

Schlussbericht

zu IGF-Vorhaben Nr. 20642 BG / 2

Thema

Entwicklung eines Diagnosesystems für kooperative, intelligente Verkehrssysteme zur Inbetriebnahme und Wartung am Beispiel von Car2X

Berichtszeitraum

01.04.2019 - 31.03.2021

Forschungsvereinigung

Gesellschaft zur Förderung angewandter Informatik e.V. - GFal

Forschungseinrichtung(en)

Gesellschaft zur Förderung angewandter Informatik e.V. - GFal

Institut für Automation und Kommunikation e.V. - ifak

Gefördert durch:

Inhaltsverzeichnis

1. Zusammenfassung.....	3
2. Gegenüberstellung der durchgeführten Arbeiten und der Ergebnisse mit den Zielen	4
AP 1: Spezifizierung von Anwendungsfällen und Nutzergruppen.....	4
AP 2: Spezifizierung der Gesamtsystemarchitektur und Festlegung des Testszenarios für den Demonstrator.....	9
AP 3: Konzeption und Entwicklung von domänenspezifischen grafischen Sprachen für die Modellierungssysteme	11
AP 4: Konzipierung des Verfahrens zum automatischen Auswerten von Diagnosedaten.....	12
AP 5: Darstellungsformen der Diagnoseergebnisse.....	14
AP 6: Entwicklung des Gesamtsystems bezüglich Modulschnittstellen und der Teilkomponente „Diagnosesystem“	16
AP 7: Entwicklung und Implementierung der Schnittstellen zu technischen Komponenten von intelligenten Verkehrssystemen	19
AP 8: Entwicklung und Implementierung des Teilsystems zur Modellierung von Diagnoseprozessen	22
AP 9: Entwicklung und Implementierung des Teilsystems zur Modellierung sowie automatisierten und hoch performanten Auswertung von Diagnoseregeln.....	24
AP 10: Erzeugen der Testanwendung mit dem Software-Demonstrator.....	24
AP 11: Evaluierung des Gesamtsystems anhand der Demonstrator-Anwendung	25
AP 12: Projektmanagement, Berichte und Publikationen	26
3. Angaben zu den aus der Zuwendung finanzierten Ausgaben.....	27
4. Erläuterung der Notwendigkeit und Angemessenheit der geleisteten Arbeit.....	27
5. Darstellung des wissenschaftlich-technischen und wirtschaftlichen Nutzens.....	27
6. Fortschreibung des Plans zum Ergebnistransfer in die Wirtschaft	29
Durchgeführte Transfermaßnahmen	29
Geplante Transfermaßnahmen nach Abschluss des Vorhabens.....	30
Einschätzung zur Realisierbarkeit des Transferkonzepts	32

1. Zusammenfassung

Innerhalb der geplanten Projektlaufzeit konnten alle Arbeitspakete erfolgreich bearbeitet werden. Unterstützt wurde dies durch die sehr gute Zusammenarbeit mit Mitgliedern des projektbegleitenden Ausschusses (PA). Im AP 12 (Projektmanagement) wurden PA-Treffen durchgeführt, die wirtschaftlichen Interessen des PA im Themenfeld V2X abgefragt und Feedback zu den Projektzielen und -fortschritten eingeholt.

Durch dieses Feedback konnten in AP 1 die potenziellen Nutzergruppen definiert und über eine User Story Anwendungsfälle für V2X-Diagnosen aufgestellt werden. Aus Gesprächen mit dem PA ergab sich der Wunsch, den Status von V2X-Sendern in einem Dashboard darzustellen, z. B. auf einer Karte.

In AP 2 wurde eine allgemeine Systemarchitektur für die einzelnen Teile des Diagnose-Systems definiert. Dazu zählen die Definition der Kommunikationspartner (OBUs, RSUs), das passive Mitlesen der unverschlüsselten Kommunikation über ETSI-G5-Broadcasts, Schnittstellen zwischen den geplanten Softwareteilen von GFal und ifak sowie der generelle Ablauf beim Planen und Durchführen einer Diagnose.

Zu den wesentlichen Ergebnissen von AP 3 zählen die Beschreibung der Diagnoseprozesse über Ablaufgraphen sowie ein Konzept für eine grafische domänenspezifische Sprache zur Modellierung von Diagnoseprozessen.

Parallel dazu wurden in AP 4 mögliche Verfahren zum effizienten Auswerten von Diagnosedaten zur Laufzeit zusammengestellt. Zu diesem Zeitpunkt waren Petrinetze die präferierte Methode, verteilte Systemzustände über Software zu modellieren, um einen fortschreitenden Diagnoseprozess nachverfolgen zu können.

AP 5 befasste sich mit geeigneten Darstellungsformen von Diagnoseergebnissen, z. B. Statusdaten von V2X-Sendern oder die Einhaltung von durch den Nutzer aktivierten Diagnoseregeln. Als nützliche Bestandteile des Laufzeit-Diagnosesystems stellten sich eine Karte sowie eine tabellarische Protokollansicht heraus, für die Papierprototypen erstellt wurden.

Nachdem in den vorhergehenden APs Konzepte erstellt wurden, konnte ab AP 6 mit den Modulschnittstellen die Implementierung beginnen. Es wurde ein MQTT Message Broker eingesetzt, an dem sich die Teilkomponenten des Diagnosesystems als Clients anmelden und Daten in Form von JSON-Strings austauschen. Diese Form der Interprozesskommunikation ermöglichte den Betrieb von Datenerfassung, Vorverarbeitung, Auswertung und Darstellung auf unterschiedlichen Geräten an unterschiedlichen Standorten. Ebenfalls wurde in diesem AP eine Mensch-Maschine-Schnittstelle mit Dialogen in der GUI umgesetzt.

Die Implementierung der Datenerfassung aus den Verkehrssystemen bzw. von potenziellen V2X-Sendern erfolgte in AP 7. Sie basiert nun auf dem Erfassen von Bytestreams in ASN.1-UPER-Kodierung, der Dekodierung nach aktuellen ETSI-Standards und der Überführung in JSON-Strings für die weitere Verteilung im Diagnosesystem.

In AP 8 geschah die Implementierung des Teilsystems zur Modellierung von übergeordneten Diagnoseprozessen. Dabei wurde die Software QuickSteps der GFal für die entworfene grafische domänenspezifische Sprache erweitert.

Daraufhin konnte in AP 9 die Modellierung von detaillierten Diagnoseregeln vorgenommen werden. Die Regeln werden in einer Datenbank abgelegt. Durch Befehle an die Diagnose-Engine

(wie dem Start einer Regel über eine ID) kann die Diagnose-Engine auf die Datenbank zugreifen. Zur Diagnose-Laufzeit geschieht die Auswertung über ein internes Zustandsmodell, das alle erfassten V2X-Rohdaten gegen alle derzeit aktivierten Regeln abgleichen kann.

Nachdem eine erste Version des prototypischen Diagnosesystems fertiggestellt war, konnte die zu Beginn konzipierte Testanwendung in AP 10 erzeugt werden. Ebenfalls wurde am ifak eine Roadside Unit (RSU) aufgebaut und derart parametrierung, dass sie für das Testszenario passende V2X-Nachrichten verschickte. Sie simulierte über SPATEM und MAPEM die Daten einer Lichtsignalanlage.

Zuletzt konnte in AP 11 anhand dieser Demo-Anwendung das Gesamtsystem evaluiert werden. Dabei unterstützten einige Mitglieder des PA, denen die Software zur Verfügung gestellt wurde. Über Aufzeichnungen der ifak-RSU im JSON-Format konnten die Partner auch ohne entsprechende Hardware das Szenario bei sich abspielen und die Ergebnisse in der GUI beobachten.

Letztendlich wurden die Projektziele mit der Fertigstellung aller wesentlichen Softwareteile erreicht. ifak und GFal bleiben mit den PA-Mitgliedern in Kontakt, um eine zukünftige Weiterentwicklung und Verwertung zu planen.

2. Gegenüberstellung der durchgeführten Arbeiten und der Ergebnisse mit den Zielen

Für die detaillierte Darstellung der Projektergebnisse werden in diesem Bericht die Arbeitspakete herangezogen.

AP 1: Spezifizierung von Anwendungsfällen und Nutzergruppen

Dieses Arbeitspaket diente der Vorbereitung aller weiteren Arbeiten. AP 1 hatte zum Ziel, in einem Dokument die wichtigsten technischen Begriffe, bekannte V2X-Tesfelder, eine User Story, Nutzerrollen und schließlich konkrete Anwendungsfälle festzuhalten. An diesen Anwendungsfällen konnten sich die folgenden Konzepte und Implementierungen orientieren. Die Recherche und Dokumentation geschah bereits in Zusammenarbeit mit Mitgliedern des Projektbegleitenden Ausschusses.

Zu Beginn wurden die wichtigsten technischen Begriffe aus der Domäne der Intelligenten Verkehrssysteme (ITS) mit Fokus auf V2X-Kommunikation definiert. Dadurch konnte Fachwissen des ifak mit den Projektpartnern bei der GFal geteilt werden, um eine gemeinsame Basis für die Projektbearbeitung zu schaffen. Unter den wichtigsten Abkürzungen, auch zum besseren Verständnis dieses Berichtes, sind:

- **V2X** (Vehicle-to-anything-Kommunikation). Synonym: Car2X. Meint für den Verkehr standardisierte Nachrichten und Dienste, die entweder über WLANp (derzeit Standard-Technik) oder zukünftig und alternativ über Mobilfunk (5G, Ad-hoc) übertragen werden.
- **WLANp** (WLAN nach dem Standard IEEE 802.11p für den Verkehr). Dies ist eine Möglichkeit zur drahtlosen Nachrichtenübertragung. Eine Alternative wäre Mobilfunk (LTE-V2X oder 5G). Die standardisierten Nachrichten wie CAMs und DENMs sind jedoch von diesen Funkstandards unabhängig und können prinzipiell über alle übertragen werden.
- **LSA** (Lichtsignalanlage). Fachbegriff für Ampel.

- **RSU** (Roadside Unit). Stationäre V2X-Basisstation. Z. B. an Kreuzungen für LSA-Informationen installiert.
- **OBU** (On-board Unit). Mobile V2X-Basisstation für Fahrzeuge.
- **GLOSA** (Green Light Optimal Speed Advise). Vorschlag einer optimalen Geschwindigkeit für Fahrzeuge, damit sie bei Grün die nächste LSA erreichen können.
- **PKI** (Public Key Infrastructure). System aus Servern und Diensten, um die Signierung von Nachrichten zu ermöglichen. Es werden geheime private Schlüssel verwendet und öffentliche Schlüssel durch die PKI bereitgestellt, um Sender als authentisch zu kennzeichnen.

Als Nächstes wurden im AP 1 die Domänen für das geplante DIAKO-System abgesteckt. Intelligente Verkehrssysteme mit drahtloser Kommunikation lagen von Anfang an im Fokus und wurden hier detailliert im entsprechenden Dokument beschrieben.

Beispiele für mobile Nutzer sind private PKW- und Motorradfahrer (Überbegriff *Motorisierter Individualverkehr*, MIV), Busse und Straßenbahnen im Öffentlichen Personennahverkehr (ÖPNV) oder, zusammen mit Eisenbahnen und weiteren Verkehrsträgern, Öffentlicher Verkehr (ÖV). Entsprechende Fahrzeuge eignen sich potenziell für die Ausstattung mit Kommunikationsmodulen und sind daher für Diagnosen zu berücksichtigen.

Beispiele für juristische Personen sind Verkehrsbehörden und Kommunen (als Oberbegriff für Städte und Gemeinden), die Fahrzeuge (wie Busse), aber auch Infrastrukturen (wie Lichtsignalanlagen) betreiben können. Ebenfalls sind Kommunen potenzielle Betreiber von V2X-Roadside-Units und somit für die Gruppe der Anwender des DIAKO-Systems zu berücksichtigen.

Vor allem für mobile Nutzer sind drahtlose Kommunikationstechnologien nötig. Beispielsweise wird für Notrufe über den jetzt gesetzlich vorgeschriebenen *eCall* Mobilfunk genutzt. Mobilfunk hatte aber in den bisherigen Generationen bis 3G immer das Problem, Funkzellen zu erfordern. Da aber der Bedarf bestand, auch auf Landstraßen und Autobahnen, wo ggf. keine Funkzellenabdeckung vorhanden ist, Dienste bereitzustellen, wurde der etablierte WLAN-Standard *IEEE 802.11* herangezogen. Für Endnutzer in Privat- oder Bürogebäuden existieren hier Erweiterungen des Standards wie a, b, g oder n. Für den Verkehrssektor wurde der Standard *IEEE 802.11p* geschaffen, der im Folgenden zur Vereinfachung *WLANp* genannt wird.

Ein entscheidender Vorteil von *WLANp* ist die *Broadcast-Fähigkeit*: Alle *WLANp*-Empfänger in der Nähe können eine versandte Nachricht erhalten, ohne zuvor mit dem Sender ein Netzwerk aufbauen zu müssen. Mit der Mobilfunkerweiterung LTE-V2X wurden diese Broadcasts bereits beim 4G-Mobilfunk erprobt. Und in einer späteren Version von 5G ist dies ebenfalls vorgesehen. Fakt ist aber, dass bisher nur *WLANp* marktreif ist und in ersten Fahrzeugen und Kommunen genutzt werden kann.

Auf diese Kommunikationsstandards der unteren OSI-Layer setzen Nachrichten-Spezifikationen auf. In Europa sind die V2X-Nachrichten der ETSI (Europäisches Institut für Telekommunikationsnormen) relevant. Folgende standardisierte Nachrichten wurden für die Anwendungsfälle in DIAKO als besonders relevant identifiziert:

- **CAM**: Cooperative Awareness Message. Zyklisch versendete Statusdaten, z. B. Fahrzeugposition und -geschwindigkeit.
- **DENM**: Decentralized Environmental Notification Message. Zeitlich begrenzte Ereignismeldungen, z. B. Gefahrenwarnungen wie „Liegen gebliebenes Fahrzeug“.

- **MAP bzw. MAPEM:** Map (Karte) bzw. Map European Message. Einfache digitale Karte eines Verkehrsknotens mit Fahrspuren, die über eine Punktkette definiert werden.
- **SPAT bzw. SPATEM:** Signal Phasing and Time bzw. SPAT European Message. Informationen über Lichtsignalanlagen (bzw. den Signalgruppen) wie aktuelle Farbe, prognostizierter Umschaltzeitpunkt, Geschwindigkeitsempfehlungen und die Zuordnung zu Fahrspuren aus einer zugehörigen MAP.
- **SRM bzw. SREM:** Signal Request (European) Message. Anfrage an LSAs für eine Priorisierung (Grün für eine Signalgruppe bzw. Fahrspurbeziehung erhalten).
- **SSM bzw. SSEM:** Signal Status (European) Message. Antwort auf eine SREM.

Wichtig ist, zu verstehen, dass diese eigentlichen V2X-Nachrichten mit ihren Nutzdaten (Payload) prinzipiell jeden Übertragungsweg nehmen können, sei es WLANp, Mobilfunk oder drahtgebundenes Ethernet (LAN). Somit untersuchte DIAKO Anwendungsfälle und Nutzer, die derzeit aufgrund der Marktreife WLANp und ETSI-Nachrichten nutzen, und richtete darauf den Demonstrator aus. In Zukunft sollte aber die Übertragbarkeit auf 5G-Mobilfunk und darüber versendete (ETSI-) Nachrichten prinzipiell möglich sein.

Eine weitere Domäne, die eine enorme wirtschaftliche Bedeutung in Deutschland hat und in der das ifak seit seiner Gründung Expertenwissen aufgebaut hat, ist die Automatisierungstechnik. Hier kommen vor allem Feldbussysteme zum Einsatz, von denen PROFIBUS, Echtzeit-Ethernet und PROFINET einige Beispiele sind. Hier spielen Speicherprogrammierbare Steuerungen (SPS) und Feldgeräte mit Schnittstellen zu Sensoren und Aktoren eine wichtige Rolle.

Im Verlauf von AP 1 wurde vorrangig auf die Kommunikation und Daten von V2X in Intelligenten Verkehrssystemen eingegangen und es wurden vorherrschende Abläufe untersucht, um daraus Anforderungen an die Diagnose abzuleiten. Für eine Übertragbarkeit der Ergebnisse wurden zusätzlich Abläufe in Feldbussystemen am Beispiel PROFINET untersucht, da erwartet wurde, dass sich ähnliche Anforderungen ergeben. Die Gemeinsamkeit ist eine paketbasierte Kommunikation über ein Übertragungsmedium, sei es drahtlos oder drahtgebunden.

Die umgesetzte Diagnosemethode zielte nicht auf Merkmale der unteren Protokollschichten des OSI-Schichtenmodells ab. Stattdessen wurden Anforderungen betrachtet, die ab Layer 3 (Vermittlungsschicht mit Paketen) bis hin zur Anwendung (über Layer 7) bestehen.

In einem weiteren Schritt wurden durch die Forschungsstellen bekannte V2X-Testfelder und V2X-Projekte in Deutschland recherchiert und dieses durch Gespräche mit dem PA ergänzt. So war Green Way Systems beispielsweise in den Projekten „fast traffic“ und „fast sign“ involviert, bei denen vor allem die schnelle Übermittlung von Informationen und Authentifizierung von Sendern im Vordergrund standen. Weitere wichtige Projekte mit entsprechenden Testfeldern waren SIRENE (Braunschweig), 5GLabBraWo (Braunschweig und Wolfsburg), VERONIKA (Kassel), BiDiMoVe (Hamburg), HarmonizeDD (Dresden), KoMo:D (Düsseldorf) und Safe now (Ingolstadt).

Durch diese Zusammenstellung wurde zum einen festgestellt, dass viele V2X-Aktivitäten in Deutschland durch öffentliche Gelder gefördert wurden und werden. Sowohl die Privatwirtschaft als auch Kommunen bauen dabei vor allem Know-how mit neuen Technologien, Abläufen und Anwendungsfällen auf. Diagnoseprozesse sind noch nicht etabliert. Daher sehen die Forschungsstellen die Ergebnisse von DIAKO als sehr gute Gelegenheit, in diesem frühen Stadium der Qualitätssicherung für V2X unterstützen zu können, um in Zukunft leistungsfähige Werkzeuge auf den Markt zu bringen.

Zum anderen wurden die Anwendungsfälle der Priorisierung an Lichtsignalanlagen sowie die Verbreitung von LSA-Informationen (GLOSA) über V2X in vielen Projekten angetroffen (z. B. SIRENE und BiDiMoVe). Die Beispiele wurden daher als sehr gut geeignet angesehen, um damit das Diagnosekonzept in den Anwendungsfällen zu beschreiben und darauf die spätere Testanwendung für Demonstration und Evaluation aufzubauen.

Bereits im Projektantrag wurde eine sogenannte User Story skizziert und nun im AP 1 ausgebaut. Sie beschreibt, wie ein imaginärer Anwender das geplante DIAKO-System nutzen würde. Dabei wurde beschrieben, wie ein LSA-Hersteller eine V2X-fähige LSA als neues Produkt auf den Markt bringen möchte. Während der Entwicklung als auch nach der Installation in einer Kommune werden die Diagnoseprozesse und Werkzeuge aus DIAKO benutzt, um die Kommunikation der V2X-LSA zu untersuchen und potenzielle Fehler zu finden. Als Vorteil wurde dabei herausgestellt, wie einmal aufgestellte Diagnoseregeln an Kollegen oder Auftraggeber weitergereicht werden können, um Expertenwissen weitergeben zu können.

Durch diese User Story konnten zum einen potenzielle Anwender in Gruppen eingeteilt werden. Es wurden Hersteller, Kommunen, Dienstleister und Entwickler von Test-Lösungen identifiziert. Bei der Anwendung von DIAKO-Lösungen können diese Anwender prinzipiell zwei verschiedene Nutzerrollen einnehmen: Einrichter und Anwender. Diese wurden beim späteren Design vom Modellierungs- und dem Laufzeitdiagnosesystem bedacht.

Zuletzt wurden im AP 1 aus der User Story konkrete Anwendungsfälle und resultierende Anforderungen an die DIAKO-Software abgeleitet. Für jeden Anwendungsfall wurden involvierte Akteure, Anfangsbedingungen, Ablauf und Endbedingungen jeweils ohne und dann mit der geplanten Diagnose beschrieben. Folgende Fälle wurden definiert:

- Kontinuierliches Versenden von Ampelphasennachrichten (SPATEM) nachweisen
- Nachweisen, dass eine LSA-Priorisierungsanfrage (über SREM) beantwortet wird (über SSEM)
- Nachweisen, dass eine nicht signierte LSA-Priorisierungsanfrage (über SREM) abgewiesen wird

Aus diesen Fällen wurden Anforderungen abgeleitet. Folgende wurden als absolut erforderliche Features festgelegt:

- Eine Auswahl von standardisierten ETSI-Nachrichten wie SPATEM etc. verarbeiten können
- Empfangszeitpunkte verarbeiten und mit bisherigen empfangenen Nachrichten vergleichen können, um z. B. Zykluszeiten ermitteln zu können
- Zustände für „kein Fehler“, „Fehler“ und ggf. mit Codes festlegen, um sie in der GUI präsentieren zu können
- Im Fehlerfall geeignete Lösungsschritte präsentieren können
- Die Lösungsschritte manuell festlegen und erweitern können (z. B. im Modellierungssystem)
- Timer implementieren, der das Nicht-Auftreten von V2X-Nachrichten überwachen kann

Parallel zu diesen Arbeiten wurde das erste PA-Treffen geplant und am 18.09.2019 am ifak in Magdeburg durchgeführt. Es wurden alle Vertreter des Projektbegleitenden Ausschusses eingeladen. Teilgenommen haben vier Vertreter der Forschungseinrichtungen sowie sieben

Personen des PA von insgesamt fünf Organisationen. Dadurch waren auch Branchenzugehörigkeit und Projektinteressen vielfältig: LSA-Hersteller, Softwareentwickler, RSU-Hersteller, Dienstleister im ÖPNV-Bereich, Anbieter von Telekommunikationslösungen sowie eine Hochschule. Den Anwesenden wurden das Projekt vorgestellt. Anschließend wurden aktuelle und zukünftige Projekte sowie Anwendungsfälle im Bereich Kommunikation im Verkehr mit Schwerpunkt V2X diskutiert. Auch wurden existierende und geplante Testfelder in Deutschland gesammelt und Anforderungen mit dem PA diskutiert.

Die weiteren PA-Treffen wurden aufgrund von Reisebeschränkungen in der Corona-Pandemie über Web-Konferenzen abgehalten. Dabei konnten jedes Mal weitere Texte im AP 1-Dokument ergänzt werden. So wurde zum Beispiel erst später im Projekt durch ein PA-Mitglied der Wunsch geäußert, eine Art Dashboard zur Verfügung zu haben: Hierüber könnte ein Anwender live die erfassten Daten von beobachteten V2X-Sendern betrachten. Dazu würden aktuell sendende Fahrzeuge auf einer Karte sowie deren Zählung gehören. Dieser Wunsch wurde bei der späteren Implementierung berücksichtigt.

Das überarbeitete AP 1-Dokument wurde mit dem PA geteilt, um Feedback einzuholen bzw. diese ersten Projektergebnisse bekannt zu machen. In ähnlicher Weise wurden die weiteren APs bei den Forschungsstellen begonnen, dem PA über Web-Konferenzen und bilaterale Gespräche vorgestellt und Feedback bei der weiteren AP-Bearbeitung berücksichtigt.

Zuletzt wurde noch ein Vergleich zwischen der paketbasierten Kommunikation in V2X- und PROFINET-Systemen mit Speicherprogrammierbaren Steuerungen (SPS) gezogen. Während bei V2X Broadcasts über drahtlose Kommunikation vorherrschen, finden sich bei Feldbussystemen wie PROFINET Punkt-zu-Punkt-Verbindungen über Ethernet wieder. Zudem wird bei PROFINET erst eine Verbindung zwischen einer Steuerung (Controller) und einem Endgerät (Device) aufgebaut, bevor Daten übertragen werden. Dies ist bei V2X nicht nötig. Nichtsdestotrotz ließen sich Gemeinsamkeiten finden: Es existieren zyklische Kommunikation und Nachrichtenabfolgen, die bei Tests und Diagnosen nachgewiesen werden müssen. Bei PROFINET ist ein typisches Beispiel der Verbindungsaufbau (Connect Request und Response), bei V2X (ETSI-ITS-G5) die LSA-Priorisierung über SREM und SSEM (siehe Abbildung 1).

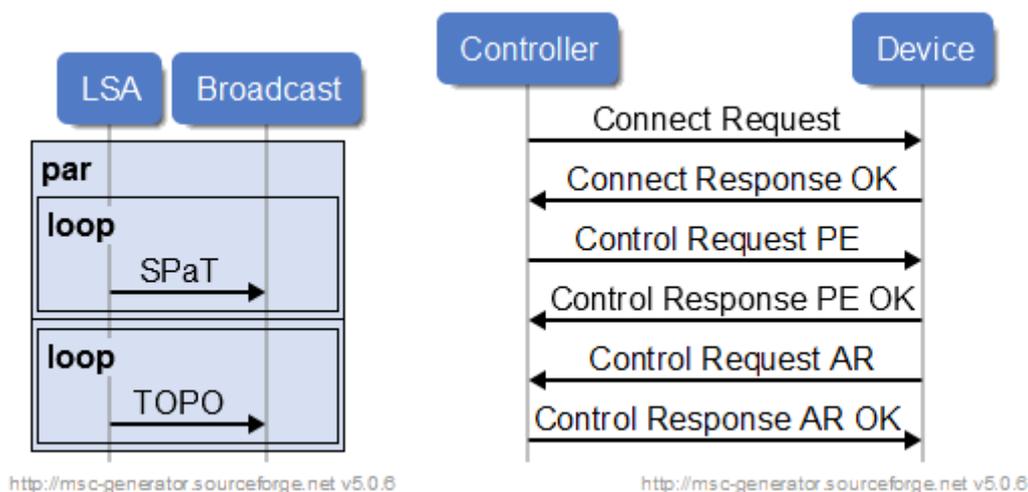


Abbildung 1: Mögliche Darstellung von Kommunikationssequenzen über UML-Sequenzdiagramme in einem V2X-System (links) sowie in einem PROFINET-System (rechts)

Die Schnittstellen zu Verkehrssystemen (AP 7) wurden in DIAKO zwar für WLANp-basierte Kommunikationssysteme ausgelegt. Die angeschlossenen Systeme für Regelerstellung und deren Auswertung würden aber prinzipiell mit Nachrichten aus anderen Domänen funktionieren, da auf dieser höheren Abstraktionsebene die Abfolge von allgemeinen Nachrichten die zentrale Rolle spielt. Dadurch wurde das DIAKO-Projekt seinem Anspruch gerecht, für allgemeine verteilte, kooperative Systeme vorbereitet zu sein.

AP 2: Spezifizierung der Gesamtsystemarchitektur und Festlegung des Testszenarios für den Demonstrator

Zunächst wurde auf der Grundlage der Ergebnisse aus AP 1 die Gesamtsystemstruktur spezifiziert. Durch die Voruntersuchungen im Zuge der Antragsstellung bestand bereits ein grundlegender Lösungsansatz, der auf zwei Anwendergruppen basiert: den Systemeinrichtern und den Endanwendern.

Darauf aufbauend kann die zu entwickelnde Systemarchitektur grob in die folgenden Bestandteile unterteilt werden:

- Modellierungssysteme für die Gruppe der Systemeinrichter mit Expertenwissen, die den Endanwendern spezifische Diagnosen zur Verfügung stellen. Es erfolgt eine weitere Unterteilung in
 - das Modellierungssystem für die zugrundeliegenden Diagnoseprozesse (Diagnose-, Wartungs- und Inbetriebnahmeprozesse), durch die der Endanwender geführt wird, sowie
 - das Modellierungssystem für Diagnoseregeln, über das sich Betriebsvorgaben des zu untersuchenden Verkehrssystems festlegen lassen.
- Diagnosesystem für die Endanwender, das mit Hilfe der Modellierungssysteme parametrisiert wird.
- Technische Schnittstellen zur verteilten Datenerfassung.
- Gesamtsystem, das die einzelnen Teilsysteme integriert.

Um die Fähigkeiten des zu entwickelnden Systems evaluieren und demonstrieren zu können (AP 11), war es erforderlich, eine Testanwendung als Beispiel zu erstellen. Hierfür wurden auf der Basis der Ergebnisse aus AP 1 die beispielhaften Diagnoseprozesse und -regeln festgelegt. Um die Testanwendung zum Zweck der Evaluierung (AP 11) ausführen zu können, wurde außerdem ein beispielhaftes Verkehrssystem („Testumgebung“) auf der Basis von Car2X/802.11p konzipiert. Es besteht aus mehreren Hardware-Modulen (On-Board-Units, Roadside Units) und Applikationen (Software bzw. eingerichtete Dienste).

Über verschiedene Architekturdiagramme wurden Struktur und Abläufe des geplanten Diagnosesystems definiert. Für die statische Sicht wurden die Kommunikationsteilnehmer sowohl des zu diagnostizierenden Systems (Roadside Units, On-board Units) als auch des DIAKO-Systems mit ihren möglichen Schnittstellen festgehalten, siehe Abbildung 2.

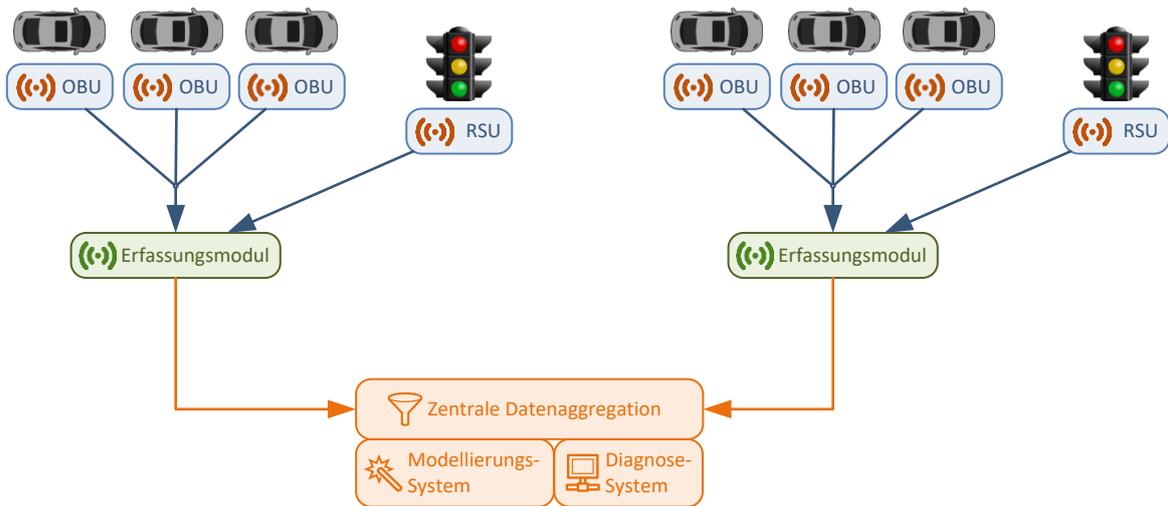


Abbildung 2: Statische Betrachtung der DIAKO-Systemarchitektur. DIAKO-Komponenten in Grün und Orange, zu diagnostizierendes System in Blau.

Die Schnittstelle zwischen Erfassungsmodulen und zu diagnostizierendem ITS-System ist IEEE 802.11p. Die Erfassung von Datenpaketen kann durch passives Mitlesen erfolgen, da V2X auf unverschlüsselten Broadcasts ohne Anmeldung basiert. Die Schnittstelle zwischen den Erfassungsmodulen und der zentralen Datenaggregation kann über Mobilfunk, Internet und generell IP-basierte Kommunikation erfolgen. Hier bietet sich das Kommunikationsprotokoll MQTT an, das erfasste Daten im JSON-Format und bei Bedarf über CBOR komprimiert überträgt.

Darüber hinaus wurde für die Planung des Ablaufs bei der Benutzung des DIAKO-Systems eine dynamische Sicht erstellt. In Abbildung 3 lässt sich vor allem die Unterteilung der DIAKO-Komponenten für die verschiedenen Nutzergruppen gut erkennen.

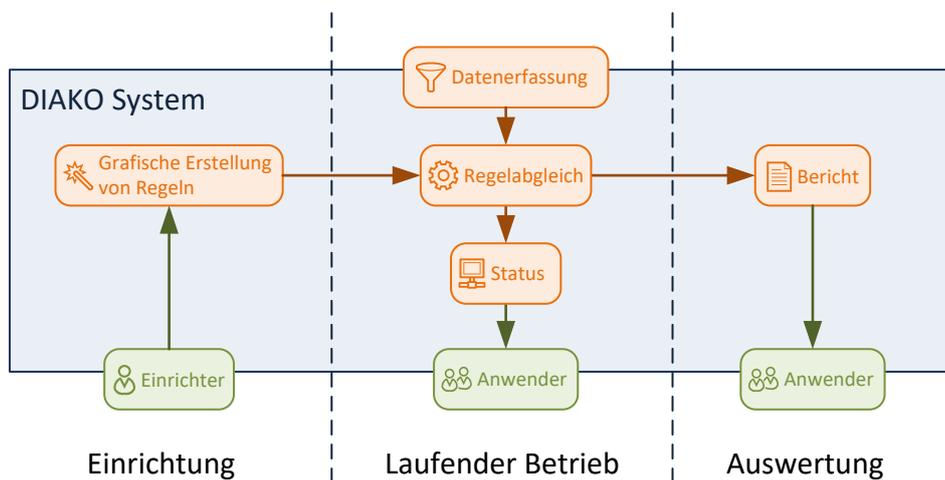


Abbildung 3: Dynamische Betrachtung der DIAKO-Systemarchitektur

Nicht zuletzt wurde in AP 2 ein Testszenario mit zugehöriger Testanwendung konzipiert, um in den späteren APs umgesetzt zu werden. Das Szenario basiert auf der in AP 1 als sehr relevant identifizierten LSA-Priorisierung über SREM- und SSEM-Nachrichten. Die Anmeldung soll von der RSU als gewährt beantwortet werden, wenn die Parameter Längengrad, Breitengrad und Liniennummer korrekt sind.

AP 3: Konzeption und Entwicklung von domänenspezifischen grafischen Sprachen für die Modellierungssysteme

Ausgehend von den Anforderungen und Anwendungsfällen aus AP 1 wurde untersucht, wie Diagnosen, Wartungen und Inbetriebnahmen in der Praxis durchgeführt werden können. Dafür wurden die einzelnen Prozessbestandteile identifiziert und in Ablaufgraphen miteinander verkettet, siehe Abbildung 4.

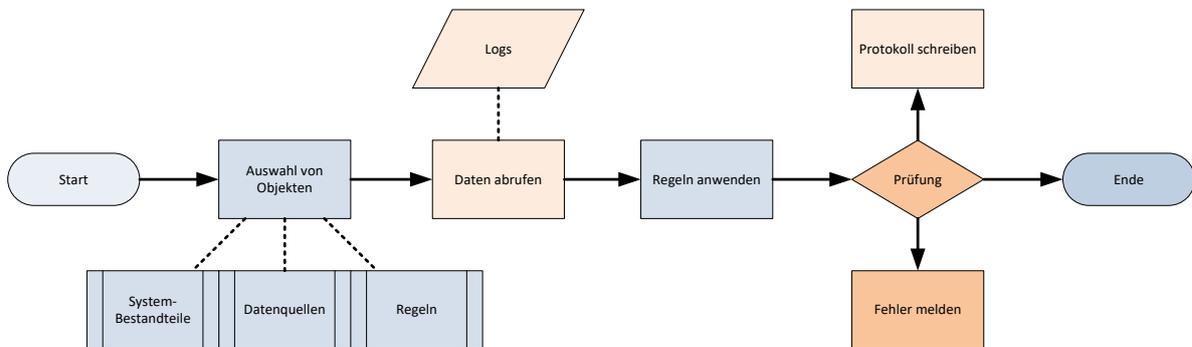


Abbildung 4: Verallgemeinerter Ablauf von Diagnoseprozessen

Ein Projektziel ist die Entwicklung eines Modellierungssystems für Diagnoseprozesse. Dem Nutzer soll ermöglicht werden, individuelle Abläufe zu definieren. Dafür wurde in diesem AP eine domänenspezifische grafische Sprache entwickelt. Diese basiert auf früheren Arbeiten der Forschungsstelle GFal und hat dort ihre Eignung für Diagnoseanwendungen bewiesen. Der Hauptteil der durchgeführten Arbeiten bestand demnach in der Übertragung der grafischen Sprache (siehe Abbildung 5) auf die neue Domäne der intelligenten Verkehrssysteme.

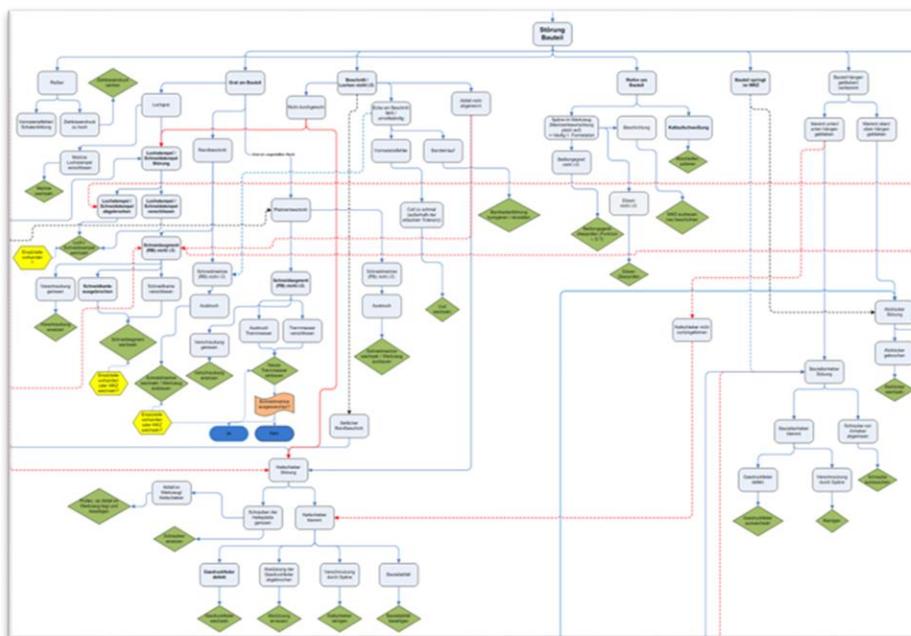


Abbildung 5: Diagramm für die Modellierung von Abläufen in der GFal-Software QuickSteps

Innerhalb dieser Sprache nehmen die Diagnoseregeln eine zentrale Rolle ein. Sie sollen in einem Modellierungssystem von Benutzern definiert und parametrisiert werden. Daher wurden mögliche

Regeln für die ausgewählten Anwendungsfälle untersucht und die Beziehungen zwischen Regeln und anderen Systemteilen modelliert. Anschließend wurde eine weitere domänenspezifische grafische Sprache für die Definition von Diagnoseregeln entwickelt.

AP 4: Konzipierung des Verfahrens zum automatischen Auswerten von Diagnosedaten

Über Schnittstellen (AP 7) zum zu diagnostizierenden Verkehrssystem werden Daten gesammelt, die gegen die Diagnosevorgaben (Regeln, AP 3) abzugleichen sind. Dies wird zur Darstellung des Systemzustandes (AP 5) benötigt. Der automatische Abgleich bzw. die Auswertung potenziell vieler, gleichzeitig gültiger Regeln muss über ein performantes Verfahren zur Laufzeit geschehen. Die entsprechenden Grundlagen wurden in diesem AP geschaffen und sind in Abbildung 6 dargestellt.

Die Herausforderung war, verschiedenartige Regeln zur Laufzeit parallel auswerten zu können. Dazu können sowohl das Prüfen einer Nachrichtenabfolge (die exakt eingehalten werden muss) als auch das Zählen von Verkehrsteilnehmern zählen. Eine etablierte Herangehensweise ist das Beschreiben der erwarteten Kommunikation über Modelle. Am ifak existierten Vorarbeiten zum Modellbasierten Testen (MBT) im Rahmen von Industrie- und V2X-Kommunikation. Dabei werden Anforderungen in Form von Modellen (z. B. Sequenzdiagrammen, Zustandsautomaten oder Petrinetzen) notiert und teilweise automatisch generiert. Die Ausführung dieser Modelle zur Laufzeit von Tests wurde im AP 4 konzeptionell untersucht und unterstützte später im AP 9 die automatische Auswertung.

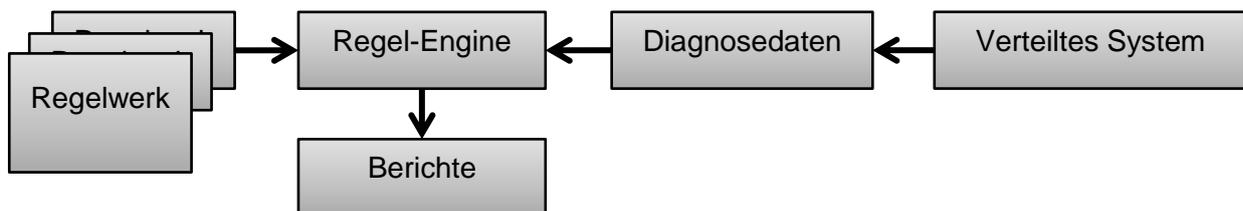


Abbildung 6: Grundlegender Ablauf der Diagnose

Da im Projekt Systeme mit paketbasierter Kommunikation (z. B. Ethernet, WLANp) vorausgesetzt wurden, konnte das zu diagnostizierende System als ereignisdiskretes, verteiltes System betrachtet werden. Mitgeschnittene Datenpakete sind beobachtbare Ereignisse, die aus der Ausführung von Programmen und Protokollen bei den Kommunikationsteilnehmern (z. B. Fahrzeuge oder RSUs) resultieren. Ohne eine zusätzliche Test- oder Diagnoseschnittstelle sind die passiv über das Kommunikationsmedium (z. B. Luftschnittstelle bei V2X) beobachtbaren Daten bzw. Pakete die einzig beobachtbaren Ereignisse.

In DIAKO war vor allem die Auswertung potenziell vieler parallel stattfindender Diagnose-Ereignisse vorgesehen. Für die Modellierung paralleler Abläufe sind sowohl UML-Sequenzdiagramme als auch Petrinetze sehr gut geeignet. Daher wurden in AP 4 Anforderungen aus den bereits identifizierten Anwendungsfällen in diesen grafischen Sprachen modelliert.

Es stellte sich heraus, dass sich Sequenzdiagramme gut für die bildliche Darstellung von Anwendungsfällen eignen. Im ingenieurwissenschaftlichen Bereich sind sie für diesen Zweck auch weit verbreitet. Petrinetze hingegen werden schnell zu komplex, um sie manuell aufzubauen. Sie eignen sich hingegen sehr gut dazu, über die Implementierung in einer Analyse-Engine die Abarbeitung von Anforderungen und Diagnoseregeln nachzuverfolgen. Über Marken

auf den Stellen lässt sich die erfolgreiche bzw. fehlerhafte Übertragung von V2X-Nachrichten modellieren. Durch Erreichen von Stellen mit vordefinierter Bedeutung lassen sich Entscheidungen treffen, die zu Diagnoseergebnissen führen, z. B. Erfolg, Warnung und Fehler.

Petrinetze sind gerichtete Graphen. Die einfachen Stellen-Transition-Netze bestehen aus Stellen zur Modellierung von Zuständen, aus Transitionen für Zustandsübergänge, gerichteten Kanten zwischen diesen Elementen sowie Marken, die Stellen belegen können. Durch „feuernde“ Transitionen können Marken durch das Netz über die verschiedenen Stellen bewegt werden. Im Projekt geschah die Zuordnung von Ereignissen zu den Transitionen. Es wurde weiterhin die automatische Netzerstellung und Abbildung von Regelementen auf die Netzelemente konzipiert.

Beim hierarchischen Aufbau der Petrinetze bot es sich an, sich an den wichtigsten Strukturelementen der UML zu orientieren, vor allem Parallelen (par), Schleifen (loop) und Alternativen (alt).

Es wurde weiterhin festgelegt, dass eine Kapselung von Teilnetzen über einzelne Start- und End-Stellen geschehen kann. Auf diese Weise lassen sich beispielsweise Kommunikationssequenzen modellieren.

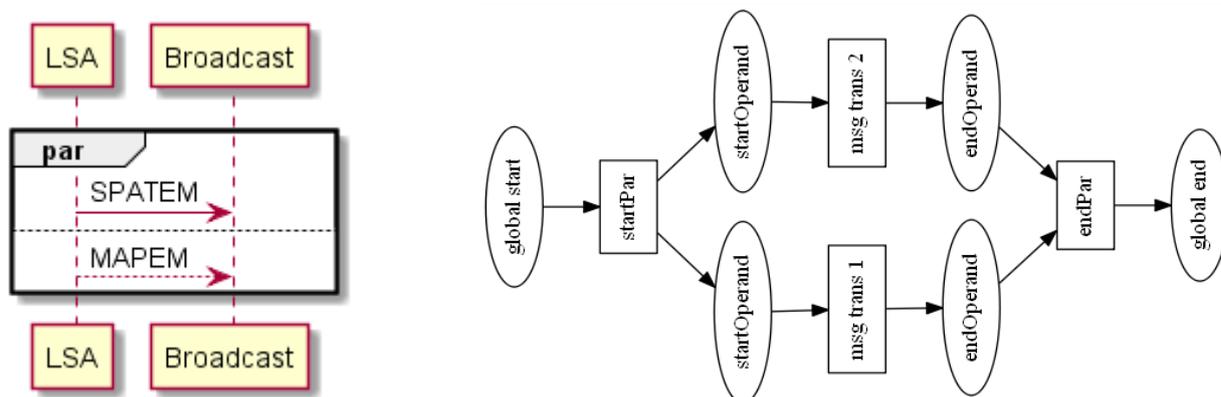


Abbildung 7: Paralleler Versand von V2X-Nachrichten im UML-Sequenzdiagramm (links) und mögliche Modellierung von parallelen Abläufen in einem Petrinetz (rechts).

Die obige Abbildung 7 zeigt am Beispiel einer parallelen Kommunikation das entsprechende Teil-Petrinetz. Die erfassten relevanten Diagnoseereignisse werden entsprechenden Transitionen zugeordnet (hier Nachrichten-Transitionen, msg trans). Die Schnittstellen zu anderen Teilnetzen werden durch jeweils eine Stelle zum Betreten (global start) und Verlassen (global end) des Teilnetzes gebildet. Nur an diese dürfen Transitionen anderer Teilnetze anschließen.

Nicht zuletzt soll eine ausgeführte Diagnose zum Auffinden von Systemfehlern führen. Im Fall von erfüllten oder verletzen Diagnosevorgaben sind Ereignisse zu erstellen und an eine Berichtsfunktion weiterzugeben. Dies konnte ebenfalls durch entsprechende Transitionen und Stellen definiert werden, die nach dem Feuern als zugeordnete Aktion ein Erfolgs- oder Fehlerereignis aussenden. Dies wird als Alternative zu anderen bekannten Methoden betrachtet, wie dem fallbasierten Schließen (Case-based Reasoning).

Mit den bisher beschriebenen Netz-Konstruktionen lassen sich zunächst nur Abfolgen von Nachrichten in einem Kommunikationssystem modellieren und zur Laufzeit überprüfen. In DIAKO wurden die Petrinetz-Modelle um Attribute und Timer erweitert, um beispielsweise Fahrzeuge zählen oder das Einhalten von Zykluszeiten prüfen zu können.

Am Beispiel der Diagnose von V2X-LSA-Daten geschieht der Ablauf wie folgt: Es wird erwartet, dass die RSU einer LSA zyklisch SPATEM und MAPEM versendet, und zwar jede Nachricht einmal in der Sekunde. Da sowohl die Uhren der RSU als auch des Rechners mit der Diagnose-Engine niemals exakt laufen werden, wurde eine Toleranz in der Zykluszeit berücksichtigt. Sobald ein Nutzer die Regel zum Überwachen der Zykluszeiten einer LSA-RSU definiert und die Diagnose startet, war die automatische Erstellung des entsprechenden Petrinetzes mit Schleifen- (loop) und Parallel- (par) Strukturen vorgesehen. Der Beginn der Diagnose würde über einen Startbefehl im grafischen Dashboard gegeben. Dann würde im internen Modell eine Marke auf die globalen Startstelle gelegt, um abzubilden, dass die Diagnose beginnen kann, jedoch noch keine Daten erhalten wurden.

Sobald die ersten V2X-Daten über eine WLANp-Hardwareschnittstelle erfasst werden, werden die Nachrichten dekodiert, ihr Typ (z. B. SPATEM) bestimmt, relevante Parameter gespeichert (z. B. ITS station ID des Senders) und ein Eingangszeitstempel erzeugt. Dann wird geprüft, ob diese Nachricht einem vordefinierten Ereignis entspricht, das eine Transition im Petrinetz zum Feuern bringen kann. Falls ja, dann werden entsprechend Marken bewegt, generiert oder vernichtet. Über Marken auf speziellen Systemzustandsstellen wird entschieden, welche Berichte an das Dashboard gesendet werden – z. B. „Zykluszeiten eingehalten“ oder „zu viele SPATEMs pro Sekunde erhalten“.

Zu Beginn des Projektes bestand der Wunsch, möglichst flexibel auf beliebige Nutzeranforderungen und Regeln reagieren zu können. Dies hätte eine sehr flexible Erstellung von Petrinetzen aus den Nutzerregeln erfordert. Die Herausforderung einer automatischen Netzerstellung ist, das Netz im Nachhinein verifizieren zu können. Die formale Modellverifikation ist ein eigenes Wissenschaftsgebiet und entsprechende Implementierungen hätten im Projekt keine Zeit gefunden. Daher wurde sich dazu entschieden, die wichtigsten Petrinetz-Strukturen im AP 9 vorzubereiten, vorab in der Diagnose-Engine zu hinterlegen und durch das Aktivieren von Nutzerregeln im Dashboard lediglich aufrufen zu müssen.

AP 5: Darstellungsformen der Diagnoseergebnisse

Die Arbeiten an AP 5 wurden mit einer Untersuchung von am Markt befindlicher Software begonnen. Dabei wurden sowohl Werkzeuge aus dem Bereich der intelligenten Verkehrssysteme (z. B. Vector CANoe.Car2X oder Nordsys WaveBEE Touch), Diagnosewerkzeuge (z. B. QuickSteps oder Wireshark), wie auch allgemeine Software (z. B. Office-Anwendungen) betrachtet. Hier konnten „best practices“ für die Gestaltung eines grafischen Nutzerinterfaces (GUI) gewonnen werden.

Aus den User Stories und Anwendungsfällen von AP 1 sowie den modellierten Diagnoseprozessen von AP 3 konnten dann die relevanten Diagnoseergebnisse und Systemzustände definiert werden. Ihre Darstellung wurde in Form von Papierprototypen entworfen. Alle Mockups gehen von einer Windows-Applikation aus, die aus verschiedenen Fenstern besteht. Hiermit hat der Nutzer die Möglichkeit, die Oberfläche nach seinen spezifischen Anforderungen zu gestalten und sich auf die aktuell wichtigen Informationen zu fokussieren.

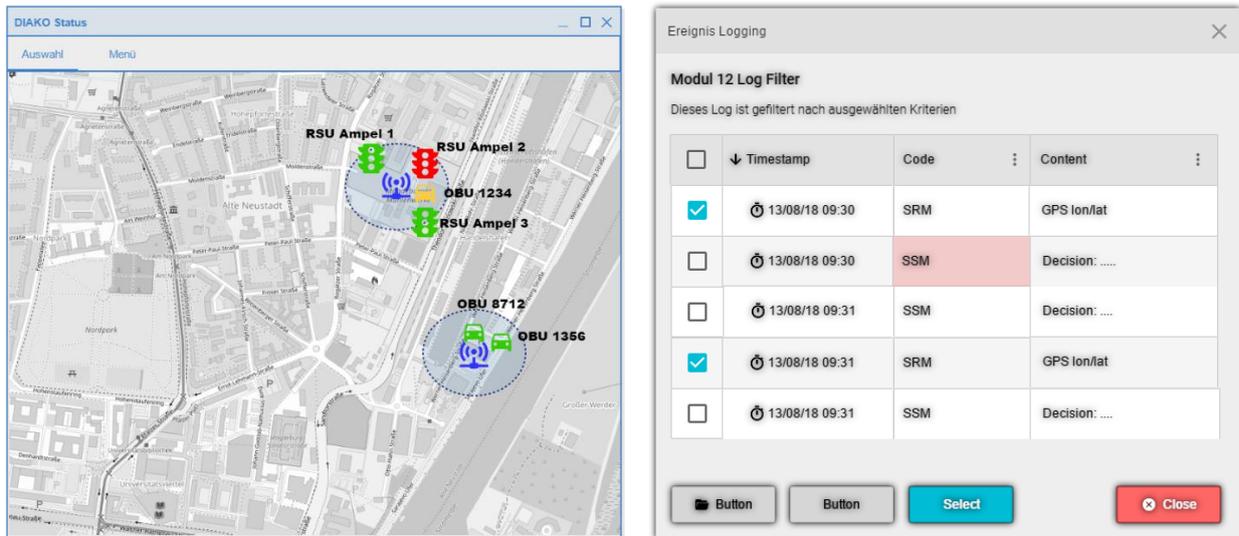


Abbildung 8: Mockups. Statusansicht für Systembestandteile (links) bzw. Protokollansicht für Ereignisse (rechts).

Der aktuelle Status von Systembestandteilen könnte auf einer Kartenansicht dargestellt werden (siehe Abbildung 8, links). Die Darstellung enthält einfache Informationen zu den DIAKO-Erfassungsmodulen und zu erfassten Verkehrsteilnehmern wie RSUs und OBUs. Die Nutzung eines Ampel-Systems (rot-gelb-grün) erlaubt dem Nutzer, direkt den Zustand des Systems zu erfassen. Detailliertere Informationen sind durch Auswahl eines Objekts abrufbar.

Für das Protokollieren von Ereignissen ist eine Listenansicht vorgesehen (siehe Abbildung 8, rechts). Die Anwendung von Filtern erlaubt es dem Nutzer, schnell konkrete Fragen zu beantworten oder Probleme zu lösen. Event-Logs können ebenfalls in verschiedenen Formaten exportiert und weiterverarbeitet werden.

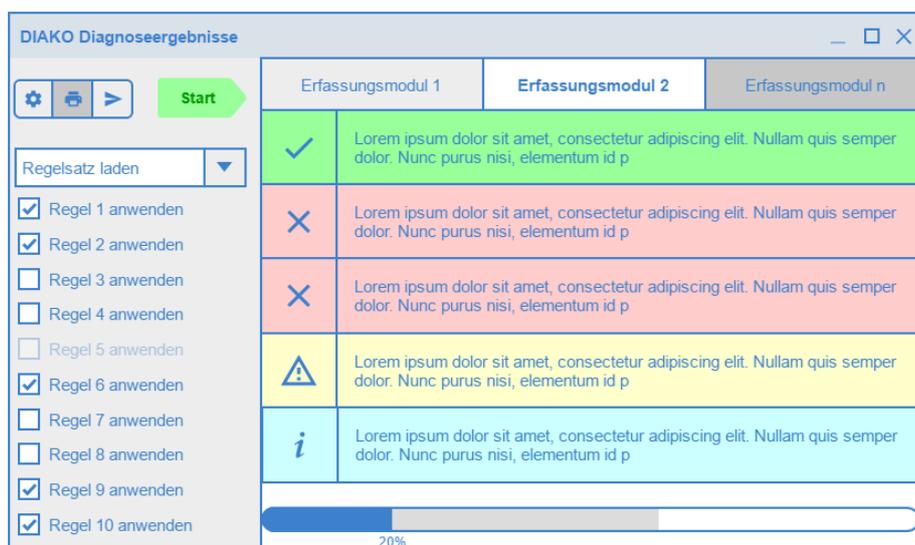


Abbildung 9: Ergebnisansicht für Diagnoseprozesse (Mockup)

Nach der Ausführung von konkreten Diagnosen und Regeln werden deren Ergebnisse ebenfalls als Ampelsystem dargestellt (siehe Abbildung 9). Ausführliche Dokumentationen der Diagnoseresultate erfolgen in Textform und/oder in weiteren Ansichten. Die Ergebnisse der

durchgeführten Arbeiten wurden mit den Anwendern im PA diskutiert und für die Umsetzung in späteren AP verfeinert.

AP 6: Entwicklung des Gesamtsystems bezüglich Modulschnittstellen und der Teilkomponente „Diagnosesystem“

Ausgehend von den Konzepten, die in den vorhergehenden APs erarbeitet wurden, begann in diesem AP die Entwicklung des Gesamt- und des Diagnosesystems. Um im Projektrahmen einen adäquaten Stand zu erreichen, wurde hierfür auf existierende Software-Frameworks zurückgegriffen. Der Projektpartner GFal verfügt über langjährige Erfahrungen in der Entwicklung von Anwendungen für schematische Darstellungen von verschiedenen Strukturen. Insbesondere die Implementierung von grafischen Sprachen kann so mit vertretbarem Aufwand gewährleistet werden. Ein vorhandenes Software-Framework bietet auch Hilfsmittel für die Gestaltung von grafischen Nutzerschnittstellen (GUI) mit geeigneten Nutzerinteraktionen.

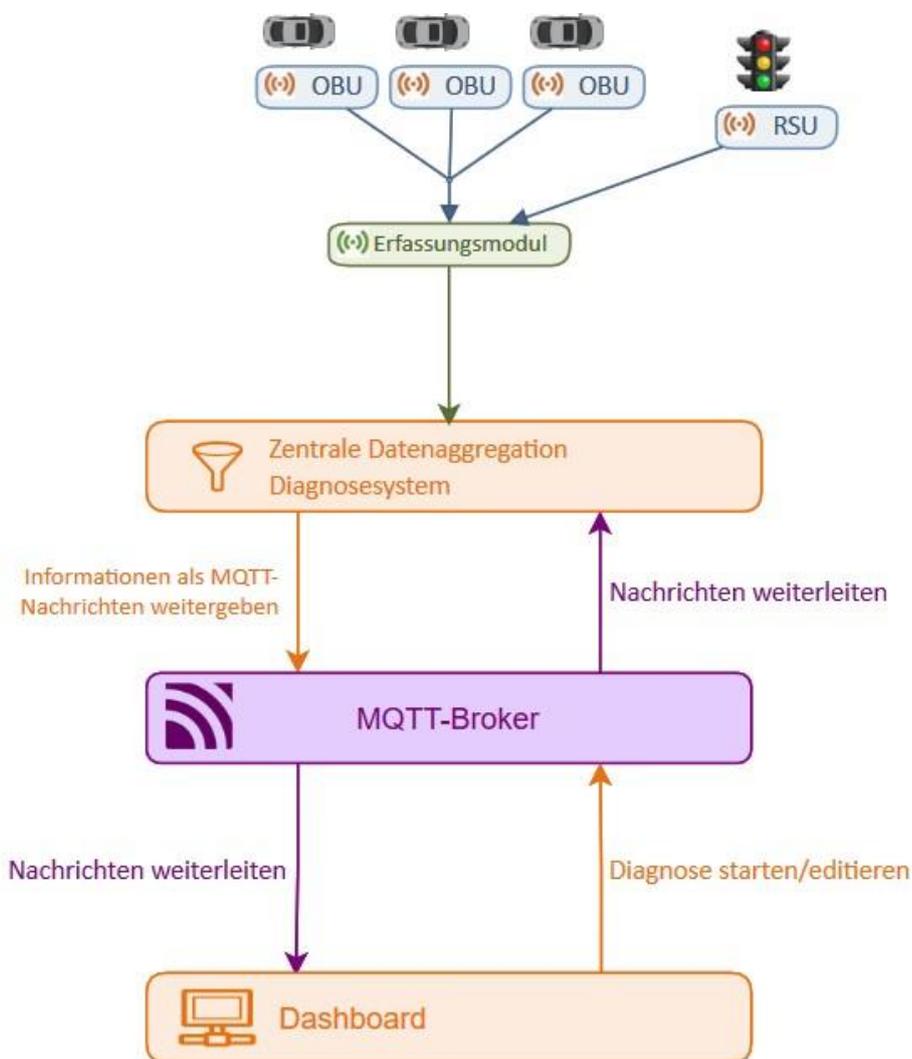


Abbildung 10: Schnittstellen ausgehend vom Diagnosesystem (hier: Dashboard)

Für das Gesamtsystem erfolgte in diesem AP vor allem die Entwicklung von Schnittstellen zwischen den einzelnen Systemkomponenten. Der modulare Aufbau des Gesamtsystems erfordert die Integration der einzelnen Teilsysteme, um deren Kommunikation und Datenaustausch zu gewährleisten. Als wesentliche Schnittstellentechnologie wurde dabei MQTT

eingesetzt. MQTT ist ein offenes Netzwerkprotokoll für die Übertragung von Nachrichten zwischen beliebigen Geräten. Dabei hält ein MQTT-Server (sogenannter „Broker“) die Kommunikation aller Partner. MQTT findet im Bereich IoT große Verwendung. Die Gesamtsystemarchitektur ist in Abbildung 10 skizziert.

Für alle beteiligten Systemkomponenten wurden MQTT-Clients integriert und ein gemeinsamer, projektweiter MQTT-Server wurde aufgesetzt. Letzterer verwaltet die Kommunikation zwischen den einzelnen Systembestandteilen. Als Payload der Kommunikation wurden JSON-Formate eingesetzt, siehe Abbildung 11. Diese sind sowohl menschenlesbar, als auch gut durch entsprechende Softwarebibliotheken unterstützt.



```
1 {
2   "report_type": "summary",
3   "reports": [
4     {
5       "report_type": "ok",
6       "diagnosis_rule_id": 12345,
7       "affected_its_station_ids": [
8         123,
9         456
10      ]
11    },
12    {
13      "report_type": "warning",
14      "diagnosis_rule_id": 67890,
15      "affected_its_station_ids": [
16        78,
17        90
18      ]
19    }
20  ]
21 }
```

```
1 {
2   "road_user_id": 12345,
3   "type": "car",
4   "pos_latitude": 52.137732,
5   "pos_longitude": 11.652019,
6   "speed_mps": 18.0,
7   "timestamp": 1605113381
8 }
```

Abbildung 11: JSON-Daten als Kommunikation im DIAKO-System (Beispiele)

Für das Diagnosesystem wurde die Darstellung der Dialoge für die Endanwender zusammen mit entsprechenden Interaktionen implementiert. In einem ersten Schritt wurde eine prototypische C++-Anwendung implementiert, die sowohl die Nutzeroberfläche als auch die zugrundeliegende Schnittstellentechnologie validieren sollte, siehe Abbildung 12.

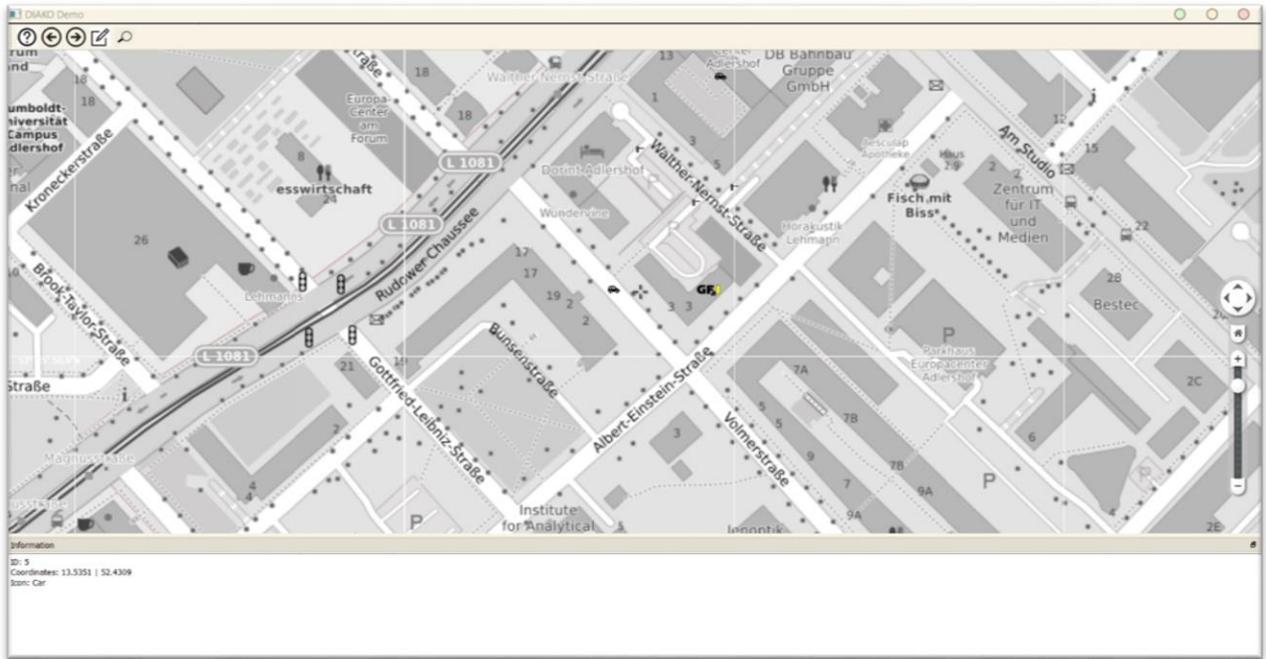


Abbildung 12: Prototypische C++-Anwendung für das Diagnosesystem

Die Umsetzung der Kartendarstellung als zentrales Element für den Benutzer war unter Verwendung gängiger C++-Softwareframeworks sehr komplex. Daher wurde – im Projektteam und nach Rücksprache mit den PA-Teilnehmern – entschieden, ein Dashboard in Form einer Weboberfläche zu entwerfen, siehe Abbildung 13 und Abbildung 14. Diese kann einfach von verschiedensten Endgeräten bedient werden und erlaubt es, den Systemzustand auf einen Blick zu erfassen und Fehlerzustände zu erkennen.

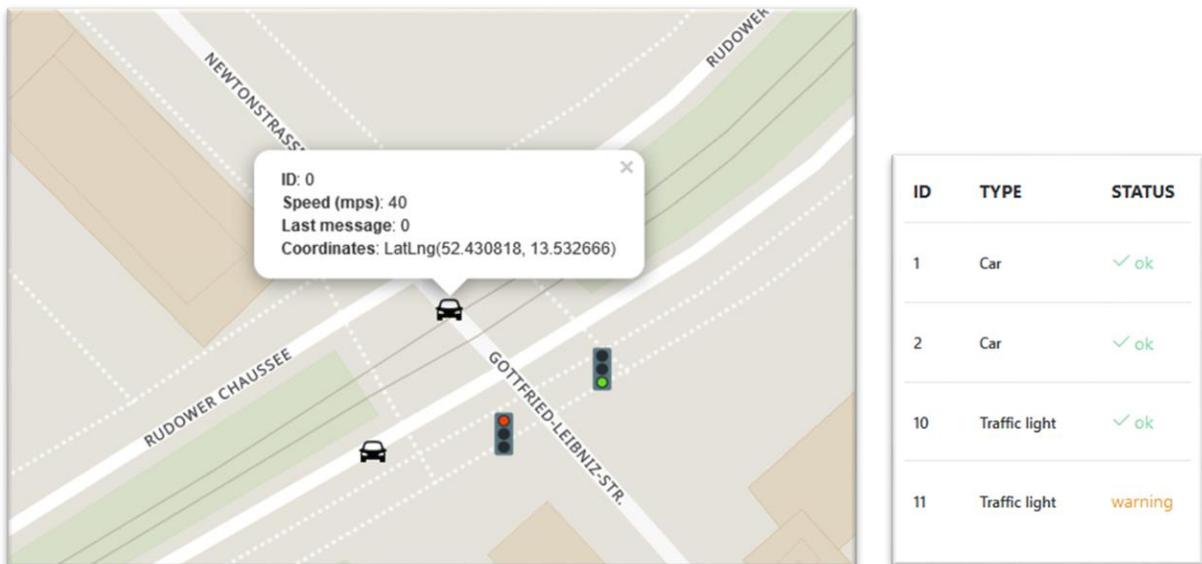
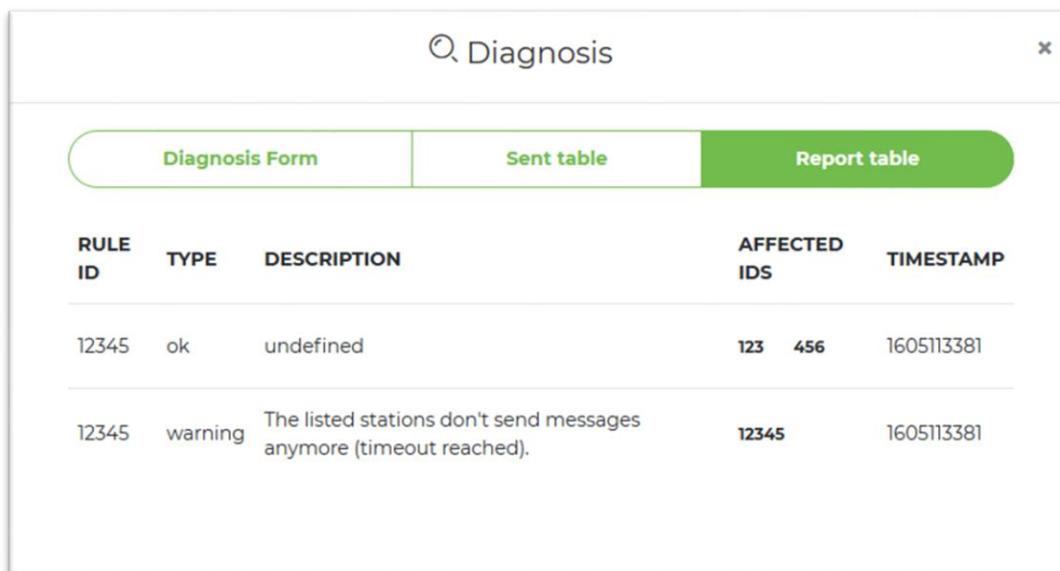


Abbildung 13: Prototypisches Dashboard für das Diagnosesystem (Dialogbeispiele)



RULE ID	TYPE	DESCRIPTION	AFFECTED IDS	TIMESTAMP
12345	ok	undefined	123 456	1605113381
12345	warning	The listed stations don't send messages anymore (timeout reached).	12345	1605113381

Abbildung 14: Prototypisches Dashboard für das Diagnosesystem (Dialogbeispiel)

AP 7: Entwicklung und Implementierung der Schnittstellen zu technischen Komponenten von intelligenten Verkehrssystemen

In diesem AP wurden die konzeptionellen Grundlagen des AP 2 in Software umgesetzt, um Echtzeit-Daten aus WLANp-basierten Systemen erfassen und verarbeiten zu können.

Die Schnittstelle zum DIAKO-System bilden die V2X-Nachrichten, die als Broadcast und unverschlüsselt versendet werden. Dies ist das derzeit vorgesehene Konzept für den öffentlichen Bereich, in dem Nachrichten diskriminierungsfrei an jedem Verkehrsteilnehmer versendet werden sollen. Zusätzlich können sie signiert sein, um mittels einer Public Key Infrastructure (PKI) die Echtheit von Sendern bestätigen zu können (Authentifizierung). Weiterhin begann in 2019 mit dem neuen VW Golf das Rollout eines weiteren Fahrzeugmodells, welches ein Funkmodul auf dem IEEE 802.11p-Status („WLANp“) besitzt. Die darauf aufbauenden Nachrichten sind in Europa durch die ETSI standardisiert. Entsprechende Standards werden fortlaufend aktualisiert und sind über die Website der ETSI frei erhältlich.

Das DIAKO- (Demonstrator-) System benötigt letztendlich eine Schnittstelle, um die drahtlosen WLANp-Nachrichten empfangen, nach ETSI-Standard dekodieren und verarbeiten zu können. Dabei wurde absichtlich (nur) auf WLANp und nicht etwa eine extra einzurichtende Testschnittstelle (per WLAN, Ethernet o. Ä.) abgezielt. Dies sollte die Übertragbarkeit der Diagnose auf beliebige Anwendungsfälle und Testfelder ohne zusätzlichen Installationsaufwand am zu diagnostizierenden System sicherstellen. Das DIAKO-System sollte nur die Nachrichten bzw. Datenpakete passiv empfangen, die im Normbetrieb vom Verkehrssystem versandt werden. Es sollte nicht nötig sein, das System (z. B. OBUs in Fahrzeugen) zur Vorbereitung in aufwendige Testzustände zu versetzen.

Prinzipiell kann jeder (bewegliche) Verkehrsteilnehmer eine mobile ITS-Station erhalten. Sie wird dann als On-board Unit (OBU) bezeichnet. In der Regel sind Fahrzeuge des motorisierten Individualverkehrs (MIV) und des Öffentlichen Personennahverkehrs (ÖPNV) vorgesehen. Es gibt aber auch Ansätze, Radfahrer und andere schwache Verkehrsteilnehmer mit OBUs auszustatten.

Prinzipiell kann jede (unbewegliche) Infrastruktur eine fest installierte ITS-Station erhalten. Sie wird dann als Roadside Unit (RSU) bezeichnet. Installationsorte sind häufig städtische Kreuzungen (bzw. „Verkehrsknoten“) zur Anbindung an dortige Lichtsignalanlagen (LSAs) oder Schilderbrücken auf Autobahnen.

RSUs müssen sich von der grundlegenden Hardware her nicht von OBUs unterscheiden. Der Unterschied besteht in der Regel in einem wetterfesten Gehäuse sowie der Software-Konfiguration. Als Beispiel sei die MK5 von Cohda Wireless genannt, die es sowohl als OBU als auch als RSU gibt. Auf beiden Hardware-Varianten kann die gleiche V2X-Anwendung laufen. Den Unterschied, welche Nachrichten versandt werden, macht dort nur die Verwendung einer spezifischen Konfigurationsdatei. Dementsprechend konnte davon ausgegangen werden, dass das DIAKO-System eine einheitliche drahtlose Schnittstelle zu OBUs und RSUs besitzen kann.

Die Datenerfassung wurde über WLANp-fähige und als Erfassungsmodule bezeichnete Hardware konzipiert. Diese wurden als eigenständige Geräte definiert, die ein Anwender straßenseitig installieren müsste, um passiv die Daten von V2X-Sendern in der näheren Umgebung erfassen zu können. Dadurch wurde kein Eingriff in das zu diagnostizierende System (Fahrzeuge, RSUs etc.) erforderlich. Am ifak existieren zum einen kommerzielle Erfassungsmodule der Firma Cohda Wireless (MK5-Modelle) als auch Eigenentwicklungen (ifak-OBUs). Die folgende Abbildung 15 zeigt Beispiele aus früheren Projekten, in denen die Erfassungsmodule des ifak in Fahrzeugen oder straßenseitigen Schaltschränken untergebracht wurden, um vor Ort Daten passiv mitzuschneiden. Prinzipiell eignen sich all solche Module, auf denen ein Anwender eigene Software installieren und betreiben kann, um auf die auf die WLANp-Schnittstelle (Radiomodul für IEEE 802.11p bei 5,9 GHz) zugreifen zu können.

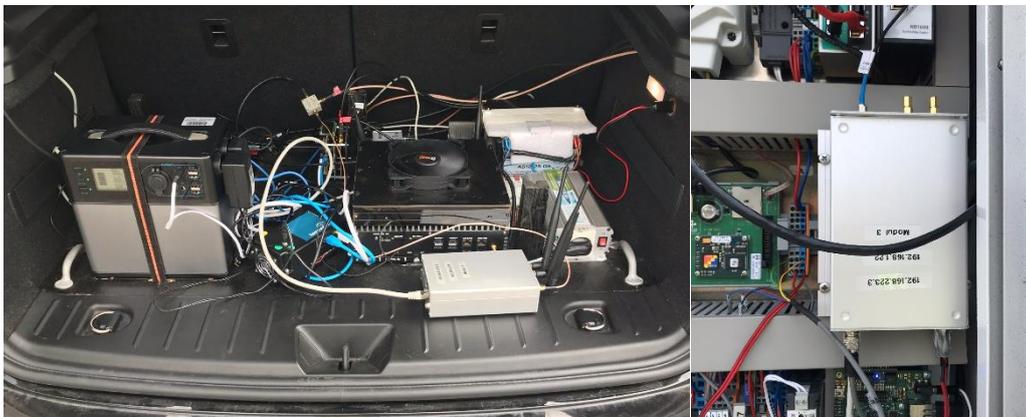


Abbildung 15: Beispielhafte Integration jeweils eines Erfassungsmoduls des ifak in einem Fahrzeug (links) sowie einer Streckenstation (rechts)

Weiterhin wurde im AP 7 untersucht, welche Softwarelösung sich für die Erfassung und Dekodierung von V2X-Nachrichten direkt auf den Erfassungsmodulen eignet. Mit der beim ifak vorhandenen Hardware bestanden folgende grundlegende Möglichkeiten:

- Nutzen der freien Programme tshark oder tcpdump (auf denen auch die weit verbreitete Diagnosesoftware Wireshark aufbaut) und Verarbeiten ihrer Ausgaben
- Nutzen eines kommerziellen Kommunikationsstacks (z. B. von Cohda Wireless) und verfügbarer Software-Entwicklungshilfen
- Nutzen von Open-Source-Kommunikationsstacks (z. B. OpenC2X oder VANETZA)
- Nutzen von Python und zusätzlichen freien Python-Modulen, die einen tshark-Wrapper implementieren (z. B. pyshark)

- Nutzen der LLC-Lib von Cohda Wireless, um zunächst nur rohe Bytestreams zu erfassen und im Anschluss selbst zu dekodieren

Alle Lösungen basieren auf dem Fakt, dass V2X-Nachrichten entsprechend dem OSI-Referenzmodell aus einzelnen Protokollschichten aufgebaut sind. Dazu zählen vor allem Logical Link Control (LLC), GeoNetworking und der ETSI-Payload mit den eigentlichen für DIAKO relevanten Daten. Die Implementierung einer Dekodierung von Hand allein über die Standardisierungsdokumente der ETSI und der ISO wäre zwar möglich, jedoch extrem aufwendig. Die Kodierung erfolgt textuell über den Standard ASN.1 und wird bei V2X im sogenannten Unaligned-PER-Format drahtlos übertragen. Hierbei werden die Bit so eng wie möglich gepackt, um die entstehenden Datenpakete so klein wie möglich zu machen.

Es existieren sowohl kommerzielle als auch freie ASN.1-Compiler, die aus den ISO- und ETSI-Standards zum Beispiel C-Code erzeugen, um über WLANp erfasste Bytestreams automatisch dekodieren zu können, was die Implementierung einer Dekodierung von Hand erübrigt. Das prinzipielle Vorgehen wurde im AP 7 mithilfe des freien Compilers [asn1c](https://github.com/vlm/asn1c) (<https://github.com/vlm/asn1c>) und einigen ETSI-Standards wie CAM und DENM untersucht. Nachrichten wie SPATEM und MAPEM basieren jedoch auf den ISO-Standards SPAT und MAP, deren ASN.1-Definitionen nicht frei erhältlich sind. Außerdem erwies sich die Erstellung und Nutzung des Dekodierers im Zusammenhang mit der LLC-Lib als recht aufwendig, weshalb dieser Ansatz für die weitere Projektbearbeitung verworfen wurde.

Das Konsolenprogramm tcpdump kann zwar seit einiger Zeit V2X-Daten dekodieren, da die ETSI-Standards nachgepflegt wurden. Jedoch ist zum einen eine WLANp-Schnittstelle entsprechend vorzubereiten, zum anderen müssten die Ausgaben interpretiert und in einer eigenen Software ausgewertet werden. Besser zu warten und auch effizienter ist die Nutzung von Bibliotheken mit PCAP-Schnittstelle, die sich direkt in eigene Programme integrieren lassen. Im AP 7 wurde mit pyshark ein freies Modul der Programmiersprache Python untersucht. Hier stellte sich aber als hinderlich heraus, dass es nicht ohne weiteres auf der am ifak vorhandenen Hardware zu betreiben war. Die Installation war lediglich auf einem Windows-Entwicklungsrechner erfolgreich, auf dem mithilfe von pyshark Aufzeichnungen im PCAP-Format untersucht werden konnten, jedoch keine Live-Schnittstelle für WLANp zur Verfügung stand.

Die bisher diskutierten Lösungen waren alle für das passive Dekodieren von V2X-Daten vorgesehen, was in DIAKO völlig genügt hätte. V2X-Kommunikationsstacks ermöglichen hingegen das Empfangen und dynamische Versenden von Nachrichten. Freie Lösungen wie OpenC2X und VANETZA haben derzeit noch das Problem, nicht auf jeder Hardware zu laufen und keine Dekodierung für alle V2X-Protokollschichten zu bieten. In der Regel fehlt der LLC-Layer, der für das Lesen live erfasster Daten über WLANp erforderlich ist.

Letztendlich wurde sich in DIAKO dazu entschieden, auf Vorarbeiten mit dem V2X-Stack von Cohda Wireless aufzubauen, den das ifak bereits erworben hatte und für den es ein Software Development Kit (SDK) besitzt. Eine Beispiel-Software des SDK wurde genutzt, um V2X-Nachrichten auf der verfügbaren Hardware zu erfassen und zu dekodieren. In DIAKO wurde die nun erweitert, um die meisten erfassten Rohdaten ausgewählter Nachrichtentypen in JSON-Strukturen zu packen und über MQTT der Diagnose-Engine zur Verfügung zu stellen.

Zur besseren Wartbarkeit wurde sich dazu entschieden, diese Engine in einem eigenen Programm auszulagern. Durch die MQTT-Schnittstelle kann die Diagnose-Engine zudem direkt auf den Erfassungsmodule als auch einem Entwicklungsrechner laufen. Dieses Programm wurde im Rahmen von AP 9 in Python implementiert.

Als Beispiel dient eine reale SPATEM, die durch die V2X-Erfassungssoftware über WLANp empfangen, dekodiert und folgendermaßen als JSON-String über MQTT bereitgestellt wurde:

```
{ "v2x_message": { "station_id": 12345, "intersections": [ { "id": 1986,
"states": [ {
  "signal_group_id": 1, "state_time_speed_list": [ { "timings": {
"likely_time": 13750, "max_end_time": 13980, "min_end_time": 13730 },
"event_state": 6 } ] } ] } ] }, "obu_timestamp_sec": 1611130974.067125
}
```

Diese Daten sind für Menschen leicht lesbar, verbrauchen jedoch relativ viel Speicher bei der Übertragung. Als mögliche Verbesserung wurde das CBOR-Format im Projekt untersucht. Nach diesem Standard (RFC 8949) lassen sich Eingangsdaten wie JSON-Strings serialisieren, platzsparend als Bytestream übertragen und beim Empfänger wieder deserialisieren. Folgendes allgemeines JSON-Beispiel findet sich im Standard:

```
{_ "Fun": true, "Amt": -2} → 0xbf6346756ef563416d7421ff
```

Für den Fall, dass die Erfassungsmodule ihre Daten per Mobilfunk bereitstellen, wäre CBOR eine sinnvolle Optimierung. Sie wurde im Projekt nicht implementiert, wäre aber bei einer Weiterentwicklung der Software nach dem Projekt zu diskutieren.

AP 8: Entwicklung und Implementierung des Teilsystems zur Modellierung von Diagnoseprozessen

In diesem AP wurde das erste der in AP 3 konzipierten Modellierungssysteme softwaretechnisch umgesetzt. Ein wesentlicher Bestandteil ist die Implementierung einer neuen grafischen domänenspezifischen Sprache in einem grafischen Editor. Wichtige Aspekte bei der Entwicklung von Editoren sind gute Interaktionsunterstützungen und Visualisierungen. Hierzu müssen geeignete Layoutverfahren entwickelt und implementiert werden. Das entwickelte Teilsystem zur Modellierung von Diagnoseprozessen wird in das Gesamtsystem integriert.

Das Modellierungssystem für Diagnoseprozesse wurde als Erweiterung der existierenden Software QuickSteps prototypisch implementiert. Der Vorteil der Anwendung von QuickSteps bestand in der schnellen Entwicklung von wesentlichen Programmbestandteilen. Insbesondere bieten das existierende Softwarewerkzeug und -framework bereits geeignete Eingabemöglichkeiten für Nutzer. Als Qualifizierung von QuickSteps wurde die grafische domänenspezifische Sprache aus AP 3 in QuickSteps umgesetzt. Damit ist das prototypische Werkzeug in der Lage, Diagnoseprozesse für intelligente Verkehrssysteme abzubilden und ein Systemeinrichter kann neue Prozesse modellieren oder bestehende Prozesse editieren. Die Modellierung erfolgt dabei durch ein Baukastensystem, in dem grafische Elemente auf einer Zeichenfläche kombiniert werden (siehe folgende Abbildung 16). Notwendige Parameter werden über entsprechende Dialoge eingegeben.

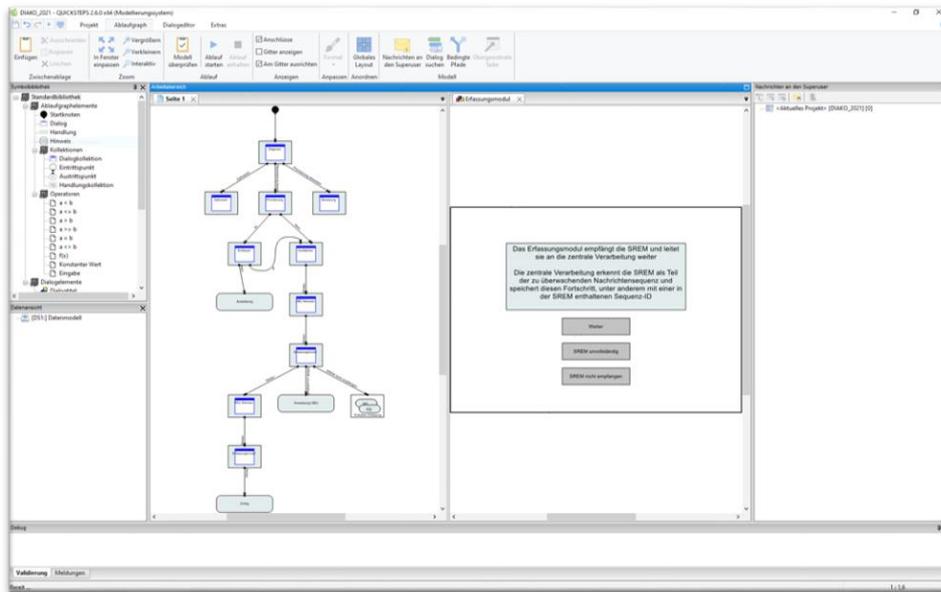


Abbildung 16: Modellierung von Diagnoseprozessen in QuickSteps

Die Integration des Modellierungssystems in das Gesamtsystem erfolgt über Datenbankschnittstellen, siehe Abbildung 17. In der Modellierung von Diagnoseprozessen gibt es verschiedene Objekte, die zwischen einzelnen Systembestandteilen ausgetauscht werden müssen. Zuerst sind die eigentlichen Diagnoseregeln zu nennen. Diese müssen einerseits modelliert werden und andererseits für die Diagnoseauswertung zur Verfügung stehen. Die weiteren Objekte in Diagnoseprozessen sind Fehlerfälle und Handlungsanweisungen für den Nutzer des Diagnosesystems. Diese müssen ebenfalls modelliert werden und auf dem Dashboard aus AP 7 dargestellt werden.

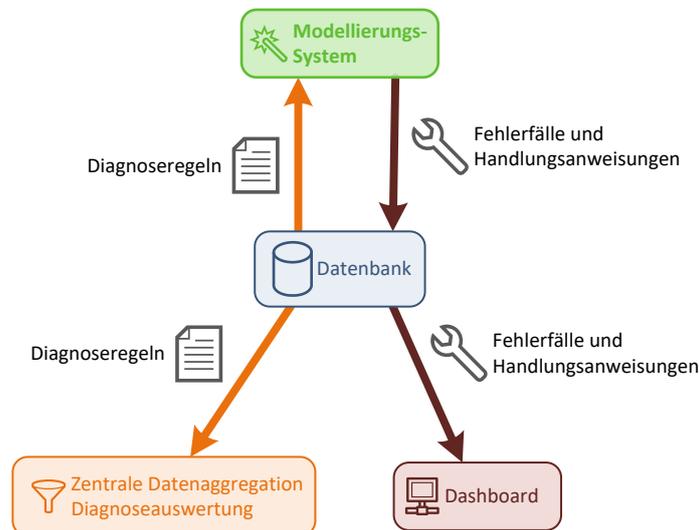


Abbildung 17: Schnittstellen für die Modellierung von Diagnoseprozessen

Daher sind zwei Schnittstellen vom Modellierungssystem vorzusehen und zur Vereinheitlichung wurde eine Datenbank als Zwischenschicht zur Datenhaltung angelegt. Das Softwarewerkzeug QuickSteps kann Datenbanken anbinden und Daten dort ablegen.

AP 9: Entwicklung und Implementierung des Teilsystems zur Modellierung sowie automatisierten und hoch performanten Auswertung von Diagnoseregeln

In AP 4 wurde mit den Petrinetzen ein mögliches Konzept zur Auswertung von V2X-Daten zur Laufzeit vorgestellt. Es wäre ein enormer Aufwand entstanden, Modelle für jegliche Nutzervorgaben dynamisch zu erstellen. Daher wurden in der Implementierungsphase Petrinetzmodelle für vorbereitete Regeln wie die Zykluszeitüberprüfung von SPATEMs erstellt. Parametriert ein Nutzer solch eine Regel über das Dashboard, so werden die Parameter (z. B. eine SPATEM-Zykluszeit) in der zentralen Datenbank hinterlegt. Nach dem Start der Diagnose durch den Nutzer werden die vorbereiteten Petrinetze samt Timern parametriert und die Markenbewegung kontinuierlich überwacht.

Die Diagnoseregeln für ein konkretes intelligentes Verkehrssystem sind stark von den Eigenschaften dieses Systems abhängig. Insbesondere die Quantität und Qualität der enthaltenen technischen Komponenten wirken sich entscheiden darauf aus, welche Regeln sinnvoll betrachtet werden können. Für die prototypische Umsetzung im Forschungsprojekt wurden daher Diagnoseregeln in Textform festgelegt. Die Modellierung durch einen Endanwender erfolgte rein in Form einer Parametrierung. Diese wurde im Rahmen des Dashboards (Diagnosesystem, AP 7) umgesetzt und nicht in einem separaten Softwarewerkzeug. Dadurch konnte eine benutzerfreundliche Bedienung bei hoher Komplexität beibehalten werden.

Die Datenerfassung in den Erfassungsmodulen wurde in C programmiert, da auf einer Beispielsoftware aus dem SDK von Cohda Wireless aufgebaut werden konnte. Die Diagnose-Engine mit den Petrinetzmodellen und Regelauswertungen zur Laufzeit wurde dagegen in Python programmiert. Für beide Sprachen existieren MQTT-Clients (mosquitto bzw. paho-mqtt), über die der Datenaustausch der verteilten Anwendungen realisiert wurde.

AP 10: Erzeugen der Testanwendung mit dem Software-Demonstrator

Durch die Vorarbeiten in AP 1 und 2 wurde eine Testanwendung definiert. Aus den Anwendungsfällen und Gesprächen mit dem PA ergab sich, dass die Überwachung von über V2X bereitgestellten LSA-Informationen sehr praxisrelevant ist und am besten für den Software-Demonstrator geeignet wäre.

Nachdem eine erste Version des prototypischen Diagnosesystems fertiggestellt war, konnte die Testanwendung in AP 10 erzeugt werden. Das ifak trat in der Rolle als Systemeinrichter auf, definierte Abläufe im erweiterten QuickSteps sowie Regeln für die Datenbank, auf die die Diagnose-Engine zugreifen kann. Jede Regel hat eine ID, über die eine Regel im Dashboard gestartet und gestoppt werden kann. In einer Regel ist etwa hinterlegt, wann der Diagnose Timer zu starten ist, um die Zykluszeit von 1 Sekunde zu prüfen. Dabei wurden Toleranzen berücksichtigt, sodass erst beim Unterschreiten von 0,75 s bzw. dem Überschreiten von 1,25 s eine Warnung an das Dashboard gesendet wird. Ebenfalls ist in der Testanwendung hinterlegt, dass alle 3 Sekunden ein Bericht (Status „OK“ oder „Warnung“) an das Dashboard gesendet wird.

Als Nächstes wurde am ifak eine Roadside Unit (RSU) der Firma Cohda Wireless aufgebaut und derart parametriert, dass sie für das Demoszenario passende V2X-Nachrichten verschickt. Sie simuliert über SPATEM und MAPEM die Daten einer Lichtsignalanlage. Zusätzlich wurde eine

ifak-OBU als V2X-Erfassungsmodule mit der Erfassungssoftware und der Diagnose-Engine bespielt und in der Nähe der RSU platziert. Das Hardware-Setup ist in Abbildung 18 dargestellt. Außerdem wurde das Dashboard auf einem PC installiert, um über einen Browser abrufbar zu sein.



Abbildung 18: Erweiterter Demonstrator zur Diagnose eines V2X-Systems

Über die derart vorbereitete Hardware und Software waren somit alle Voraussetzungen erfüllt, um im nachfolgenden AP die Evaluierung zu starten. Zusätzlich wurden einmalig die Daten der RSU im JSON-Format aufgezeichnet. Von nun an konnten SPATEMs über ein Skript per MQTT eingespielt und simuliert werden. Dies erleichterte die Evaluierung, da diese zwar mit der realen RSU ausgeführt werden konnte, jedoch von der Hardware nicht zwingend abhängig war und sich das Testszenario rein durch Software simulieren ließ.

AP 11: Evaluierung des Gesamtsystems anhand der Demonstrator-Anwendung

Zuletzt konnte in AP 11 anhand der Beispiel-Anwendung das Gesamtsystem evaluiert werden. Als Erstes wurde die Software bei den Forschungsstellen ausgeführt. Dafür wurde ein öffentlicher MQTT-Broker (test.mosquitto.org) verwendet, um die Diagnose-Engine beim ifak in Magdeburg und das Dashboard bei der GFal in Berlin betreiben zu können. Die Voraussetzung dafür war, dass der Broker MQTT über TCP für die Diagnose-Engine und MQTT über Websockets für das Dashboard unterstützt.

Nach der Vorstellung der Software beim 3. PA-Meeting (online per MS Teams) am 23.02.2021 wurde sie auch an die PA-Mitglieder verteilt. Hierfür wurde die Diagnose-Engine als nutzerfreundliche Windows-Anwendung (.exe) zur Verfügung erstellt, um die sonst erforderliche

Installation von Python nebst mehreren Python-Modulen per Textkommandos zu umgehen. Auch wurde eine Konfigurationsdatei für einen lokalen Mosquitto-MQTT-Broker bereitgestellt, um TCP und Websockets zu unterstützen. Letztendlich entstand ein Software-Bundle aus der Mosquitto-Config, dem Skript zum Senden der einmalig aufgezeichneten V2X-Daten im JSON-Format, der Diagnose-Engine (.exe) und dem Dashboard (HTML und Javascript für den Browser). Allen PA-Mitgliedern wurde dieses Bundle über einen Download zur Verfügung gestellt. Über Aufzeichnungen der ifak-RSU im JSON-Format konnten die Partner auch ohne entsprechende Hardware das Szenario bei sich abspielen und die Ergebnisse in der GUI beobachten.

Von einigen PA-Mitgliedern wurde die Software ausprobiert und allgemein positives Feedback mitgeteilt. Bei einigen Partnern wird über die firmeninterne Weiterentwicklung nachgedacht. Zunächst ist aber eine interne Vorstellung der DIAKO-Ergebnisse bei den PA-Mitgliedern geplant, um den Software-Ablauf noch mehr Personen vorzuführen. Daher wird weiteres Feedback erst nach dem Projektende erwartet.

Aufgrund der Reisebeschränkungen in der Corona-Pandemie wurde die Evaluation intensiver bei den Forschungsstellen als – wie ursprünglich geplant – bei den Partnern durchgeführt. Daneben konnten auch Vor-Ort-Treffen mit Einweisungen und Workshops nicht wie geplant stattfinden.

Durch die Evaluation sind Fehler in der Auswertung und der grafischen Darstellung durch ifak und GFal erkannt und in der Folge die entsprechenden Softwarekomponenten verbessert worden. Für die im Projekt geplante prototypische Lösung wird dies als voller Erfolg gewertet, da das Demoszenario vollständig ausgeführt und den PA-Mitgliedern zum Ausprobieren vollständig übergeben werden konnte.

AP 12: Projektmanagement, Berichte und Publikationen

Das AP 12 lief projektbegleitend. Primär wurden hier die Planung, Durchführung und Nachbereitung von PA-Treffen durchgeführt. Dabei wurden die wirtschaftlichen Interessen von den Unternehmen des PA im Bereich V2X abgefragt und Feedback zu den Projektzielen und -fortschritten eingeholt. Dies geschah darüber hinaus in Telefonaten, über zusätzliche Web-Meetings und per E-Mail-Kommunikation.

Insgesamt wurden drei Konferenzen mit dem PA durchgeführt: Im September 2019, im Mai 2020 und im Februar 2021. Aufgrund von pandemiebedingten Reisebeschränkungen wurden die letzten beiden Treffen virtuell über Web-Konferenzen durchgeführt. Es war zur Zeit des Projektantrags ein viertes PA-Treffen geplant, unter anderem, um vor Ort beim ifak mit realer Hardware wie Roadside Units zu arbeiten und demonstrativ Daten zu erzeugen. Aufgrund der Reisebeschränkungen wurden die wesentlichen Ergebnisse in den übrigen PA-Treffen virtuell präsentiert und Feedback vorwiegend in nachgelagerten Einzelgesprächen eingeholt. Konnten PA-Mitglieder nicht bei den Web-Konferenzen anwesend sein, so wurden diese mithilfe des Protokolls und der Präsentationsunterlagen über das Treffen und den Projektfortschritt informiert.

Weitere Tätigkeiten in AP 12 waren

- die Unterzeichnung des Weiterleitungsvertrags zwischen ifak und GFal,
- die Einrichtung der gemeinsamen Dokumentbearbeitung über eine Nextcloud,
- die Durchführung eines internen Kick-Off-Treffens der beiden Forschungsstellen bei der GFal in Berlin, sowie
- die Erstellung eines Zwischenberichts (2019) und dieses Abschlussberichtes.

Nicht zuletzt wurden in AP 12 die laut Antrag geplanten Transfermaßnahmen der Partner koordiniert und nachverfolgt. Eine Übersicht findet sich in Abschnitt 6 dieses Berichts. Unter anderem wurde der Projektstart über die Website bei ifak und GFal verkündet und Zwischenergebnisse auf Workshops, Seminaren und Kolloquien bei den Forschungsstellen Interessenten mitgeteilt.

3. Angaben zu den aus der Zuwendung finanzierten Ausgaben

Das wissenschaftlich-technische Personal wurde im Projekt im folgenden Umfang eingesetzt:

FE1 (Gesellschaft zur Förderung angewandter Informatik e.V.): 27 PM

FE2 (Institut für Automation und Kommunikation e.V.): 20,97 PM

Es wurden keine Geräte beschafft und keine Leistungen Dritter herangezogen.

4. Erläuterung der Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die im Vorhaben durch ifak und GFal geleisteten Arbeiten waren für das Erreichen der Projektziele angemessen und notwendig. Die Arbeiten wurden im Wesentlichen gemäß dem Forschungsantrag durchgeführt.

Die aus der Corona-Pandemie resultierenden Arbeitsumstände im Jahr 2020 führten zu einer schwierigeren Einbeziehung der PA-Teilnehmer in die damals laufenden Arbeitspakete. Dies führte am Ende der Projektlaufzeit von DIAKO zu einem verstärkten Personalbedarf für interne Evaluierungen der entwickelten Systeme und Komponenten (AP 11) sowie bei der Umsetzung der Ergebnisse dieser Evaluierung (zugeordnet zu AP 9 und 10). Aus diesem Grund entstand bei der GFal ein geringer Mehrbedarf für das wissenschaftlich-technische Personal im Vergleich zum Projektantrag.

5. Darstellung des wissenschaftlich-technischen und wirtschaftlichen Nutzens

Mit der Entwicklung und dem Betrieb technischer Systeme ist die Prüfung der reibungslosen Funktion mittels Tests und Diagnosen im Rahmen der Qualitätssicherung untrennbar verbunden. Diese Prüfungen erfordern im Durchschnitt ca. 20 % des Entwicklungsaufwandes, bei sicherheitskritischen Projekten bis zu 50 % (Ammann, P. and Offutt, J.: Introduction to Software Testing, Cambridge University Press, Cambridge, 2008). Sie sind entscheidend für den Erfolg einer Technologie bzw. eines Produktes. Dies gilt ebenso für vernetzte (Rechner-) Systeme, wobei kooperative Verkehrssysteme nur ein Beispiel darstellen. Entwickler und Betreiber von Anwendungen und Software-Diensten benötigen letztendlich leistungsfähige Monitoring- und Diagnosesysteme, um frühzeitig Fehler und Inkonsistenzen identifizieren zu können.

Es existieren immer noch nur wenige Lösungen (sowohl Open-Source- als auch kommerzielle Werkzeuge) für die gezielte, regelbasierte Diagnose von Car2X-Systemen. Zudem werden bislang noch viele Nutzerbedürfnisse durch gar kein Produkt am Markt abgedeckt. In der Projektlaufzeit konnten folgende Innovationen konzipiert und im Software-Demonstrator umgesetzt werden, die einen hohen wirtschaftlichen Nutzen versprechen:

Gleichzeitiges Monitoring/Diagnose für räumlich getrennte Geräte. Dies ist über mehrere Erfassungsmodule an verschiedenen Orten möglich. Alle Module benötigen eine laufende Erfassungssoftware, die Daten z. B. einer zentralen Diagnose-Engine über MQTT und im JSON-Format zur Verfügung stellen. Diese Teillösung ist bereits alleine lauffähig, benötigt keine spezielle Parametrierung und könnte mit dem Rest des DIAKO-Systems zusammenarbeiten oder in eigene Lösungen externer Anwender integriert werden. Weiterhin ermöglicht das Dashboard die gleichzeitige Darstellung räumlich verteilter V2X-Daten, sei es in einer Stadt oder gar mehreren Städten. Die Tabellen mit Berichten werden durch alle eingehenden Daten bespielt. Eine wirtschaftliche Verwendung könnte in Verkehrsleitzentralen für Städte oder ganze Bundesländer gefunden werden.

Automatisierte Prüfung von Betriebsvorgaben über Regeln. Dies geschieht dauerhaft im Dashboard und über Logs, sodass sich die mühsame manuelle Auswertung, z. B. von Wireshark-Aufzeichnungen, erübrigt. Dies könnte eine wirtschaftliche Verwertung bei Entwicklern und Betreibern von entsprechenden V2X-Anwendungen und Systemen finden – oder auch bei Testdienstleistern, die die Regeln erstellen, starten und im Nachgang Berichte erstellen.

Ausrichtung auf Anwender mit verschiedenem Hintergrundwissen. Es wird zum einen mit dem erweiterten QuickSteps ein Modellierungswerkzeug für Diagnoseprozesse geboten, das Expertenwissen in den Ablaufbäumen hinterlegt und durch Endanwender mit weniger Wissen ausgeführt werden kann. Dies ermöglicht die erste einfache Hilfestellung bei Problemen, die bei Entwicklung, Test und Inbetriebnahme von V2X-Systemen auftreten könnten.

Bis 2030 könnte der weltweite Markt für V2X-Kommunikation auf Umsätze von etwa 15 Mrd. US\$ wachsen (Statista, Inc.: Projected size of the global market for V2X technologies in vehicles from 2020 to 2050, New York, 2017). Der Markt für Connected Cars (d. h. jegliche Art der drahtlosen Kommunikation) beträgt sogar 40 Mrd. € und könnte bis 2021 auf 122 Mrd. € anwachsen (Adam, Christian: Connected Car. Das vernetzte Auto, 2015).

Es werden jegliche Akteure aus den Bereichen „Intelligente Verkehrssysteme“ (seit längerer Zeit etabliert) sowie „Connected Car“ (seit den 2010er Jahren im Wachstum) an Diagnosen zur Qualitätssicherung interessiert sein. Sie profitieren von den in DIAKO entwickelten Methoden und den darauf aufbauenden Software-Modulen, da sie diese für die interne Qualitätssicherung oder die Bereitstellung von externen Dienstleistungen verwenden können. Da Deutschland durch Fahrzeughersteller (Daimler, VW, Audi ...) als auch Zulieferer (Infineon, Bosch, Continental ...) sehr stark im Automotive-Bereich vertreten ist, wird hier das Marktpotenzial auch als entsprechend hoch eingeschätzt.

Bietet ein Unternehmen ein V2X-fähiges Produkt (ein Gerät oder einen Dienst) an, so ist im Rahmen der Qualitätssicherung die korrekte Funktion durch Tests und Diagnosen abzusichern. Da vor allem KMU zu kurzen Entwicklungszyklen gezwungen sind, nehmen effiziente Diagnoseprozesse einen unternehmerisch hohen Stellenwert ein. Hier liefert DIAKO einen Beitrag, Fehler oder Inkonsistenzen aufzudecken und damit Kosten deutlich zu senken. Die entwickelten Methoden können im Ganzen oder in Teilen von Interessenten verwendet und ggf. weiterentwickelt werden. Darüber hinaus werden die Projektergebnisse die Grundlage für die Entwicklung eines marktreifen Diagnosewerkzeugs bilden. Je nach Anwendergruppe bietet ein solches Diagnosewerkzeug viele Möglichkeiten, die Wettbewerbsfähigkeit vor allem von KMUs zu steigern. Nachdem die Software beim letzten PA-Webmeeting vorgestellt wurde, hat bereits

ein KMU Interesse für die firmeninterne Weiterentwicklung bekundet. Der Kontakt wird nach Projektende durch die Forschungsstellen weitergeführt.

Nicht zuletzt leisten Intelligente Verkehrssysteme einen großen Beitrag, um die Verkehrssicherheit zu erhöhen und Unfallzahlen zu senken. Von V2X wird erwartet, durch die Unterstützung von Fahrerinformations- und -assistenzsystemen noch mehr Gefährdungen und Unfälle zu verhindern. Außerdem wird durch Dienste, wie der drahtlosen Bereitstellung der Ampelphasen, eine Verbesserung des Verkehrsflusses und letztlich eine Senkung des CO₂-Ausstoßes erwartet. Hierbei ist die Qualitätssicherung durch Diagnosen ein essenzieller Bestandteil. Sie unterstützt die nachhaltige Einführung und Akzeptanz von V2X bei den Anwendern.

6. Fortschreibung des Plans zum Ergebnistransfer in die Wirtschaft

Durchgeführte Transfermaßnahmen

Maßnahme	Ziel	Wann durchgeführt
News-Beitrag auf Website des ifak: https://www.ifak.eu/de/news/leistungsstarke-einfach-bedienende-werkzeuge-zur-diagnose-car2x-kommunikation	Interessenten das Projekt bekannt machen	April 2019
Interne Projektvorstellung beim ifak	Kollegen die allgemeinen Projektinhalte vorstellen und bereits frühzeitig eine Anschlussverwertung planen	Mai 2019
Ansprache weiterer PA-Mitglieder	Intensivierung des Ergebnistransfers während der Projektlaufzeit	Über die gesamte Laufzeit hinweg, vor allem in thematisch ähnlich gelagerten Forschungsprojekten
Weiteren Unternehmen wurde das Projekt vorgestellt. Über Projekttreffen themenverwandter Projekte.	Interessenten das Projekt bekannt machen; weitere Anwendungsfälle sammeln	SIRENE (29.08.2019) und BiDiMoVe (08. u. 09.10.2019)
Vorbereitungen zum Bericht über Projektbeginn im Institutsbericht 2019 des ifak	Interessenten das Projekt bekannt machen	Ende 2019 – wurde Anfang 2020 veröffentlicht

Vorbereitungen zum Bericht über Projektbeginn im Jahresbericht 2019 der GFal	Interessenten das Projekt bekannt machen	Ende 2019 – wurde Anfang 2020 veröffentlicht
Aufbau eines ersten einfachen Test-Systems als Demonstrator, potenziell durch DIAKO-Arbeiten erweiterbar	Präsentation des Gesamtsystems im Technikum des ifak Magdeburg sowie auf Tagungen und Messen	Mai 2019, erste Präsentation bei Langer Nacht der Wissenschaft am 25.05.2019
2. PA-Treffen über Web-Konferenz durchgeführt	PA-Mitgliedern die Fortschritte vorstellen, vor allem Anwendungsfälle und Systemarchitektur. Auch konnten neue Mitglieder gewonnen werden.	Mai 2020
Interne Vorstellung bei GFal im Rahmen eines Kolloquiums	Projekthinhalte vorstellen, Anschlussverwertung planen, Feedback zur prototypischen Umsetzung erhalten	August 2020
3. PA-Treffen über Web-Konferenz durchgeführt	PA-Mitgliedern die Fortschritte vorstellen, vor allem den funktionierenden Software-Prototypen samt Demo.	Februar 2021
Vorstellung und Tests sowie Optimierung der Methoden und des Tools in Unternehmen (des PA)	Überprüfung der Funktionalität, Einbringen von Feedback, Verbesserungen	Seit der Bereitstellung über einen Download Anfang März 2021

Geplante Transfermaßnahmen nach Abschluss des Vorhabens

Maßnahme	Ziel	Zeithorizont
Veröffentlichung des Abschlussberichtes	Breite Streuung der Forschungsergebnisse	ca. 6 Monate nach Projektende

<p>Gezielte Ansprache potenziell interessierter Unternehmen. Z. B. im Rahmen von V2X-Projekten wie BiDiMoVe oder LOGIN Hannover. Auch ist die Bekanntmachung in mFUND-Workshops des BMVI denkbar.</p>	<p>Verbreitung der Methodik auch in Unternehmen außerhalb des Projekts</p>	<p>Ab Mitte 2021</p>
<p>Angebot von Schulungen und Workshops für interessierte Unternehmen. Mitglieder des Projektbegleitenden Ausschusses, gezieltes Anschreiben von Unternehmen.</p>	<p>Sicherstellung der korrekten Anwendung der Methodik in Unternehmen</p>	<p>Ab 2022</p>
<p>Beiträge in internationalen Fachzeitschriften sowie Newslettern von Verbänden. Z. B. Magazin „Internationales Verkehrswesen“, „Softwaretechnik-Trends“ (STT) oder Car2Car CC-Newsletter.</p>	<p>Informationsverbreitung für die praxisnahe Wissenschaft</p>	<p>Ab Mitte 2021</p>
<p>Erstellung von Newslettern und Verbreitung über eigene Informationskanäle (Anschreiben der Unternehmen mit Newslettern)</p>	<p>Informationsverbreitung an weitere Unternehmen verschiedener Branchen</p>	<p>Ab Mitte 2021</p>
<p>Vorträge auf Tagungen, Kongressen und Kolloquien. Z. B. Automotive meets Electronics (Dortmund), ConCarExpo, Car2Car Forum, geeigneter VDE-Kongress.</p>	<p>Informationsverbreitung an weitere Unternehmen</p>	<p>Ab 2022</p>
<p>Vorstellung ausgesuchter Projektergebnisse (entwickelte Methoden) bei geeigneten Standardisierungs-Gremien. Z. B. über Arbeitsgruppen des Car2Car CC, der ETSI, des VDI oder des ISTQB.</p>	<p>Diskussion von und evtl. Einbringen in Standards, zum Beispiel zum systematischen Testen und der Diagnose</p>	<p>Kontinuierlich</p>
<p>Weiterentwicklung (über die Vorwettbewerbphase hinaus) durch PA-Mitglieder und externe Unternehmen bei diskriminierungsfreier Bereitstellung der entwickelten prototypischen Systemkomponenten</p>	<p>Marktreifes Softwaresystem</p>	<p>Ab Mai 2021 (direkt nach Projektende)</p>
<p>Nutzung der Projektergebnisse bei der Beratung von Unternehmen, ifak und GFal</p>	<p>Transfer der Ergebnisse in KMU</p>	<p>Ab 6 Monate nach Projektende und kleineren Weiterentwicklungen der Software</p>

Einschätzung zur Realisierbarkeit des Transferkonzepts

Die Transfermaßnahmen nach Abschluss des Forschungsprojekts wurden und werden durch die Covid19-Pandemie behindert. Eine Durchführung gemäß des Transferkonzepts muss daher als ambitioniert bezeichnet werden.

Vor-Ort-Termine wie Schulungen, Konferenzen etc. konnten auch im Jahr 2021 nicht stattfinden. Entsprechend geplante Maßnahmen wurden daher auf das Jahr 2022 verschoben. Das Nicht-Stattfinden von Messen und anderen Veranstaltungen erschwert zudem die gezielte Ansprache von KMU über den PA-Teilnehmerkreis hinaus. Insbesondere durch die Forschungsstelle ifak können aber durch weitere Forschungsaktivitäten im Bereich entsprechende Kontakte hergestellt und gepflegt werden.

Ein Teil der nach Projektende geplanten Transfermaßnahmen konnte jedoch bereits gestartet werden. So wurden etwa entsprechende Veröffentlichungen in eigenen Informationskanälen vorbereitet. Weiterhin konnten die Projektergebnisse für die Beratung von Unternehmen in kleinem Rahmen eingesetzt werden. Die Weiterentwicklung der Projektergebnisse durch externe Unternehmen wird ebenfalls durch die Covid19-Pandemie (z.B. wirtschaftliche Situation) behindert. Die entwickelte Demonstrator-Lösung wurde allerdings den PA-Teilnehmern zur Verfügung gestellt und Hilfestellungen beim Transfer werden angeboten.