicht es einem Fahrzeug, eine bestimmte, dauerhafte dynamische Fahraufgabe (Dynamic Driving Tasks, DDT) auszuführen. Die Hardware- und Softwarekomponenten eines ADS automatisieren gemeinsam diese Fahraufgabe durch die Implementierung verschiedener ADS-Funktionen. Autonomes Fahren dagegen bezieht sich auf ein Fahrsystem, das in der Lage ist, alle seine Funktionen selbstständig und autark auszuführen.

¹ Quelle: <https://safe-intelligence.fraunhofer.de/artikel/odd-enger-rahmen-sorgt-fuer-sicherheit>



Die Norm SAE J3016 enthält eine klare Definition der Autonomiestufen, um die Fähigkeiten des automatisierten Fahrens vom autonomen Fahren zu unterscheiden. In den Stufen 1 und 2 bietet das automatisierte Fahrsystem Funktionen, die eine Fahrerassistenz bzw. eine teilweise Automatisierung des Fahrens ermöglichen. Die Stufen 3 bis 5 reichen von einer bedingten Fahrautomatisierung, die einen einsatzbereiten Fahrer erfordert, bis hin zu einer vollständigen Fahrautomatisierung, bei der kein Fahrer mehr benötigt wird.

Ein Fahrzeug, das mit einem ADS ausgestattet ist, kann mehrere Funktionen mit unterschiedlichen Autonomiestufen einsetzen. Dabei ist es wichtig zu beachten, dass jede ADS-Funktion eine Nutzungsspezifikation erfüllen muss, die ihren Automatisierungsgrad und den vorgesehenen zulässigen Betriebsbereich (Operational Design Domain, ODD) angibt.

Die ODD stellt also einen wichtigen Aspekt dar, um die Sicherheit eines ADS zu gewährleisten. Sie liefert das nötige Wissen über die Fähigkeiten und Grenzen des ADS für eine Reihe von vordefinierten Betriebsbedingungen. Eine ADS-Operation, das heißt der Einsatz von ADS-Funktionen, gilt als unsicher, wenn sie unter Bedingungen durchgeführt wird, die nicht der unterstützten ODD entsprechen. Mit anderen Worten: der Betrieb erfolgt außerhalb der ODD. Ein Beispiel: Wenn sich ein Fahrzeug außerhalb der ODD für eine bestimmte automatisierte Fahrfunktion bewegt, muss es einen nahegelegenen Parkplatz am Straßenrand ansteuern und auf weitere Anweisungen warten.

3.2 Beschreibung des OECON-Routingsystems

Die folgenden Absätze beschreiben das im Projekt weiterentwickelte Routingsystem, das einerseits die neuen Anforderungen erfüllt und andererseits auf den aktuellen Stand der Technik gebracht wurde. Diese betrifft beispielsweise den aus 2015 stammenden Quellcode, der auf Frameworks basiert, die bereits zu Projektbeginn 2022 nicht mehr gewartet wurden. Es wurden auch Verbesserungen im Code durchgeführt, die zu erheblichen Geschwindigkeitssteigerungen führten und den ursprünglich geschätzten Bedarf an benötigten Servern für die Berechnung deutlich reduzierten – dazu später mehr.

3.2.1 Routing-Dienst

Für die Verbindungssuche im Individualverkehr wird ein Routing-Server bereitgestellt und weiterentwickelt, bei dem eine sehr schnelle Routenberechnung möglich ist. Das umgesetzte Routing basiert auf einem Ansatz der „Contraction Hierarchies“². Generell wird bei den Diensten auf Datenbeständen von OpenStreetMap (OSM) zurückgegriffen.

Für die Routenberechnung sind verschiedene Einstellungen möglich. Beim „Auto“ wird die jeweils schnellste Verbindung gesucht. Als „Fahrrad“ werden passende Nebenstrecken vorgeschlagen.

Die Unterschiede zwischen „1:1-Routing“ und „1:N-Routing“ sind in den folgenden Abbildungen dargestellt. Bei 1:1 werden die Zielpunkte nacheinander berechnet. Bei 1:N werden von einem Startpunkt die Zielorte immer vom ersten Startpunkt ausgegeben.

² Siehe hierzu: Delling et al (2009), Engineering Route Planning Algorithms, <http://www.informatik.uni-freiburg.de/~ki/teaching/ws1011/readinggroup/private/delling-et-al-2009.pdf>, Karlsruhe

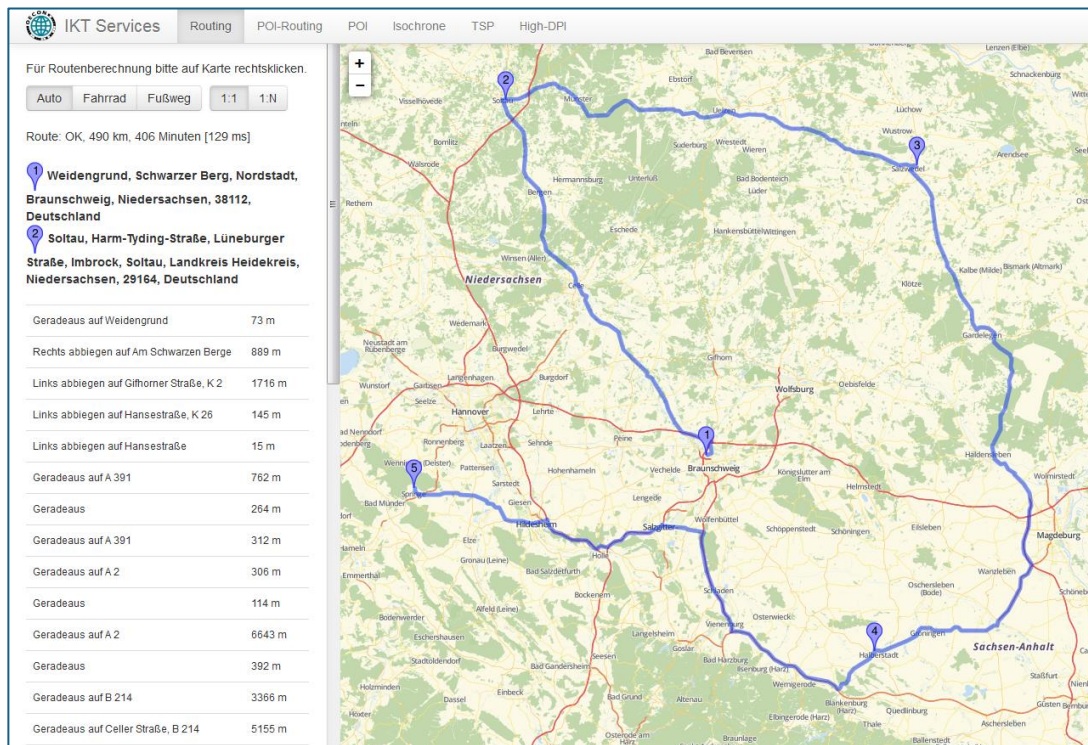


Abbildung 1: Routing 1:1

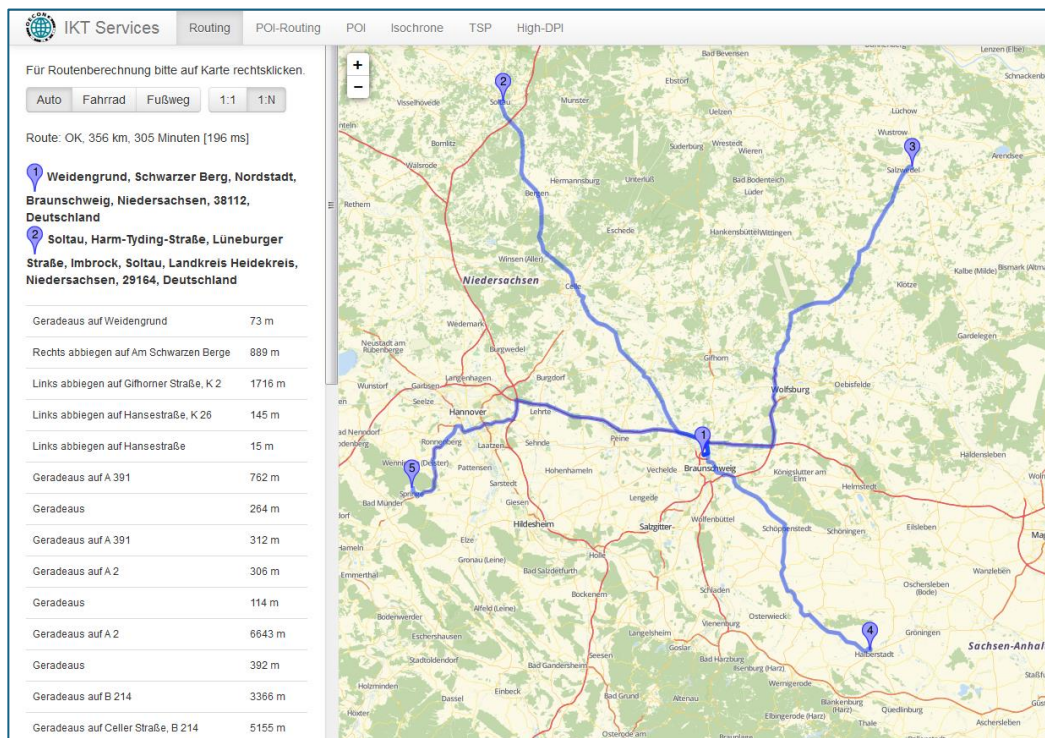


Abbildung 2: Routing 1:N



3.2.2 POI Routing

Bei Elektrofahrzeugen gibt es nach wie vor technische Beschränkungen bezüglich der Reichweite. Außerdem befindet sich die entsprechende Ladesäuleninfrastruktur in Deutschland im Aufbau. Um aktuell ein weiter entferntes Ziel (größer als Reichweitenangabe der Automobilhersteller) zu erreichen muss dabei die aktuell vorhandene Ladeinfrastruktur berücksichtigt werden.

In dem „POI-Routing“-Dienst werden Ladesäuleninformationen abgerufen und mit der Eingabemöglichkeit der zu erwartenden Reichweite von Elektrofahrzeugen bei einer Routenberechnung angewendet.

Das Ergebnis eines Routenvorschlages ist in der nachfolgenden Abbildung dargestellt. Bei einer Route von Oldenburg nach Braunschweig mit einer angenommenen Reichweite von 100 km sind somit drei Zwischenladungen notwendig.

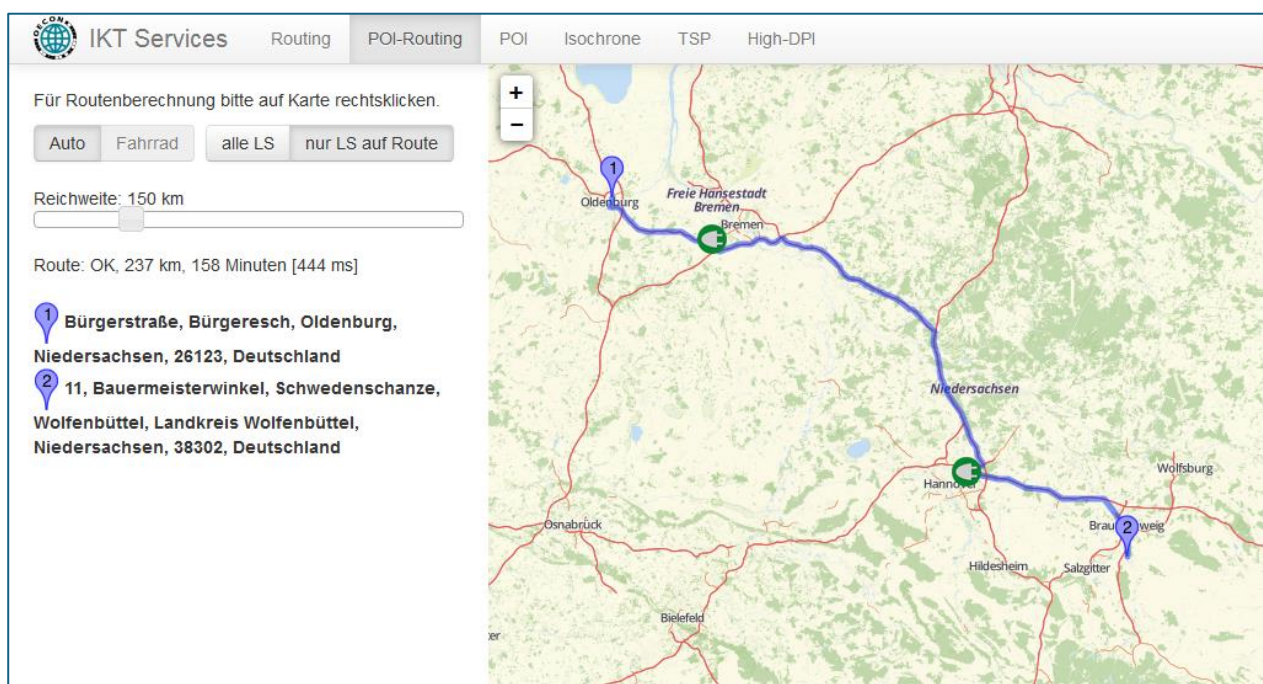


Abbildung 3 Routing über Ladesäulen mit Reichweitenbeschränkung

3.2.3 Umkreissuche

Eine weitere Fragestellung für Besitzer von Elektrofahrzeugen ist, wo sich die nächsten Ladesäulen befinden. Diese Frage kann über eine Umkreissuche einfach beantwortet werden.

3.2.4 Reichweitenanalyse

Von einem Standort werden die erreichbaren Ziele innerhalb einer Zeit / einer Entfernungsvorgabe auf Basis des Routing-Netzes ausgewertet und graphisch dargestellt.

Bei einer zeitlichen Auswertung werden die Geschwindigkeitsprofile des Straßennetzes ausgewertet. In der nachfolgenden Abbildung ist die Erreichbarkeit aus Braunschweig mit einer Zeitvorgabe von 40 Minuten dargestellt. Es ist deutlich zu erkennen, dass durch die Nutzung der Autobahnen eine weitere Wegstrecke zurückgelegt werden kann, als bei Nutzung des untergeordneten Straßennetzes. Eine entfernungsabhängige Auswertung ist zusätzlich möglich.

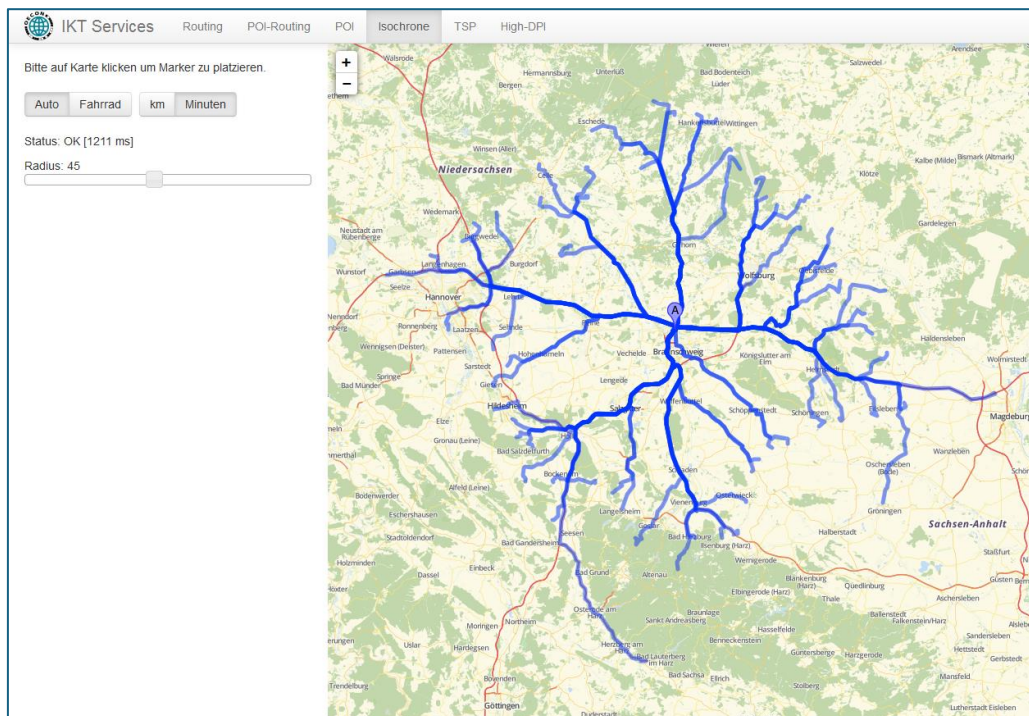


Abbildung 4 Erreichbare Ziele mit Zeitvorgabe

3.2.5 Attribute

Folgende Attribute können im Routing-Dienst konfiguriert werden:

- Startpunkt in Lat/Lon-Koordinaten oder als Bezeichnung aus Ort/Straße/Hausnummer
- Zielpunkt in Lat/Lon-Koordinaten oder als Bezeichnung aus Ort/Straße/Hausnummer
- Fahrzeugprofil mit Durchschnittswerten für Geschwindigkeiten auf Autobahn, Bundesstraße, Landstraße und Stadtstraße, jeweils noch unterteilt in schnell, mittel und langsam.
- Gewünschte Start- oder Ankunftszeit (Tag, Uhrzeit)
- Berücksichtigung von statischen oder schwach dynamischen Verkehrsinformationen, bspw. Sperrungen, temporäre Sperrungen, Baustellen
- Berücksichtigung von Wetterverhältnissen, bspw. Nebel, Schneefall. Schlechte Wetterbedingungen verändern gewöhnlich nicht die Route, wohl aber die berechnete Fahrzeit (spätere Ausbaustufe).
- Berücksichtigung von hochdynamischen Verkehrsinformationen, bspw. Staus, stockender Verkehr (spätere Ausbaustufe)-
- Umkreis, in dem hochdynamischen Verkehrsinformationen Einfluss auf die berechnete Route haben sollen, wahlweise in Minuten oder Kilometer. Ein Stau, auf den das Fahrzeug erst nach 4 Stunden trifft, ist zumeist beim Eintreffen des Fahrzeugs nicht mehr vorhanden. Solche Staus werden dann bei der Berechnung zwar ausgewiesen, bleiben aber unberücksichtigt (spätere Ausbaustufe).
- Alternativrouting

Spezielle Einstellungen für Elektrofahrzeuge

- Batteriekapazität des Fahrzeugs in kWh
- Anfangsreichweite in Prozent der Batteriekapazität. Dieser Wert ist erforderlich, wenn ein Ziel nur erreicht werden kann. Die am Weg liegenden Ladesäulen werden dann anhand dieser Angabe berücksichtigt, so dass eine möglichst maximale Ausnutzung erfolgt
- „80%-Ladung“. Diese Angabe verkürzt die Reichweite des Fahrzeugs so, dass eine 80%-Ladung des Fahrzeugs ausreicht, um die nächste Ladesäule zu erreichen. Die 80%-Größe hat sich etabliert als Messwert für Schnellladezeiten.
- Reichweite in km bei 100% Batterieladestand. Das Routing erlaubt eine direkte Angabe, aber auch eine Auswahl des Fahrzeugtyps, dem dann ein Profil hinterlegt ist. Dieses Profil berücksichtigt Erfahrungswerte, die u.a. auch die Außentemperatur einbeziehen. In einer späteren Ausbaustufe ist auch die Berücksichtigung von Fahrerprofilen möglich.
- Zulässiger Umweg in km: wie weit darf eine Ladesäule maximal von der optimalen Route entfernt sein?
- "Ladesäule schnell ab" in kW: teilt Ladesäulen in "schnell" oder "langsam" ein, schnelle LS werden beim Routing bevorzugt
- Mehrfachauswahl für die Ladesäulentypen, die bei der Berechnung von Routenvorschlägen mit Zwischenladepausen berücksichtigt werden sollen:
 - CCS
 - CEE blau
 - CEE rot
 - CHAdeMO
 - Schuko
 - Tesla HPC
 - Tesla Supercharger
 - Typ1
 - Typ2

Funktionen, die mit „spätere Ausbaustufe“ gekennzeichnet sind, werden im Rahmen des Projekts entwickelt und nach und nach zur Verfügung gestellt.

3.3 Schnittstellen

3.3.1 Ortssuche

Es können die Adressdaten manuell eingegeben oder über eine XML Schnittstelle angesprochen werden.

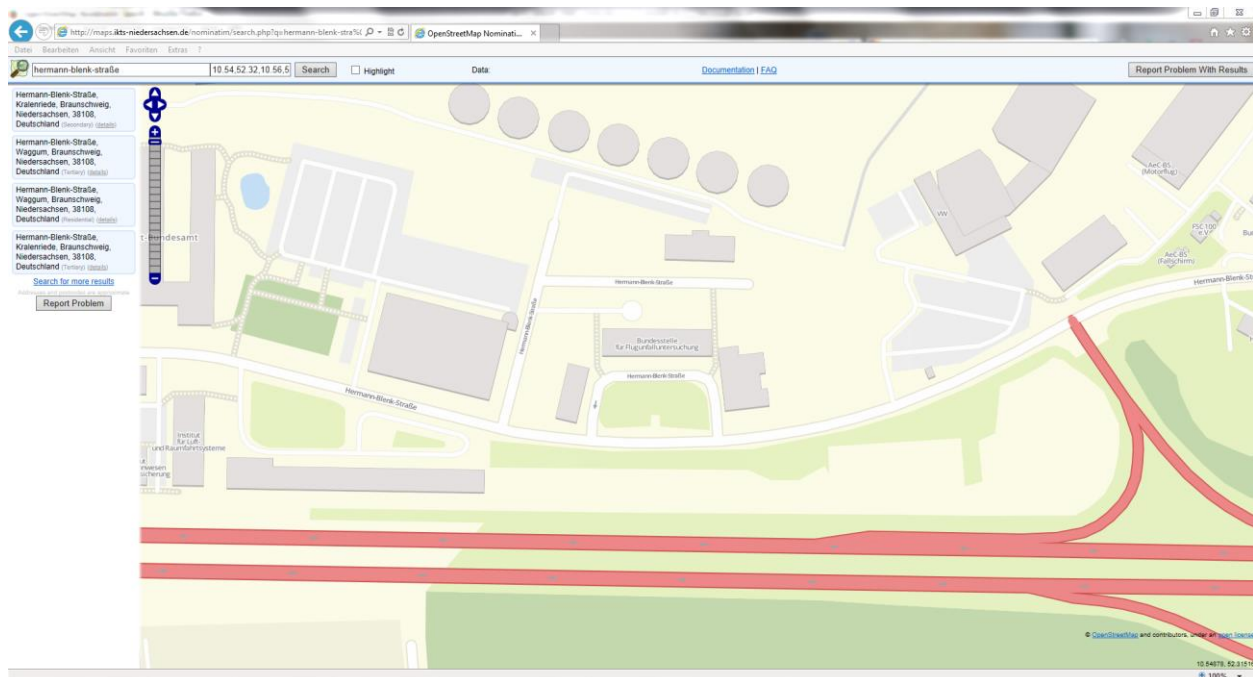


Abbildung 5: Adresssuche

Für die Ortssuche wird der Nominatim-Ansatz von OSM verwendet. Es kann über Namen und Adressen nach Objekten in der PostgreSQL-Datenbank gesucht werden um daraus die Georeferenzierung zu erhalten.

Um die Schnittstelle bedienen zu können, ist der Link zur Webseite um die Parameter „&format=xml“ zu ergänzen. An dem folgenden Aufruf ist eine beispielhafte Anfrage ersichtlich:

```
http://maps.ikts-niedersachsen.de/nominatim/search.php?q=hermann-blenk-stra%C3%9Fe&viewbox=9.82%2C52.34%2C9.84%2C52.33&format=xml
```

Als Rückgabe wird eine XML Liste mit den Antworten geliefert.

```
<?xml version="1.0" encoding="UTF-8" ?>
<searchresults timestamp='Tue, 24 Sep 13 15:16:04 +0200' attribution='Data © OpenStreetMap contributors, ODbL 1.0. http://www.openstreetmap.org/copyright' querystring='hermann-blenk-stra%C3%9Fe' viewbox='9.82,52.34,9.84,52.33' polygon='false' exclude_place_ids='12410198,12597470,5465725,12041311,3364660,11896192,11492290,12433272,12664060,12664066,5586237,12701543,12419267,11957423,3462986,12110745,12468868,3451440' more_url='http://maps.ikts-niedersachsen.de/nominatim/search?format=xml&exclude_place_ids=12410198,12597470,5465725,12041311,3364660,11896192,11492290,12433272,12664060,12664066,5586237,12701543,12419267,11957423,3462986,12110745,12468868,3451440&accept-language=de-DE&viewbox=9.82%2C52.34%2C9.84%2C52.33&q=hermann-blenk-stra%C3%9Fe'>
<place place_id='12410198' osm_type='way' osm_id='165484796' place_rank='26' boundingbox="52.3143877,52.3145746,10.5369954,10.5370852" lat='52.3144879' lon='10.5370852' display_name='Hermann-Blenk-Straße, Kralenriede, Braunschweig, Niedersachsen, 38108, Deutschland' class='highway' type='secondary' importance='0.6'/>
<place place_id='5465725' osm_type='way' osm_id='30622613' place_rank='26' boundingbox="52.3146245,52.3146283,10.5591037,10.5596932" lat='52.3146245' lon='10.5591037' display_name='Hermann-Blenk-Straße, Waggum, Braunschweig, Niedersachsen, 38108, Deutschland' class='highway' type='tertiary' importance='0.6'/>
<place place_id='3364660' osm_type='way' osm_id='4764256' place_rank='26' boundingbox="52.313771,52.3141021,10.5524758,10.5535226" lat='52.3141021' lon='10.5533501' display_name='Hermann-Blenk-Straße, Waggum, Braunschweig, Niedersachsen, 38108, Deutschland' class='highway' type='residential' importance='0.6'/>
<place place_id='11492290' osm_type='way' osm_id='151627452'
```



```
place_rank='26' boundingbox="52.3142924,52.3143877,10.5370484,10.5376601" lat='52.3143022'
lon='10.5373843' display_name='Hermann-Blenk-Straße, Kralenriede, Braunschweig,
Niedersachsen, 38108, Deutschland' class='highway' type='tertiary'
importance='0.6'/></searchresults>
```

Tabelle 1: XML Adressantwort

3.3.2 Routing

Für die Ansprache der Dienste ist eine Web-Service Schnittstelle vorhanden.

Eine der ersten Anforderungen ist, dass detaillierte Routinganweisungen (Routingverlauf, Geometrie Anweisungen) des Individualverkehrs (IV) übergeben werden können.

Die XML-basierte Schnittstelle ermöglicht die Kommunikation zu externen Geoinformationssystemen, wie dem OECON-Kartenservice auf TCP/IP Basis.

Für die Anfragen wird ein XML-Dokument mit GisRequest als Root-Element im POST-Body erwartet.

Zum aktuellen Zeitpunkt wird der „RouteProperties request“ und „RouteDescription request“ unterstützt. Der „GetLocation request“ ist in Verbindung mit dem vorhandenen Geocoding zu sehen.

3.3.2.1 RouteProperties request

Die RouteProperties Anfrage dient dazu, Informationen über Routen (Distanz, Dauer, Kosten) von einem Ort zu n anderen Standorten ("1: n-Routing") abzurufen.

Die Struktur der XML Anfragen ist wie folgt aufgebaut:

```
<RoutePropertiesRequest direction="..." dateTime="..." id="1">
  <Start>
    <Location x="..." y="..." coordType="..." type="..." name="..."
    id="..." />
  </Start>
  <Stop routingType="..." id="...">
    <LocationList>
      <Location type="..." id="..." x="..." y="..."
      coordType="..." />
    </LocationList>
  </Stop>
  <RoutingOptions>
    <Speed weight="..." />
    <Priorities levelOfDifficulty="..." powerEffort="..."
    bikePathUsage="..." />
  </RoutingOptions>
  <Parameter name="..." value="..." />
</RoutePropertiesRequest>
```

3.3.2.2 RouteDescription request

Die RouteDescription Anfrage hat den Zweck, Karten und textlichen Beschreibungen der Routen sowie Karten der Gebiete rund um einen bestimmten Ort abrufen.

3.3.2.3 GetLocation request

Die GetLocation Anfrage wird verwendet, um Koordinaten für eine gegebene Adresse zu bestimmen.

3.3.2.4 Schnittstelle zur POI Suche

Zusatzdaten mit punktuellm geographischem Bezug (POI) werden mittels GUI angezeigt und können über eine gesonderte Schnittstelle abgerufen werden. Im Bereich der Elektromobilität werden hierbei z.Z. Ladesäulen für E-Fahrzeuge angezeigt. In der Datenbank werden aktuell exemplarisch einige relevante Informationen zu den Ladesäulen abgelegt.

Eine mögliche Anfrage für POI ist z.B. wie folgt aufgebaut:

<http://maps.oecon-line.de/iktsnds/poi/nearby?lat=52.42&lng=10.78&radius=10000>

Dabei werden zu einer Position (Lat./Lon.) alle Ladesäulen im Umkreis von 10km ge-sucht.

Die Auflistung der Ergebnisse erfolgt dann in der dargestellten Form (Tabelle 2).

```
[
{"kind":"chargingpoint","name":"CCS 50kW DC Ladesäule","lat":52.416888,"lng":10.783317},
{"kind":"chargingpoint","name":"Ateliercafe","lat":52.42265,"lng":10.786546},
{"kind":"chargingpoint","name":"Gewerbe-Akademie","lat":52.429233,"lng":10.820146},
{"kind":"chargingpoint","name":"Autostadt","lat":52.43469,"lng":10.79846},
{"kind":"chargingpoint","name":"Wolfsburger Landstraße","lat":52.42234,"lng":10.7309},
{"kind":"chargingpoint","name":"E-Mobility-Station","lat":52.41667,"lng":10.78322},
{"kind":"chargingpoint","name":" Point S und Reifen
Henning","lat":52.421508,"lng":10.721444},
{"kind":"chargingpoint","name":"Hinterm Hagen 13","lat":52.416091,"lng":10.719497},
{"kind":"chargingpoint","name":"Heßlinger Str. 1-5","lat":52.428094,"lng":10.79138},
{"kind":"chargingpoint","name":"Major-Hirst-Straße 11","lat":52.424221,"lng":10.74362},
{"kind":"chargingpoint","name":"everynear","lat":52.4254703388844,"lng":10.7917801089483}
]
```

Tabelle 1 Standorte Ladesäulen

4 Zusammenspiel verschiedener Routing-Systeme

Ein Ziel des Projektes war es, Dienste im GAIA-X-Standard austauschbar und auswählbar zu machen. Der Nutzer sollte dann bei der Berechnung einer ODD-konformen Route die Möglichkeit haben, aus verfügbaren Diensten jenen auszusuchen, der seinen Anforderungen am besten entspricht. Der OECON-Dienst sollte dazu bspw. das besonders auf Elektrofahrzeuge ausgerichtete Routing hervorheben, während andere Dienste ihre speziellen Besonderheiten darstellen. Ursprünglich wurde auch eine Kostenfunktion diskutiert, allerdings ist Routenplanung als Basisfunktion inzwischen omnipräsent, ohne dass Nutzer dafür speziell zahlen. Natürlich erfolgen Zahlungen auf indirektem Wege, bspw. durch die Kenntnisnahme von Werbung, dem Kauf eines Smartphones oder auch durch Daten, die der Anbieter zur Optimierung von Werbung nutzen kann. Dies alles ist aber im Rahmen der GAIA-X-Dienste irrelevant und kann auch nicht abgebildet werden. Daher haben sich die Partner im Rahmen des Projekts darauf verständigt, dass lediglich die Routingfunktion selbst als umschaltbarer Dienst angeboten wird.



Zum Zeitpunkt des Projektstarts war daher zunächst festzulegen, welches Grundformat die Dienste unterstützen müssen, um in die Auswahl für den Nutzer eingebunden zu werden. Die Unterstützung unterschiedlicher Eingabeformate würde eine ständige Anpassung und Erweiterung der Clientsoftware im Fahrzeug erfordern und wurde daher als unpraktikabel verworfen.

Um die Problematiken rund um die Routenplanung und die Integration von ODD-Attributen dem Projektteam zu verdeutlichen, wurde außerdem ein Dokument erstellt, in dem die technischen Grundlagen beschrieben wurden. Die folgenden Abschnitte entstammen teilweise diesem Projektdokument.

OECON leitete bis März 2023 die Arbeitsgruppe „TP4 ODD-Routing“, die einmal pro Woche am Mittwoch für eine Stunde zusammenkam, um die Fortschritte der Arbeiten zu diskutieren und die weiteren Ziele festzulegen. Seitdem wurden die Arbeiten wieder in einer gemeinsamen Arbeitsgruppe zusammengeführt. Der Schwerpunkt lag zunächst auf der Umsetzung des Minimum Viable Demonstrator (MVD). OECON hatte hierzu zusätzliche Schnittstellen geschaffen, mit deren Hilfe ODDs an das Routing zur weiteren Verarbeitung übergeben werden. Diese können routen-relevante Informationen enthalten, bspw. Sperrungen, Baustellen oder auch Wetterinformationen. Um die Arbeiten auch visuell darstellen zu können, wurde ein Frontend geschaffen, das es erlaubt, die entwickelten Funktionen auch ohne Verbindung zu den anderen Komponenten (ODD Matching und Reaktionsplaner) zu testen und zu demonstrieren. Dieses Frontend wurde auch im Rahmen des MVD zum Einsatz kommen – dort allerdings inklusive Verbindung zu den anderen Diensten. Es ist in Kapitel 5.6 beschrieben.

4.1 Informationen zum Aufbau der Datenbanken für Straßen und Routing

Die Darstellung eines Straßennetzes in digitaler Form erfolgt gewöhnlich in Form von einzelnen Segmenten, die über Knoten miteinander verbunden werden. Da Straßen sich gewöhnlich nicht wie Gerade verhalten, sondern Kurvenform besitzen, besitzen die meisten Segmente nur Knoten mit je einem weiteren Nachbarn. Die einzelnen Segmente bilden dann die Kurvenform nach. Lediglich jene Segmente, die direkt an Kreuzungen grenzen, haben an ihren Enden mehr als einen Nachbarn.

Straßennetze können gewöhnlich nicht komplett planar dargestellt werden, da einzelne Segmente (auch als Kanten bezeichnet) sich kreuzen können aufgrund von Brücken-konstruktionen. Es entsteht daher ein 3D-Graph, dessen dritte Dimension allerdings nur schwach besetzt ist.

Jedes Segment besitzt mehrere Attribute. Die Klassifizierung beschreibt die visuelle Darstellung, damit bspw. alle Autobahnen in der Kartendarstellung gleich aussehen. Ein weiteres beschreibt die interne Typisierung, die festlegt, wie schnell man auf dem aktuellen Segment tatsächlich vorankommt. Hinzu kommt ein Fahrtrichtungsattribut – bspw. dürfen Einbahnstraßen und Autobahnen nur in einer Richtung durchfahren werden. Hinzu kommen noch die physische Ausdehnung, textuelle Beschreibungen für Segmente und Knoten und natürlich die geographischen Koordinaten.

Die Knoten besitzen keine Klassifizierung, sie enthalten aber dennoch wesentliche Informationen für das Routing. Beispielsweise enthalten sie Abbiegeverbote, also Informationen, ob bspw. an Kreuzungen ein Linksabbiegen wegen des Vorhandenseins eines entsprechenden Segments prinzipiell möglich wäre, aber aufgrund eines Verkehrszeichens untersagt ist.

Aus den Basisdaten werden gewöhnlich einerseits die visuellen Straßendaten generiert – diese Datenbank ist nur zur Darstellung der Karte auf einem Bildschirm geeignet und ist oftmals in Form von rechteckigen Arealen gespeichert, die wiederum in unterschiedlichen Auflösungen vorliegen. Damit kann eine Gesamtübersicht der Karte sehr schnell und ohne Details dargestellt werden, während

Details nur in höheren Zoomstufen dargestellt werden. Ein Beispiel hierfür ist das standardisierte PSF³.

Zum anderen wird ein aus Kanten und Knoten ein Graph gebildet, der für die Routenberechnung geeignet ist. Dieser enthält gewöhnlich keine visuellen Attribute, ist dafür aber als einheitliche spatiale Datenbank angelegt. Um die Routenberechnung zu beschleunigen, werden heutzutage dank bezahlbarem Speicherplatz Routen vorberechnet und ab-gespeichert, meist nach dem Contraction-Hierachies-Modell. Das erlaubt einen extrem schnellen Zugriff auf Routen. Diese Dateien sind meist sehr groß, enthalten aber keine dynamischen Informationen zu Verkehrszuständen, sofern sie nicht ständig aktualisiert werden.

4.2 Spezialitäten der Datengrundlagen

In den 90er Jahren kristallisierten sich aus einer Vielzahl von Datenanbietern die Firmen NavTech (später NavTeq, seit 2011 HERE) und TeleAtlas (seit 2011 TomTom) heraus. Beide boten ab 1996 ein hausnummerngenaues Routing und vollständige Straßennetze für immer mehr Städte Europas an. Bis dahin kannte man digitale Karten nur in grober Auflösung und mit Informationen bis hin zu größeren Stadtstraßen. Beide Firmen wendeten für die Darstellung der Karten auf dem Bildschirm unterschiedliche Verfahren an, so dass die digitalen Karten der Hersteller nicht deckungsgleich waren.

NavTech nutzte für die Darstellung von Kurven eine Aufteilung in Einzelsegmente mit einer Mindestlänge von 10 Metern. Besonders bei Autobahnzufahrten fällt das auf – die gebogenen Straßen sind dort bei hoher Zoomtiefe als Aneinanderreihung einzelner Segmente mit erkennbaren Winkeln dargestellt. TeleAtlas verwendete stattdessen ein mathematisches Verfahren, das Anfangs- und Endpunkt durch ein berechnetes Polygon verbindet. Daher sahen dieselben Autobahnauffahrten hier „realer“ aus – aufgrund der Berechnung sind sie es aber nicht. Beide Verfahren führten bei der Weiterentwicklung der Routenplanung hin zu Navigationssystemen mit ihrer GPS-Genauigkeit zu Problemen. Heute sind die jeweiligen Verfahren präziser, aber nach wie vor gibt es die Unterscheidung zwischen den beiden Darstellungsformen.

Mehrspurige Straßen (bspw. Autobahnen) werden bei der Erzeugung der visuellen Datenbanken bewusst entzerrt – der Abstand zwischen Spur und Gegenseite ist in den Daten gewöhnlich größer als in der Realität, damit sie auch in kleineren Zoomstufen noch als mehrspurige Autobahnen erkennbar bleiben. Dies hat Einflüsse auf die Darstellung bzw. Anbindung von GPS-genauen Daten, wie sie Navigationssystemen zur Verfügung stehen – das Fahrzeug steht dann bei hohen Zoomstufen neben der Fahrbahn. Dies wird gewöhnlich durch visuelle Tricks gelöst – da Fahrzeuge nicht „neben“ der Fahrbahn fahren können, stellt man sie visuell verschoben auf der Straße dar. Besonders beim Start der Navigation stellt dies eine Herausforderung dar, weil der Startpunkt gewöhnlich nicht auf einem Segment liegt, sondern irgendwo daneben. Es wird dann eine Anbindung durch eine Lotfällung erreicht – dies kann insbesondere in innerstädtischen Bereichen zu falschen Anbindungen führen. Aber auch an Autobahnzufahrten treten solche Anbindungs-fehler gehäuft auf, weil oftmals die Auf- und Abfahrten extrem nah an der Autobahn selbst liegen.

Mit dem Übergang auf hochauflösende Karten für automatisiertes Fahren und einer deutlich genaueren Positionsbestimmung durch die Nutzung mehrerer GNSS-Dienste (GPS, Galileo, Glonass) gehören die genannten Effekte allerdings zunehmend der Vergangenheit an.

Neben der reinen geographischen Positionierung spielt auch die Attributierung der Straßen eine wesentliche Rolle. Alle bekannten Datengrundlagen unterscheiden die Straßen auf zweierlei Weisen: nach der Art der Darstellung und nach der Art des Straßentyps. Die Darstellung ist tatsächlich nur für Kartenansichten relevant. Sie sorgt dafür, dass Auto-bahnen, Bundesstraßen, Wohnstraßen etc. immer als solche auf einer Karte dargestellt werden, unabhängig davon, wie schnell sie tatsächlich befahren

³ <https://www.iso.org/obp/ui/en/#iso:std:iso:ts:20452:ed-1:v1:en>



werden können. Der Straßentyp hingegen ist relevant für die Routenberechnung, denn er fügt jedem Segment eine Typisierung hinzu, die angibt, wie schnell auf dieser Straße gefahren werden kann. Zusammen mit der Länge des Segments ergibt dies die grundsätzlichen „Kosten“. Je nach Wunsch der Routenoptimierung (schnellste, kürzeste, kostenoptimale Route) erhalten wir dann unterschiedliche Routen. Zusätzliche Attribute (beispielsweise „für au-tomatisiertes Fahren geeignet“, „mautpflichtig“, „gesperrt für Fahrzeuge des Typs X“) können die berechneten Kosten zusätzlich beeinflussen.

Manche Kartensysteme fügen diesen beiden Bewertungen der Segmente noch ein drittes hinzu, dass ebenfalls nur die Darstellung der Karte beeinflusst: ein Attribut der Vollständigkeit. Je nach Zoomstufe werden Karten nur mit Autobahnnetz, mit zusätzlichem Bundesstraßennetz, mit Land- und Stadtstraßennetz gezeichnet. Um nun auch bei geringen Zoomstufen eine vollständige Verbindung von A nach B darstellen zu können, werden bspw. einige Bundesstraßen, die als Verbindung zwischen Autobahnen dienen, bereits dann gezeichnet, wenn eigentlich nur ein Autobahnnetz dargestellt werden soll. Ansonsten könnten in der Kartendarstellung Lücken entstehen, die als fehlende Verbindungsmöglichkeiten interpretiert werden könnten. Auf das reine Routing hat dieses Attribut keinen Einfluss.

4.3 Daten-Eingaben für das Routing

4.3.1 Start- und Zielkoordinaten

Um eine Route zu berechnen, benötigt man mindestens zwei Koordinaten: den Startpunkt und den Zielpunkt. Bei Navigationssystem ist gewöhnlich nur die Eingabe des Ziels erforderlich; als Startpunkt wird der aktuelle Standort angenommen. Bei A-priori-Berechnungen liegt ein solcher Startort gewöhnlich nicht vor, so dass er ebenfalls eingegeben werden muss.

Die interne Formatierung der Koordinaten der zugrundeliegenden Karte kann unterschiedlichen Systemen folgen. Üblich sind geographische Koordinaten im WGS-84-System (zumeist ungerechnet in interne 32- oder 64-Bit-Werte) oder das kachelbasierte UTM, das eine längentreue Darstellung bietet und daher eine besonders einfache Berechnung von Entfernungen erlaubt. Eingabewerte für Koordinaten liegen allerdings gewöhnlich nicht als Koordinaten in diesen Formaten vor; lediglich Satellitenkoordinaten aus GNSS-Systemen liefern WGS84-Koordinaten.

Üblicherweise kommt daher eine Georeferenzierung zum Einsatz, vereinfachend auch „Ortssuche“ oder „Geocoding“ genannt. Dabei werden beschreibende Werte eingegeben, zumeist Ort, Straße und Hausnummer. Eine Datenbanksuche liefert dann die Orte in der Datenbanktabelle, die zu den eingegebenen Werten passen. Hinter diesen Orten liegen dann die passenden Koordinaten, die für das Routing als Ausgangsbasis verwendet werden können.

Viele Navi-Systeme geben den aktuellen Standort in lesbarer Form aus, also als Beschreibung von Ort, Straße und Hausnummer. Dem zugrunde liegt die „Rückwärtssuche“, bei der den bekannten Koordinaten der jeweils nächste Ort aus der Ortsdatenbank zugeordnet wird. Diese Funktion wird auch als „Reverse Geocoding“ genannt. Sie ist für ein Routing nicht zwingend erforderlich.

4.3.2 Zwischenstationen und „Travelling Salesman“

Eine „Zwischenstation“ beschreibt eine Position, die beim Routing tatsächlich angefahren werden soll. Die Route führt also zu dieser Position und dann weiter zur nächsten Zwischenstation bzw. zum Ziel. Im Hintergrund wird dies einfach dadurch gelöst, dass mehrere Routen nacheinander berechnet werden, die dann zusammengesetzt werden.

Komplexer und anspruchsvoller ist das „Travelling Salesman“-Problem, bei dem mehrere anzufahrende Stationen nach der optimalen Route angeordnet werden sollen. Seinen Namen hat

dieses Problem nach dem den reisenden Vertreter, der seine Kunden in optimaler Weise besuchen möchte. Dabei ist es erlaubt und zumeist auch erforderlich, die ursprünglich eingegebene Reihenfolge der Stationen zu tauschen. Mathematisch handelt es sich dabei um ein NP-schweres Problem⁴. Die Routenberechnung erfolgt dabei über alle Wege, die zwischen den Stationen möglich sind. Aufgrund der Nutzung vorberechneter Routen durch Nutzung des Contraction-Hierarchies-Modells sind solche Rechnungen inzwischen in kürzester Zeit zu erstellen.

Zusätzliche Anforderungen können entstehen durch das Einfügen von Pausen und die Beachtung erlaubter Fahrzeiten pro Tag, die Übernachtungen erforderlich machen können.

4.3.3 Korridore und „Über“-Stationen

Das Routing kennt auch „unechte“ Zwischenstationen, um bestimmte Wegführungen zu erzwingen. Als Beispiel mögen drei Wege von Karlsruhe nach Berlin dienen, die allesamt mit ca. 720km ungefähr gleichlang sind, jedoch weitgehend unterschiedlich sind:

- Karlsruhe-Frankfurt-Braunschweig-Magdeburg-Berlin über A5, A7 und A2
- Karlsruhe-Frankfurt-Alsfeld-Erfurt-Berlin über A5, A4 und A9
- Karlsruhe-Walldorf-Heilbronn-Nürnberg-Berlin über A6, A81 und A9

Um nun eine dieser Routen zu „erzwingen“, kann man eine „Über-Station“ festlegen; im ersten Falle Braunschweig, im zweiten Fall Erfurt und im dritten Fall Nürnberg. Diese Positionen sind dann keine echten Stationen, sondern werden als kreisförmiges Gebiet interpretiert, durch das die Route führen muss, ohne dass dort angehalten wird.

Eine Erweiterung dieser Funktion stellt der „Korridor“ dar, bei dem über weite Strecken vorgegeben wird's, wie ungefähr die Route verlaufen soll. Dieses Verfahren kommt oft zum Einsatz, wenn Routen über nicht miteinander verwandte Kartensysteme hinweg ausgetauscht werden sollen. Wenn der Korridor schmal genug gewählt wird, kann man davon ausgehen, dass das Zielsystem dieselbe Route berechnet wie das Ausgangssystem, auch wenn die Segmentaufteilung der beiden Kartensysteme unterschiedlich ist.

Korridore können bspw. gebildet werden, indem ein Lot auf die Anfangs- und Endpunkte der Segmente der berechneten Route gebildet wird. Die neuen Punkte des Korridors werden dann durch eine Verschiebung entlang des Lots gebildet. Ein solcher Korridor sollte breit genug sein, um etwaige Differenzen in der Darstellung zu überbrücken (siehe hierzu Abschnitt 3.2), aber auch schmal genug, damit nicht andere, alternative Möglichkeiten die gewünschte Route ersetzen können. Bei aktuellen Kartensystemen genügt eine Breite von ca. 100 Metern.

4.4 Daten-Ausgabe des Routings

Nachdem aus Start- Ziel- und ggf. Zwischenstationen eine Route berechnet wurde, steht diese sowohl in visueller als auch in textueller Form zur Verfügung. Die Routingalgorithmen liefern dabei zunächst eine komplette Routenbeschreibung, die alle Segmente und alle Knoten enthält, die die Route umfasst. Für eine grafische Darstellung der Route sind diese Angaben erforderlich. Eine textuelle Ausgabe oder eine Routenführung per Navigation erfordert allerdings eine Priorisierung, die nach der eigentlichen Routenberechnung durchgeführt wird. Dabei werden die erforderlichen Fahrmanöver nach ihrer Relevanz gewichtet. Wichtige Manöver sind dabei:

- Abzweigungen, die zum Verlassen des aktuellen Weges führen. Dies sind die typischen Abbiegemanöver. Auch Kreisel und Autobahnzufahrten fallen in diese Kategorie.

⁴ [https://de.wikipedia.org/wiki/NP_\(Komplexit%C3%A4tsklasse\)](https://de.wikipedia.org/wiki/NP_(Komplexit%C3%A4tsklasse))

- Geradeausfahrten, wenn der Name der Route sich ändert, beispielsweise bei Autobahnen, deren Änderung dann mit einem “Weiter auf A7” angekündigt werden. Auch im innerstädtischen Bereich kann sich der Name längerer Straßen zwischendurch ändern.
- Geradeausfahrten, wenn es mehrere Möglichkeiten gibt, beispielsweise an Kreuzungen, bei denen zusätzlich zum schnelleren Befahren auch eine Untertunnelung vorhanden ist.

Alle Knoten mit nur zwei Segmenten ohne weitere Änderung – also reine Verbindungs-knoten – sollen nicht zu Ansagen bzw. Routenempfehlungen führen.

Was in der Theorie sehr einfach klingt, ist in der Praxis allerdings nicht immer eindeutig. Jeder, der schon einmal ein Navigationssystem benutzt hat, kennt Situationen, an denen eine Ansage fehlt, nicht eindeutig oder sogar falsch war. Die Ursachen hierfür können vielfältig sein; zumeist ist allerdings eine falsche Priorisierung der Grund für ein solches Verhalten, die allerdings wiederum zumeist in Fehlern des zugrundeliegenden Kartematerials begründet ist.

Textuelle Angaben des Routings enthalten heute auch zumeist zusätzliche Richtungsinformationen, die inzwischen auf den Daten beruhen, die sich auch auf Schildern wieder-finden, während sie in der Anfangszeit der Routenplanung noch automatisiert aus den Routingdaten selbst erzeugt wurden.

4.5 Übergabe von berechneten Routen

4.5.1 Unterschiedliche Projektionen

In der realen Welt hat ein Punkt eine eindeutige physische Koordinate, die lediglich durch unterschiedliche Koordinatensysteme beschrieben werden kann. Gleiches gilt für physische Straßen, die in den einzelnen Koordinatensystemen als Polygone, also eine Folge von Punkten mit Start- und Endpunkt beschrieben werden können.

Möchte man nun dreidimensionale Positionen in 2D-Karten darstellen, hat man es nicht mehr mit einer einfachen Koordinatentransformation zu tun, sondern mit einer Projektion: dreidimensionale Koordinaten werden auf eine Fläche abgebildet. Nun befinden sich Straßen zumeist auf der Oberfläche der Erdkugel, aber dennoch gehen bei der Projektion Inhalte wie die Höhe der Straße und ihre Steigung verloren. In neueren Datenbeständen werden Höheninformationen als zusätzliches Attribut hinzugefügt; sie werden aber gewöhnlich nicht graphisch dargestellt und fließen auch nur indirekt in die Berechnung einer Route ein, indem Steigungen ein geringeres Geschwindigkeitsprofil zugeordnet wird.

4.5.2 Deckung und Klassifizierung

Allein die Projektion stellt allerdings noch keine größere Herausforderung für den Übergang von einem zum anderen Kartensystem dar. Problematischer ist die fehlende Deckungsgleichheit der Karten. Diese kommt einerseits daher, dass reale Kurven in digitale Polygonzüge überführt werden müssen, um die Anzahl der Knoten in einem sinnvollen Rahmen zu halten. Zum anderen sind optische Verschiebungen üblich, um beispielsweise getrennte Fahrbahnen sichtbar darstellen zu können. Zum dritten hängt die Wahl der Knoten von der Digitalisierungsmethode ab – sie sind also bei verschiedenen Kartensystemen nicht automatisch deckungsgleich.

All diese Eigenschaften führen dazu, dass eine in Kartensystem A vorberechnete Route nicht automatisch zu Kartensystem B passt. Nun könnte man argumentieren, dass die exakte Route auch gar nicht das Ziel der Übertragung ist, sondern das Befahren der realen Straße, die ja Grundlage für Kartensystem A und B ist. Hier kommt nun ein weiterer Unterschied bei der Erzeugung von Kartensystemen zur Geltung – deren Typisierung. Üblicherweise kategorisiert man Straßen nach verschiedenen Klassen. Die auf NavTeq basierenden Kartensysteme kennen 12, die auf TeleAtlas-



Daten basierenden kennen 16 verschiedene Klassen, die bei einer Wegberechnung mit unterschiedlichen "Kosten" behaftet sind. Die Klassifizierung erfolgt dabei über den Typ der Straße und die darauf zu fahrende mittlere Geschwindigkeit von der "langsamen Stadtstraße" im Wohngebiet bis zur "schnellen Autobahn" ohne Geschwindigkeitsbegrenzung.

Diese Klassifizierungen sind ebenfalls nicht identisch, weshalb Routenberechnungen bei den unterschiedlichen Kartensystemen zu unterschiedlichen Ergebnissen führen können. Im Falle von ODD-konformen Routen und der Möglichkeit, Alternativrouten auszuwählen, kommt erschwerend hinzu, dass die gewählte Route möglicherweise mit keiner übereinstimmt, die das Navigationssystem im Fahrzeug berechnen würde, wenn man ihm lediglich die Koordinate des Ziels übergibt.

4.5.3 Übergabe von Routen im Projekt

Es ist daher erforderlich, die berechnete Route als Ganzes so detailliert wie möglich zu übergeben, damit das Zielsystem diese exakt so abfährt wie gewünscht. Im Projekt wählen wir dazu die Übergabe der Route in Form von einzelnen Polygonzugpunkten, die die Route im Kartenformat des Ausgangssystems beschreiben. Wie bereits erwähnt, müssen diese nicht deckungsgleich sein mit dem Kartenformat des Zielsystems. Es darf aber davon ausgegangen werden, dass das Zielsystem dieselben realen Straßen kennt wie das Ausgangssystem und es daher möglich ist, auf Basis einer Suche nach dem jeweils nächsten Segment im Zielsystem eine Route zusammenzustellen, die mit der des Ausgangssystems identisch ist.

Es sei darauf hingewiesen, dass aktuell verfügbare Navigationssysteme für Fahrzeuge keine passende Schnittstelle bereithalten. Frühe Versuche solcher Schnittstellen aus 2001 bis 2005 wurden aufgegeben, da sie de facto nirgends eingesetzt wurden. Innerhalb des Projekts musste daher ein modifiziertes Navigationssystem eingesetzt werden, das eine solche Übergabeschnittstelle bietet.

4.5.4 Navigation im Fahrzeug

Die für eine Fahrerinformation benötigten Ansagen ("In 200m rechts abbiegen") können dabei wahlweise aus dem Kartensystem im Fahrzeug oder aus dem Kartensystem des Routings stammen. Dazu wird das Polygonformat erweitert um Angaben zu wichtigen Knoten und der jeweiligen zu befolgenden Anweisung, sofern das Kartensystem des Routings dies unterstützt. Nähert sich das Fahrzeug einer solchen Anweisung, wird ein Hinweis an den Fahrer ausgegeben.

Im Falle einer Störung auf dem Weg können das System im Fahrzeug und das Routing-system aufeinander reagieren. Erfährt das zentrale Routingsystem von Störungen, kann es das Fahrzeug über eine geänderte Route informieren. Stellt das Fahrzeug ein Problem fest, das zentral nicht bekannt ist, kann ebenfalls eine neue Route angefordert werden. Zum dritten besteht aber auch die Möglichkeit, eine Alternativroute am Fahrzeugsystem ohne Kommunikation mit dem zentralen Routingsystem zu berechnen. Es ist dann Aufgabe des Fahrzeugs, eine ODD-Kompatibilität bzw. Beachtung sicherzustellen.

4.6 Verkehrsinformationen und ODD

Der eingebrachte Routenplaner konnte bezüglich des Straßennetzes nur statische Informationen verarbeiten, also Sperrungen, Baustellen usw. Dynamische und hochdynamische Daten wie aktuelle Verkehrsinformationen blieben außen vor. Im Projekt war jedoch genau dies verlangt. Um nun die Problematik zu verstehen, muss man erklären, wie das System von Contraction Hierarchies funktioniert.

Die räumlichen Straßensegmente sind dort in Kacheln abgelegt. Im kleinsten Bereich haben diese Kacheln eine Seitenbreite von ein paar Hundert Metern, je weiter man sich aus der Karte herausbewegt, desto größer werden die Kacheln, wobei die Seitenlängen je um den Faktor 2



wachsen. Sieht man ganz Deutschland, kommen dennoch nur etwa genauso viele Kacheln zur Anzeige wie wenn man ins Detail geht und die heimische Wohnstraße betrachtet.

Die betrifft aber nur die visuelle Darstellung. Im Falle der Routenplanung werden ausgehend von den großen Kacheln Zug um Zug auch die kleineren Kacheln genutzt, um am Ende eine bis ins Detail korrekte Route zu haben. Kacheln, die nicht Teil der Route sind oder deren „Kosten“ zu groß sind, werden dabei frühzeitig verworfen. Beispielsweise macht es üblicherweise keinen Sinn, hunderte von Kilometern zunächst in die Gegenrichtung des Ziels zu navigieren.

Um nun die hohe Geschwindigkeit zu erklären, die Contraction Hierarchies muss man einerseits beachten, dass Speicherplatz im Bereich von mehreren Gigabytes heute kein Problem mehr darstellt und andererseits Zugriffe auf Massenspeicher im Zeitalter von SSDs auch kein wesentlicher Zeitfaktor mehr sind. Kurz gesagt: Contraction Hierarchies führen eine Vorberechnung aus, und zwar von jedem Segmentknoten zu jedem anderen. Die Rechnung findet also nicht mehr während der eigentlichen Routenplanung statt, sondern vorher. Während der Planung werden nur noch Datenbankoperationen und Kostenberechnungen ausgeführt. Deshalb ist das Ergebnis de facto auch gefühlt *sofort* verfügbar.

Und hier kommen nun die Verkehrsinformationen ins Spiel. Eine solche Information verändert den Status eines Segments. Ein Stau bedeutet eine höhere Laufzeit auf einem Segment. Eine Sperrung macht es vollständig undurchlässig. Eine Baustelle bewirkt ebenfalls Verzögerungen, aber zumeist geringere als ein Stau. Will man diese Informationen in ein kachelbasiertes System einbringen, dann müssen alle betroffenen Kacheln neu berechnet werden – und zwar jedes Mal, wenn sich Verkehrsinformationen ändern. Die Originalkacheln werden als ungültig markiert, sie können wieder zum Einsatz kommen, wenn die Störung behoben ist.

Dies war auch der Grund, bei Beantragung des Projekts eine größere Menge Server vorzusehen – zwei für die eigentliche Routen-„Berechnung“ und sechs weitere für die Hintergrundaktualisierung der Routen bei jeder neuen Verkehrsinformation (siehe hierzu auch Kapitel 4.7).

Das gleiche Verfahren wurde dann auch für die Berechnung ODD-konformer Routen eingesetzt. Auch hier erhält das entsprechende Segment eine Information, inwieweit es der ODD entspricht – und kann modifiziert werden, wenn sich daran etwas ändert. Der zweite Demonstrator (siehe Kapitel 5.6) zeigt dieses Verhalten – dort wird zwischenzeitlich ein Segment als nicht ODD-konform dargestellt, um dann zu zeigen, dass sofort eine neue, ODD-konforme Route berechnet und dem Fahrer mitgeteilt wird.

4.7 Modifikation zur Berechnung ODD-konformer Routen

Im späten Verlauf des Projekts stellte sich heraus, dass zwar die Aufgabenstellungen der Projektbeschreibung komplett erfüllt waren, eine wichtige Funktion aber nicht abgebildet werden konnte.

Grundsätzlich geht man bei einer Routenplanung davon aus, dass immer eine Route von A nach B gefunden werden kann. Es sind ein paar Ausnahmen möglich, bspw. wenn das Ziel in einem gesperrten Bereich liegt oder zwischen Start und Ziel auf Verkehrssysteme zurückgegriffen werden muss, die nicht Teil des Systems sind, beispielsweise Kontinent-übergreifende Reisen. Die Fa. Google behalf sich eine Weile lang damit, in solchen Fällen „Schwimmen“ zu empfehlen (bspw. von Hamburg nach New York“), was aber eher in den Bereich des Skurrilen gehört und hier nicht weiter erörtert werden soll.

Im Falle der ODD-konformen Routen war vorgesehen gewesen, dass Routenergebnisse mitteilen sollten, zu wieviel Prozent sie ODD-konform sind. Üblicherweise gibt es bei Start und Ziel immer ein paar Segmente, die typischerweise nicht ODD-konform sind, bspw. Wohnstraßen. Dies ist vom

Fahrzeug aus aber nicht konfigurierbar, und es war weniger nützlich zu erfahren, welche Route zu wieviel Prozent ODD-konform war, wobei 100% so gut wie nie erreichbar waren.

Stattdessen benötigte man eine andere Fragstellung, nämlich: „Berechne mir Routenvorschläge, die zu mindestens x Prozent ODD-konform sind“, wobei das x vom Benutzer festgelegt werden kann. Diese Frage ist komplett anders als übliche Anforderungen an Routenplaner, wo es eher darum geht, bestimmte Routen zu vermeiden, beispielsweise keine Autobahnen, keine Mautstraßen usw. Das Vermeiden ist einfach zu realisieren, weil es für alle gekennzeichneten Segmente gleichermaßen gilt: diese erhalten dann einen Kostenmalus und fallen daher zumeist aus der Rechnung heraus. Nur in dem speziellen Fall, dass kein anderer Weg möglich ist, kommen sie dennoch zum Einsatz. Ein Beispiel hierfür ist das Kleinwalsertal, eine österreichische Enklave, die nur über eine mautbehaftete Straße aus Deutschland erreichbar ist.

Es wurde daher eine Funktion benötigt, die ODD-konforme Segmente bevorzugt und diese summiert, so dass sie mindestens x Prozent der gesamten Strecke stellen. Dies ist ein rekursiver Ansatz, der nicht allein durch die Kosten pro Segment berechnet werden kann.

Im Rahmen des Projekts war für die Integration dynamischer Verkehrsinformationen ursprünglich eine Summe von acht Servern vorgesehen. Es zeigte sich im Rahmen des Projekts, dass einerseits die Hardware im Jahre 2022 deutlich potenter war als jene, die 2012 bis 2015 zum Einsatz kam, und dass andererseits auch Softwareoptimierungen durch die Umstellung auf aktuelle Frameworks und Datenbanken einen erheblichen Einfluss auf die erzielbare Rechengeschwindigkeit hatten. Daher konnte die Zahl der gemieteten Server deutlich reduziert werden; die freigewordenen Gelder wurden stattdessen in die Lösung dieser zusätzlichen Aufgabe gesteckt, so dass durch eine kostenneutrale Verlängerung des Projekts um 6 Monate auch hier eine vollständig zufriedenstellende Lösung gefunden werden konnte.

Der Einfachheit halber werden für die im Projekt zu verwendenden Formate der Schnittstellen bereits bekannte Formate benutzt:

- Geographisches Format: Alle Koordinaten werden im geographischen Lat-Long-System übergeben.
- Ortssuche: Formatbeschreibung nach Abschnitt 3.3.1: Die Ortssuche nimmt als Eingangswert verschiedene textuelle Daten an (Ort, Straßename, Hausnummer oder POI-Name) und liefert eine Liste mit gefundenen Ergebnissen in textueller Form zurück. Die Ergebnisse enthalten auch eine Koordinate als Eingangswert zur weiteren Verarbeitung.
- Hinweis: nicht alle Routingsysteme innerhalb des Projekts unterstützen eine Ortssuche.
- Routing-Eingangsformat: Formatbeschreibung nach Abschnitt 4.3: Es werden Start-, Ziel und ggf. Zwischenstationen mit geographischen Koordinaten übergeben. Das Routing generiert daraus und aus den vorgegebenen Parametern (ODD, Optimierung, Optionen für Elektrofahrzeuge...) eine oder mehrere alternative Routen.
- Routing-Ausgabeformat: json-Datei, die die Polygone einer Route in Lat-Long-Koordinaten enthält sowie optional eine Liste der wichtigen Punkte, an denen neue Weginformationen stehen ("rechts abbiegen"):

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "geometry": {
        "type": "LineString",
        "coordinates": [
```



```

    [
      10.491174618820361,
      52.265478390583375
    ],
    [
      10.49137958492442,
      52.265892042202324
    ],
    [
      10.491637002489798,
      52.26635956614523
    ],
    [
      10.491456884699842,
      52.266874401275984
    ],
    [
      10.491138744901242,
      52.267508259434436
    ],
    ....
    [
      10.938394314163796,
      52.24557728869268
    ]
  ]
},
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      10.491174618820361,
      52.265478390583375
    ]
  },
  "properties": {
    "instruction": "Geradeaus auf Feldstraße",
    "distanceMeters": 103,
    "timeMillis": 14820
  }
},
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      10.491637002489798,
      52.26635956614523
    ]
  },
  "properties": {
    "instruction": "Links abbiegen auf Amselstraße",
    "distanceMeters": 132,
    "timeMillis": 19047
  }
},
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      10.491138744901242,

```



```

    52.267508259434436
  ]
},
"properties":{
  "instruction":"Scharf rechts abbiegen auf Vogelsang",
  "distanceMeters":331,
  "timeMillis":47672
}
},
...
],
"status":"OK, 40 km, 30 Minuten"
}

```

Das ursprünglich geplante Auswahlverfahren für Routingsysteme – nicht Teil der Aufgaben von OECON - wurde nicht implementiert, weil das zweite Routingsystem der Fa. Meraner sich zu diesem Zeitpunkt in einem zu frühen Entwicklungsstadium befand und nicht damit gerechnet wurde, es zum Einsatz des MVD nutzen zu können. Da nun das Datenformat von OECON als allgemeines Eingabeformat genutzt wurde, wäre ein solches Auswahlverfahren für unser Routenplanungssystem nur eine einfache Durchreiche-Instanz gewesen, auf die dann komplett verzichtet wurde.

4.8 Nutzung von C2C-Informationen

Parametrisierbares Routing bedeutet nicht nur intelligente Navigation und Optimierung des Individual- und Gesamtverkehrs, sondern beinhaltet auch Methoden, um autonome Fahrzeuge untereinander und mit anderen Fahrzeugen zu synchronisieren und nahtlos in den städtischen Verkehrsbetrieb zu integrieren. OECON hat dazu mit anderen Partnern an einer Integration der direkten Kommunikation über Infrastruktur-zu-Fahrzeug-Botschaften (C2C-Messages) gearbeitet.

Ein Anwendungsfall mit einer Roadside-Unit für C2C-Messages mit einem bidirektionalen Konverter für ODD wurde im Labor getestet. Dazu wurde ein Optimierungskonzept für Einzelfahrzeuge und die Gesamtheit der Fahrzeuge auf der Strecke unter Nutzung des bidirektionalen Converters zwischen C2C/C2I-Daten und ODD-Daten erstellt. Eine Roadside Unit wurde erweitert um einen bidirektionalen Konverter zwischen C2C/C2I-Daten und ODD-Daten.

Ein Betrieb und Kontrolle der RSU in Zusammenarbeit mit den anderen Parteien konnte im Rahmen des Projekts nur im Labor durchgeführt werden, da kein Zugriff auf reale RSUs erreicht werden konnte. OECON stellte hierzu für einen Anwendungsfall die benötigte Infrastruktur zur Verfügung.

5 Im Projekt entwickeltes Beispielsystem

5.1 Routing-Services

Um Fahrzeuge von A nach B zu navigieren, soll ein geeigneter Service die beste Verbindung finden und dabei gewünschte Randbedingungen (ODD) – Umfahrung von Baustellen, keine Maut, geringer Energiebedarf – berücksichtigen. Die Basis dieses Service lieferte ein Routings-Server, der im Rahmen eines früheren Projektes für die Routen- und Ladeplanung von E-Fahrzeugen entwickelt wurde und der eine sehr schnelle Routenberechnung ermöglicht.

Das umgesetzte Routing basiert auf dem Ansatz der „Contraction Hierarchies“ [Delling 2009]. Dies ist ein Routing -Algorithmus, der in der Graphentheorie und Informatik verwendet wird, um „Kürzeste-Pfad-Probleme“ in großen Graphen effizient zu lösen. Der Algorithmus wurde 2008 von Robert Geisberger, Peter Sanders und Dominik Schultes entwickelt. Die Grundidee der Contraction Hierarchies besteht darin, ein bestimmtes Diagramm vorzubereiten, indem Scheitelpunkte mit niedrigem Grad rekursiv zusammengefasst werden, bis eine hierarchische Struktur entsteht. Diese

hierarchische Struktur ermöglicht schnelle Abfragezeiten, wenn die kürzesten Pfade berechnet werden, da nur eine Untergruppe der Knoten untersucht werden muss.

Der Algorithmus funktioniert wie folgt: Zunächst wird das Diagramm durch iterativ verarbeitete Scheitelpunkte mit niedrigem Grad verarbeitet, um eine hierarchische Struktur zu erhalten. Anschließend wird ein Vorverarbeitungsschritt durchgeführt, in dem das Diagramm analysiert und Hierarchien konstruiert werden. Dieser Vorverarbeitungsschritt kann rechnerisch langwierig sein, muss jedoch nur einmal durchgeführt werden. Nach dem Vorverarbeitungsschritt können Abfragen nach kürzesten Pfaden effizient beantwortet werden, indem eine bidirektionale Suche im berechneten Diagramm verwendet wird. Der Algorithmus beginnt an den Quell- und Zielknoten und erweitert die Suche nach außen, wobei die Hierarchie die Suche leitet. Die Suche stoppt, wenn sich die beiden Suchvorgänge treffen oder wenn eine untere Grenze auf den Abstand zwischen den beiden Knoten festgelegt wird.

Contraction Hierarchies haben sich in der Praxis als sehr effizient erwiesen, insbesondere in großen Graphen.

5.2 Daten-Basis

Die Kartendaten stammen aus den Datenbeständen des OpenStreetMap-Projekts. Für das Projekt wurde ein vergleichsweise kleiner Ausschnitt im Norden Braunschweigs ausgewählt, um die vorbereitenden Berechnungen für den Routing-Algorithmus und Aktualisierungen der Datenbasis in kürzerer Zeit zu ermöglichen. Eine Erweiterung des Datenbestandes auf größere Areale ist trotzdem jederzeit möglich.

Die Darstellung eines Straßennetzes in digitaler Form erfolgt gewöhnlich in Form von einzelnen Segmenten, die über Knoten miteinander verbunden werden. Da Straßen sich gewöhnlich nicht wie Gerade verhalten, sondern Kurvenform besitzen, besitzen die meisten Segmente nur Knoten mit je einem weiteren Nachbarn. Die einzelnen Segmente bilden dann die Kurvenform nach. Lediglich jene Segmente, die direkt an Kreuzungen grenzen, haben an ihren Enden mehr als einen Nachbarn.

Straßennetze können gewöhnlich nicht komplett planar dargestellt werden, da einzelne Segmente (auch als Kanten bezeichnet) sich kreuzen können aufgrund von Brücken-Konstruktionen. Es entsteht daher ein 3D-Graph, dessen dritte Dimension allerdings nur schwach besetzt ist.

Jedes Segment besitzt mehrere Attribute. Die Klassifizierung beschreibt die visuelle Darstellung, damit bspw. alle Autobahnen in der Kartendarstellung gleich aussehen. Ein weiteres beschreibt die interne Typisierung, die festlegt, wie schnell man auf dem aktuellen Segment tatsächlich vorankommt. Hinzu kommt ein Fahrtrichtungsattribut – bspw. dürfen Einbahnstraßen und Autobahnen nur in einer Richtung durchfahren werden. Hinzu kommt noch die physische Ausdehnung, textuelle Beschreibungen für Segmente und Knoten und natürlich die geographischen Koordinaten.

Die Knoten besitzen keine Klassifizierung, sie enthalten aber dennoch wesentliche Informationen für das Routing. Beispielsweise enthalten sie Abbiegeverbote, also Informationen, ob bspw. an Kreuzungen ein Linksabbiegen wegen des Vorhandenseins eines entsprechenden Segments prinzipiell möglich wäre, aber aufgrund eines Verkehrszeichens untersagt ist.

Aus den Basisdaten werden gewöhnlich einerseits die visuellen Straßendaten generiert – diese Datenbank ist nur zur Darstellung der Karte auf einem Bildschirm geeignet und ist oftmals in Form von rechteckigen Arealen gespeichert, die wiederum in unterschiedlichen Auflösungen vorliegen. Damit kann eine Gesamtübersicht der Karte sehr schnell und ohne Details dargestellt werden, während Details nur in höheren Zoomstufen dargestellt werden.

Zum anderen wird ein aus Kanten und Knoten ein Graph gebildet, der für die Routenberechnung geeignet ist. Dieser enthält gewöhnlich keine visuellen Attribute, ist dafür aber als einheitliche spatiale Datenbank angelegt. Um die Routenberechnung zu beschleunigen, werden heutzutage dank



bezahlbarem Speicherplatz Routen vorberechnet und abgespeichert, meist nach dem Contraction-Hierarchies-Modell (siehe oben). Das erlaubt extrem schnellen Zugriff auf Routen. Diese Dateien sind meist sehr groß, enthalten aber keine dynamischen Informationen zu Verkehrszuständen, sofern sie nicht ständig aktualisiert werden.

5.3 Daten-Eingabe: Start- und Ziel-Koordinaten

Um eine Route zu berechnen, benötigt man mindestens zwei Koordinaten: den Startpunkt und den Zielpunkt. Bei Navigationssystem ist gewöhnlich nur die Eingabe des Ziels erforderlich; als Startpunkt wird der aktuelle Standort angenommen. Bei A-priori-Berechnungen liegt ein solcher Startort gewöhnlich nicht vor, so dass er ebenfalls eingegeben werden muss.

Üblicherweise kommt daher eine Georeferenzierung zum Einsatz, vereinfachend auch „Ortssuche“ oder „Geocoding“ genannt. Dabei werden beschreibende Werte eingegeben, zumeist Ort, Straße und Hausnummer. Eine Datenbanksuche liefert dann die Orte in der Datenbanktabelle, die zu den eingegebenen Werten passen. Hinter diesen Orten liegen dann die passenden Koordinaten, die für das Routing als Ausgangsbasis verwendet werden können.

Viele Navi-Systeme geben den aktuellen Standort in lesbarer Form aus, also als Beschreibung von Ort, Straße und Hausnummer. Dem zugrunde liegt die „Rückwärtssuche“, bei der den bekannten Koordinaten der jeweils nächste Ort aus der Ortsdatenbank zugeordnet wird. Diese Funktion wird auch als „Reverse Geocoding“ genannt. Sie ist für ein Routing nicht zwingend erforderlich.

5.4 Daten-Ausgabe: Die Fahrtroute

Nachdem aus Start-, Ziel- und ggf. Zwischenstationen eine Route berechnet wurde, steht diese sowohl in visueller als auch in textueller Form zur Verfügung. Die Routingalgorithmen liefern dabei zunächst eine komplette Routenbeschreibung, die alle Segmente und alle Knoten enthält, die die Route umfasst. Für eine grafische Darstellung der Route sind diese Angaben erforderlich. Eine textuelle Ausgabe oder eine Routenführung per Navigation erfordert allerdings eine Priorisierung, die nach der eigentlichen Routenberechnung durchgeführt wird. Dabei werden die erforderlichen Fahrmanöver nach ihrer Relevanz gewichtet. Wichtige Manöver sind dabei:

- Abzweigungen, die zum Verlassen des aktuellen Weges führen. Dies sind die typischen Abbiegemanöver. Auch Kreisel und Autobahnzufahrten fallen in diese Kategorie.
- Geradeausfahrten, wenn der Name der Route sich ändert, beispielsweise bei Autobahnen, deren Änderung dann mit einem „Weiter auf A7“ angekündigt werden. Auch im innerstädtischen Bereich kann sich der Name längerer Straßen zwischendurch ändern.
- Geradeausfahrten, wenn es mehrere Möglichkeiten gibt, beispielsweise an Kreuzungen, bei denen zusätzlich zum schnelleren Befahren auch eine Untertunnelung vorhanden ist.

Alle Knoten mit nur zwei Segmenten ohne weitere Änderung – also reine Verbindungsknoten – sollen nicht zu Ansagen bzw. Routenempfehlungen führen.

Was in der Theorie sehr einfach klingt, ist in der Praxis allerdings nicht immer eindeutig. Jeder, der schon einmal ein Navigationssystem benutzt hat, kennt Situationen, an denen eine Ansage fehlt, nicht eindeutig oder sogar falsch war. Die Ursachen hierfür können vielfältig sein; zumeist ist allerdings eine falsche Priorisierung der Grund für ein solches Verhalten, die allerdings wiederum zumeist in Fehlern des zugrundeliegenden Kartenmaterials begründet ist.

Textuelle Angaben des Routings enthalten heute auch zumeist zusätzliche Richtungsinformationen, die inzwischen auf den Daten beruhen, die sich auch auf Schildern wiederfinden, während sie in der Anfangszeit der Routenplanung noch automatisiert aus den Routingdaten selbst erzeugt wurden.



5.5 Einbindung in das GAIA-X-Umfeld

Um den Routing-Service in das GAIA-X 4 AMS-Projekt einzubinden, wurde eine Anwendung entwickelt, die den Routing-Service im Hintergrund nutzt und zunächst per REST-Schnittstelle die Nutzung der Routenplanung über das Internet ermöglicht. Die Kommunikation startet dabei im Modul "Reaktionsplaner", das — bildlich gesprochen — die Bedienoberfläche des Fahrzeugs darstellt. Der Nutzer gibt dort Start und Ziel sowie Routenwünsche ein und gibt das Start-Kommando. Das Modul sendet daraufhin die Koordinaten an den Routing-Service und hängt die Randbedingungen in Form einer ODD an.

Der Routing-Service empfängt die Anfrage und plant eine Route vom Start zum Ziel, ohne dabei die ODD anzuschauen. Das ist kein Fehler, sondern durchaus so gewollt, denn in GAIA-X sind die einzelnen Komponenten also Micro-Services ausgelegt. Jeder Service kümmert sich nur um eine klar und eng abgegrenzte Aufgabe, in diesem Fall das Planen der kürzesten Strecke. Diese Strecke wird an das Modul "ODD-Matching" gesendet. Dort erst geschieht der Abgleich der Route mit den Randbedingungen. Alle Randbedingungen, die durch die Route verletzt werden, listet das ODD-Matching-Modul auf und sendet sie an den Routing-Service zurück. Sollten keine Fehler auftreten, enthält die Antwort eine leere Liste.

Der Routing-Service schaut sich die Antworten des Matching-Moduls an und platziert für jeden Fehler einen Malus an der entsprechenden Kante im Straßennetz. Anschließend startet der Routing-Durchgang erneut. Unerwünschte Abschnitte sollten durch den zuvor eingetragenen Malus nicht mehr in der Route berücksichtigt werden, es sei denn, die Ausweichmöglichkeiten sind noch schlechter oder nicht vorhanden. Die neue, korrigierte Fassung geht dann erneut an den Matcher. Dieser Kreis wiederholt sich so lange, bis entweder kein Fehler mehr vorhanden ist oder 5 Nachbesserungen durchgeführt wurden. Die dann letzte Route geht an den Reaktionsplaner als Ergebnis zurück.

5.6 Demonstratoren

Im Rahmen des Projektes wurden insgesamt zwei Demonstratoren entwickelt, die die Arbeit des Routing-Services veranschaulichen sollten. Beide wurden mit Spring Boot in Kotlin als Web-Anwendung entwickelt, die auf den Routing-Service über eine eigens dafür vorgesehene Schnittstelle zugreifen. Um Anwender-Interaktionen und Animationen zu ermöglichen, kam auf der Webseite auch Javascript und CSS zum Einsatz.

Der erste Demonstrator zeigt einen Kartenausschnitt sowie zwei Marker für Start und Ziel an. Die ermittelte Route wird als blaue Linie entlang der Straßen dargestellt. Der Anwender hat die Möglichkeit, die Marker für Start und Ziel auf der Karte zu verschieben. Um anschließend eine Route zu berechnen, muss der Anwender den Submit-Button klicken. Dann plant der Routing-Service eine neue Route zwischen beiden Punkten und schickt sie zur Prüfung, wie oben beschrieben, an den ODD-Matcher. Das Ergebnis aus diesem Prozess wird dann auf der Karte angezeigt.

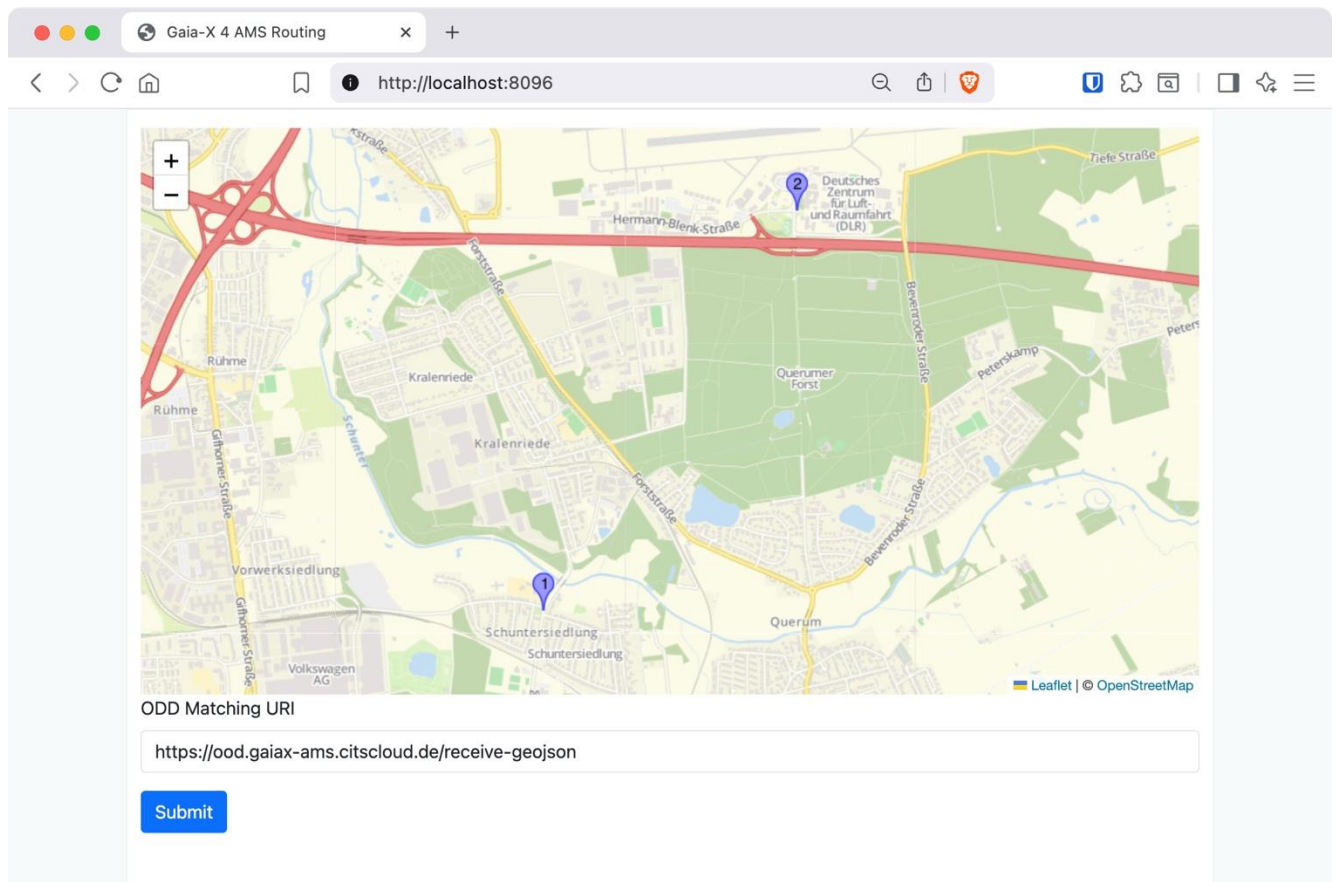


Abbildung 5: Der erste Demonstrator diente hauptsächlich der Validierung der Planungsergebnisse. Er zeigt die geplante Route nach Abstimmung mit dem ODD-Matcher auf der Karte an.

Der zweite Demonstrator ist sehr viel umfangreicher. Er zeigt neben dem Kartenausschnitt und den Markern für Start und Ziel noch ein Prozess-Diagramm, das während der Berechnung animiert die Kommunikation der einzelnen Dienste veranschaulicht. Die Animationen wurden mittels Javascript, CSS und SVG-Grafiken umgesetzt. Da die Antwortzeiten der Dienste untereinander im Millisekunden-Bereich liegen, wäre eine Animation allerdings sehr schnell vorbei und für Außenstehende gar nicht wahrnehmbar. Aus diesem Grunde wurden die Animationen der Kommunikationsschritte künstlich in die Länge gezogen. Ein Moderator hat so während der Ausführung genug Zeit, ein paar begleitende Sätze zu sprechen und die Arbeit der Software zu erklären.

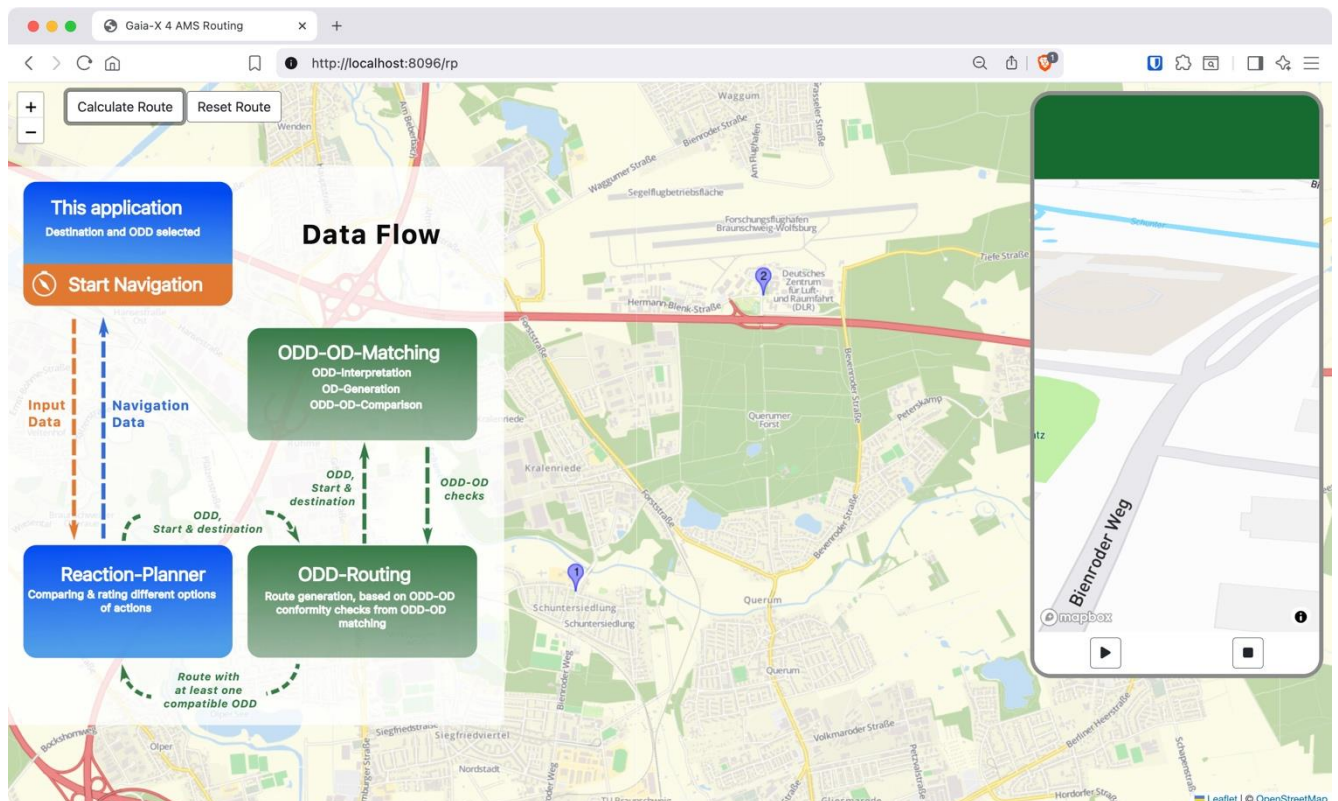


Abbildung 6: Während die Planung einer Route läuft, zeigt ein animiertes Prozess-Diagramm (links) die Kommunikation der Module und Service untereinander an.

Anders als der erste Demonstrator spricht der zweite nicht direkt mit dem Routing-Service sondern schickt seine Anfrage an den Reaktionsplaner. Er simuliert also die Eingaben eines Fahrers im Fahrzeug und schickt nur die ausgewählten Start- und Ziel-Koordinaten. Der Reaktionsplaner fügt dieser Anfrage anschließend einen Standardsatz von Randbedingungen hinzu und schickt die Anfrage weiter an den Routing-Service. Von dort läuft die weitere Planung dann wieder wie oben beschrieben ab.

Zusätzlich zum Prozess-Diagramm zeigt der zweite Demonstrator ein simuliertes Handy mit Navigations-Software an. Der Anwender hat nach dem Planen der Route hier die Möglichkeit, die Fahrt simuliert zu starten. So wird gezeigt, dass die erzeugte Route auch Navigationsanweisungen enthält, die das simulierte Handy auf dem Bildschirm anzeigt.

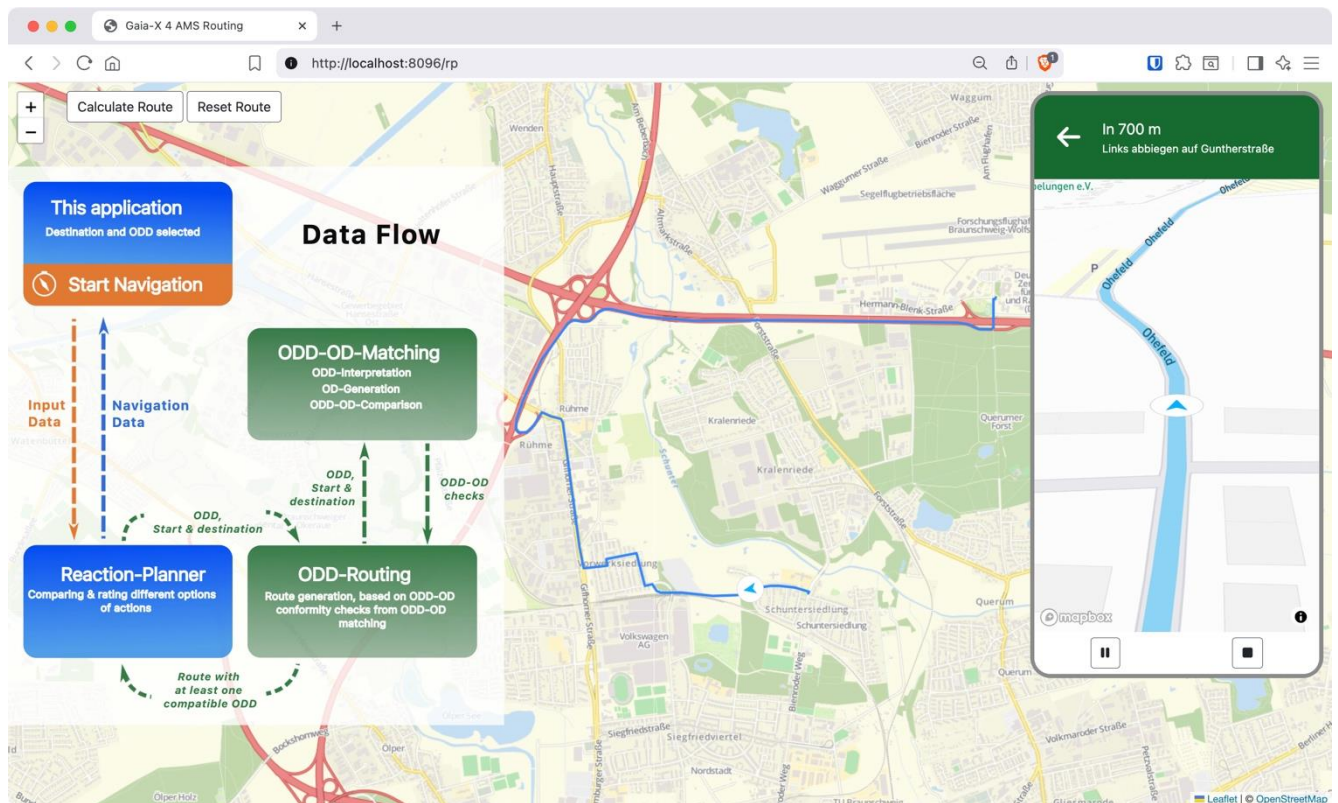


Abbildung 7: Im Anschluss an die Planung kann der Anwender die geplant Route auf einem simulierten Handy-Bildschirm (rechts) starten und erhält dort Navigationshinweise.

Das System ist nach wie vor verfü- und nutzbar unter <https://gaiax4ams01.comloc.net/rp> (User „gaiax“, Passwort „ntarian“).

6 Mitarbeit bei anderen Projektteilen

6.1 TP1 und TP2 GAIA-X Framework und Infrastruktur-Ökosystem

OECON hat an Terminen zur Koordination von TP2 teilgenommen. Unser Schwerpunkt war hier der des Lernenden; wir haben die Informationen und das Wissen, welches hier vermittelt wurde, eingesetzt, um unsere Dienste GAIA-X-konform umzugestalten.

6.2 TP3 Daten- und Dienste-Ökosystem

OECON hat seit Juni 2022 (nachträgliche Aufnahme als Partner) teilgenommen an den TP3-Meetings zur Datenkonsolidierung. Es stellte sich im Laufe der Zeit heraus, dass einige Teile der ursprünglich in TP3 geplanten Aufgaben in TP4 abgehandelt werden, da ein direkter Bezug zu der dortigen Aufgabe besteht und sich die Aufwände daher nicht immer sauber trennen lassen. Insofern finden sich einige der ursprünglich für TP3 vorgesehenen Ergebnisse nun im Kapitel 4 wieder.

OECON hat in TP3 eine Ausarbeitung von Datenaufbereitungs- und Datenvalidierungsmechanismen durch Datenanalyse- und Datenspeicherungstools vorgenommen, um das in TP4 bereitzustellende datenzentrierte SaaS darstellen zu können. Dazu gehörten ein Parametrisierbares Routing, Individualisierte Routing-Optimierung, Verkehrsführung (Emergency Lanes) und Strategieeinsatz



Monitoring. Im Rahmen von TP3 hat OECON auch eine GAIA-X-konforme Parametrisierung der Datennutzung der OECON-Dienste durchgeführt sowie Maßnahmen zur Datensicherheit bei Mobilitätsdiensten aus technischer Sicht (Plattform und IT-Architektur-Anpassungen) und aus inhaltlicher Sicht (Datenspeicherung, Datennutzungsbeschränkungen) durchgeführt

Es wurden in mehreren Terminen die Zuteilung der Verantwortlichkeiten im Daten- und Dienst-Ökosystem diskutiert, OECON hat Leitung in einem Bereich übernommen. Dabei haben u.a. wir im Rahmen des Gebiets „ODD-Routing“ Anforderungen an die User-Stories definiert und unsere Datenbestände katalogisiert und detailliert beschrieben. Wir haben auch Arbeiten rund um die Dateneinordnung übernommen; hierzu haben drei Workshops stattgefunden.

6.3 TP6 Projektmanagement

OECON hat an folgenden Projekttreffen teilgenommen und mitgewirkt:

- zweitägiges Konsortialtreffen der Partner am 21. und 22. Juni 2022
- Eintägiger Workshop der TP4-Gruppe am 1. Dezember 2022 – das erste Vor-Ort-Treffen nach dem Ende der Corona-Pandemie. Hier wurden erste Zusammenhänge zwischen den einzelnen Diensten erarbeitet und die Kommunikationsstruktur der Dienste untereinander festgelegt.
- zweitägigen Konsortialtreffen der Partner am 10. und 11. November 2022. OECON hat dort in Vertretung des Partners Peregrine die bisherigen Ergebnisse präsentiert. Außerdem wurden dort gemeinsame Anforderungen von TP4 und TP5 an Dienste definiert. Das von OECON zur Verfügung gestellte Routing ist ein solcher Dienst, daher wurden unsere Anforderungen an Schnittstellendaten bei dieser Gelegenheit eingebracht.
- zweitägiges Konsortialtreffen der Partner am 14. und 15. Juni 2023. OECON hat dort in Vertretung des Partners Peregrine die bisherigen Ergebnisse präsentiert – zu diesem Zeitpunkt nur in Papierform. In einem Workshop wurde dort besprochen, wie die Umsetzung eines Minimum Viable Demonstrators (MVD) aussehen kann – siehe hierzu Kapitel 5.6.
- zweitägiges Konsortialtreffen der Partner vom 26. bis 28. September 2023. OECON hat dort die bisherigen Ergebnisse präsentiert – erstmals auch als Live-System. In mehreren Workshops wurden dort Themen besprochen zur Datenqualität, Datenorganisation und zur weiteren Entwicklung des MVD.

OECON hat des Weiteren teilgenommen den Diskussionen zur Veranstaltung der Abschlussdemonstration und hat dort auch Aufgaben übernommen. Des Weiteren wurden im Betrachtungs-zeitraum mit allen Partnern Diskussionen über eine kostenneutrale Projektverlängerung geführt.

Durch die Übernahme des Projektmanagements durch einen externen Dienstleister wurden auch neue Konzepte zur Verwaltung der Dokumente eingeführt. Das zentrale Instrument war seitdem OpenProject, für die Spezifikationen zu den Diensten kam gitlab zum Einsatz.