

**Projekt: IntelliMar – Intelligente Assistenz- und Analysesysteme zur
Früherkennung und Bewältigung von maritimen
Gefahrensituationen**

Teilvorhaben: **Services**

Titel: **Schlussbericht**

Laufzeit des Vorhabens: 01.10.2018 - 31.08.2021

FKZ: 03SX469B

Autor: Norbert Link

Prüfer: Felix Steybe

Gefördert durch:



Bundesministerium
für Wirtschaft
und Energie

aufgrund eines Beschlusses
des Deutschen Bundestages

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor

Airbus Defence and Space GmbH

Wörthstraße 85, 89077 Ulm

Inhalt

1.	Einleitung	8
1.1.	Einführung.....	8
1.2.	Definitionen	8
1.3.	Abkürzungen	8
1.4.	Anwendbare Dokumente	10
1.5.	Referenzdokumente	10
1.6.	Autoren	10
2.	Kurzdarstellung	11
2.1.	Aufgabenstellung	11
2.2.	Voraussetzungen	11
2.3.	Planung und Ablauf des Vorhabens	12
2.4.	Ausgehender wissenschaftlicher Stand	12
2.5.	Zusammenarbeit mit anderen Stellen	13
3.	Ausführliche Darstellung der Ergebnisse	15
3.1.	AP 1200 - Data Lake	15
3.1.1.	Datenarten	15
3.1.2.	Datenquellen	15
3.1.3.	Data Lake.....	19
3.2.	AP 1500 - Airbus Brain Development	30
3.2.1.	Architektur der Brains	30
3.2.2.	Regelbasierte Detektoren.....	51
3.2.3.	Auswahlprozess der Anomalien für IntelliMar Services	76
3.2.4.	Regelbasierte Anomalien	77
3.2.5.	Benchmarking-Unterstützung	94
3.2.6.	Benutzerreferenz.....	95
3.2.7.	Entwicklung zu ML-basierten Detektoren	106
3.3.	AP 2100 - Service Concepts	127
3.3.1.	AP 2110 - User Requirements (International)	127
3.3.2.	AP 2120 - Service Requirements (International).....	133
3.3.3.	AP 2130 - Optimized NRT Processing	141
3.4.	Validierungskampagnen	157
3.4.1.	Ergebnisse der Validierungskampagne I	157
3.4.2.	Ergebnisse der Validierungskampagne II	158

3.4.3. Ergebnisse der Valdierungskampagne III	159
3.5. Wichtige Positionen des zahlenmäßigen Nachweises	159
3.6. Notwendigkeit und Angemessenheit der geleisteten Arbeit	159
3.7. Verwertbarkeit der Ergebnisse	160
3.8. Fortschritt auf dem Gebiet des Vorhabens bei anderen Stellen	160
3.9. Veröffentlichungen	160

Abbildungsverzeichnis

Abbildung 3-1: Verteilung und Anzahl verwertbarer AIS-Daten des WSV-Datensatzes zum 28.05.2012	16
Abbildung 3-2: Verteilung und Anzahl verwertbarer AIS-Daten des Providers AISHub am 28.01.2021	18
Abbildung 3-3: Schematische Darstellung eines Data Lakes im Big-Data-Umfeld	20
Abbildung 3-4: Gesamtarchitektur IntelliMar-Demonstrator	20
Abbildung 3-5: Schematische Darstellung des Data Lakes bei Intellimar	21
Abbildung 3-6: Architektur des IntelliMar-Data-Lakes	23
Abbildung 3-7: Beispiel-Auszug aus automatisch angelegter HDFS-Struktur	29
Abbildung 3-8: Verarbeitungskette aus Verarbeitungsknoten	38
Abbildung 3-9: Innere Architektur eines Verarbeitungsknotens	39
Abbildung 3-10: Verarbeitungskette mit Steuer- und Log-Integration	40
Abbildung 3-11: Gesamtarchitektur IntelliMar Services	42
Abbildung 3-12: Kommunikation zwischen Streaming Nodes und Brain Manager	43
Abbildung 3-13: Interaktion zwischen Brain Manager und Brain (li) und Zustandsübergänge der Brains (re)	44
Abbildung 3-14: Der Partitioner verteilt die Datenlast auf Partitionen	46
Abbildung 3-15: Partitionierung nach MMSIs	47
Abbildung 3-16: Partitionierung nach MMSI verhindert Erkennung einer Schiffsbegegnung	48
Abbildung 3-17: Partitionierung je Gebiet: Schiffsbegegnung wird erkannt.	49
Abbildung 3-18: Cluster-Teilung bei Partitionierung nach Gebieten	49
Abbildung 3-19: Anomalie-Scan des WSV-Datensatzes vom 28. Mai 2012	50
Abbildung 3-20: Schematische Darstellung des Datenflusses	51
Abbildung 3-21: Schematische Darstellung der Funktionsweise des Aggregators	64
Abbildung 3-22: Vereinfachte Darstellung des Prinzips des Trackings	65
Abbildung 3-23: Unterschied zwischen gegebenem und durch den Algorithmus errechneten Kurs	65
Abbildung 3-24: Brain Manager HMI (Hauptseite)	74
Abbildung 3-25: Brain Manager HMI (Parameter)	75
Abbildung 3-26: Turn eines Schiffes innerhalb eines Zeitfensters	78
Abbildung 3-27: Veranschaulichung von mehreren gefundenen Turn-Anomalien	79
Abbildung 3-28: Ansicht des Airbus Defence and Space Ulm internen HMI: Fahrt der ZUSE eines 30° Turns nördlich von Bremerhaven	80
Abbildung 3-29: Übersicht der Anomaliekette „Hard-Vessel-Turn“	81

Abbildung 3-30: AIS-Signalverlust eines Schiffes bei gleichbleibendem Kurs.....	83
Abbildung 3-31: AIS-Signalverlust eines Schiffes mit Kursveränderung	84
Abbildung 3-32: Ansicht des Airbus Defence and Space Ulm internen HMI.....	85
Abbildung 3-33: Übersicht der Anomaliekette „AIS-Signal-Loss“	86
Abbildung 3-34: Ansicht des Airbus Defence and Space Ulm internen HMI.....	88
Abbildung 3-35: Übersicht der Anomaliekette „Vessel-Close-Approach“.....	89
Abbildung 3-36: Ansicht des Airbus Defence and Space Ulm internen HMI.....	90
Abbildung 3-37: Übersicht der Anomaliekette „Small-Fast-Boat“	91
Abbildung 3-38: Ansicht des Airbus Defence and Space Ulm internen HMI.....	93
Abbildung 3-39: Übersicht der Anomaliekette „Double-Imo-Callsign“.....	94
Abbildung 3-40: Beispiel eines BNs mit den Variablen A,B,C,D und E und den zugehörigen CPDs.....	108
Abbildung 3-41: Beispiel eines BNs mit den Variablen A,B,C,D und E und den assoziierten CPTs.	109
Abbildung 3-42: Beispiel eines DBNs mit der A-priori-Verteilung B0 links und dem 2TBN rechts (basierend auf (Fischer, 2016; Sucar, 2015)).	110
Abbildung 3-43: Beispiel eines DBNs mit 3 Knoten über 4 Zeitschritte (vgl. (Sucar, 2015)).	111
Abbildung 3-44: Eine einfach modellierte Situation eines Schmuggelboots.	115
Abbildung 3-45: Das Situationsnetzwerk für die verzögerte Drogenübergabe.	116
Abbildung 3-46: Das 2TBN des Situationsnetzwerks aus Abbildung 3-44.	117
Abbildung 3-47: Anzahl der CPT-Einträge und der zu bestimmenden Parameter pro Situation	117
Abbildung 3-48: Architektur der prototypischen Realisierung.....	118
Abbildung 3-49: Robustheit mit drei verschiedenen Rauschwerten q.	120
Abbildung 3-50: Sprungverhalten mit 5 aufeinander folgende gleiche Evidenzen.	120
Abbildung 3-51: Das maritime Lagebild zeigt den Schiffsverkehr des simulierten Szenarios.	122
Abbildung 3-52: Die Schiffstypen der einzelnen Schiffe im beispielhaften simulierten Szenario.	122
Abbildung 3-53: Ergebnisse des Schiffes mit der MMSI-Nummer 4012.....	123
Abbildung 3-54: Ergebnisse des Schiffes mit der MMSI-Nummer 4210.....	123
Abbildung 3-55: Ergebnisse des Schiffes mit der MMSI-Nummer 1234.....	124
Abbildung 3-56: Ergebnisse des Schiffes mit der MMSI-Nummer 4500.....	124
Abbildung 3-57: Ergebnisse des Schiffes mit der MMSI-Nummer 6300.....	125
Abbildung 3-58: Erfassung der Wichtigkeit von Schiffsparametern (Beispiel)	128
Abbildung 3-59: Mitglieder des ECGFF und NACGF	129
Abbildung 3-60: Prototypischer Wizard zur Unterstützung bei der Sensor Auswahl und – Konfiguration.....	136
Abbildung 3-61: Bewertung der Detektionsgüte von Objekten auf offener See in TerraSAR-X Bilddaten abhängig von Sensorauflösung und Windstärke bzw. Wellenhöhe.	138
Abbildung 3-62: Prototypisches Design einer Auflistung aller für ein detektiertes Objekt bereitgestellter Informationen und deren Qualitätsindikatoren. Ein klappbares Panel könnte die Informationen aus einer Sekundärquelle (AIS / RF) zeigen.....	139
Abbildung 3-63: Vergleich automatischer Schiffsdetektion (CFAR) zwischen PAZ und TerraSAR-X.....	144

Abbildung 3-64: Sentinel-1 CFAR Ergebnisse des SUMO Algorithmus über Singapur (S1A_IW_GRDH_1SDV_20200108T224756_20200108T224821_030715_03856E_1FB2.SAFE)	146
Abbildung 3-65: Sentinel-1 Beispiel für manuelle Schiffserkennung	147
Abbildung 3-66: Toolkit Layout auf zwei Bildschirmen (Detailansicht in Abbildung 3-67 und Abbildung 3-68)	153
Abbildung 3-67: MAP VIEWER Entwurf (vgl. Abbildung 3-66 linker Bildschirm).....	154
Abbildung 3-68: DATA VIEWER Entwurf (vgl. Abbildung 3-66 rechter Bildschirm)	156

Tabellenverzeichnis

Tabelle 1-1: Definitionen/ Glossar.....	8
Tabelle 1-2: Abkürzungen	9
Tabelle 1-3: Autoren.....	10
Tabelle 3-1: Befehle zur Steuerung der Streaming Nodes.....	44
Tabelle 3-2: Brain Manager Benutzerkommandos.....	45
Tabelle 3-3: Mapping der externen Informationsfelder ins interne AVRO-Schema AisMessage.....	55
Tabelle 3-4: Verwendete Aufzählungstypen (Enums)	57
Tabelle 3-5: Mehrfach genutzte Datentypen	58
Tabelle 3-6: Internes AVRO-Schema AisAnomalyReport	60
Tabelle 3-7: JSON-Schema des JSON-Report-Formatters.....	66
Tabelle 3-8: Parameter des Speed-Filters und deren Bedeutung	69
Tabelle 3-9: Parameter des NavState-Filters und deren Bedeutung	69
Tabelle 3-10: Parameter des ShipType-Filters und deren Bedeutung.....	70
Tabelle 3-11: Parameter des Region-Filters und deren Bedeutung	70
Tabelle 3-12: Parameter des Polygon-Filters und deren Bedeutung.....	71
Tabelle 3-13: Liste der gewählten Anomalien	77
Tabelle 3-14: Parameter des Hard-Vessel-Turns und deren Bedeutung.....	81
Tabelle 3-15: Legende für Erkennungsketten	82
Tabelle 3-16: Parameter des AIS-Signal-Loss und deren Bedeutung	86
Tabelle 3-17: Parameter des Vessel-Close-Approach und deren Bedeutung	89
Tabelle 3-18: Parameter des Small-Fast-Boat und deren Bedeutung	91
Tabelle 3-19: Liste der allgemeinen Optionen und Parameter der Streaming-Nodes.....	96
Tabelle 3-20: Liste der Optionen für imar-ais-mmsi-partitioner	97
Tabelle 3-21: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-ais-signal-loss	98
Tabelle 3-22: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-hard-vessel-turn.....	100
Tabelle 3-23: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-navstate-filter	100
Tabelle 3-24: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-polygon-filter	101
Tabelle 3-25: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-region-filter.....	102

Tabelle 3-26: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-shiptype-filter	103
Tabelle 3-27: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-small-fast-boat	103
Tabelle 3-28: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-speed-filter	104
Tabelle 3-29: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-vessel-close-approach	104
Tabelle 3-30: Liste aller Optionen für BrainManagerCLI	105
Tabelle 3-31: Sensor Parameter und maritime Empfehlungen.....	135
Tabelle 3-32: Einfluss von verschiedenen Faktoren auf die Detektierbarkeit von Objekten in maritimen SAR Daten. Pfeile geben eine Indikation des Einflusses an: positiv (↑), negativ (↓) oder neutral (→).....	137
Tabelle 3-33: Daten zur Masterarbeit, welche begleitend zur IntelliMar Studie entstand.....	160

Literaturverzeichnis

- Airbus Defence and Space GmbH. (2021). *Schlussbericht zum PtJ-Vorhaben FKZ 03SX469A*. Bremen, Deutschland.
- AIS data sharing and vessel tracking by AISHub*. (23. 07 2021). Von AISHub Data Exchange: <https://www.aishub.net/> abgerufen
- Apache Hadoop*. (23. 07 2021). Von Apache Hadoop: <http://hadoop.apache.org/> abgerufen
- Apache Software Foundation. (23. 07 2021). *Apache Cassandra*. Von Apache Cassandra: <https://cassandra.apache.org/> abgerufen
- Apache Software Foundation. (23. 07 2021). *Apache Kafka*. Von Apache Kafka: <https://kafka.apache.org/> abgerufen
- Apache Software Foundation. (23. 07 2021). *Parquet*. Von Parquet: <https://parquet.apache.org/> abgerufen
- Cloudera, Inc. (23. 07 2021). *Cloudera*. Von Cloudera: <https://de.cloudera.com/> abgerufen
- Confluent, Inc. (23. 07 2021). *Confluent*. Von Confluent: <https://www.confluent.io/> abgerufen
- Confluent, Inc. (23. 07 2021). *Introduction to Kafka*. Abgerufen am 23. 07 2021 von Introduction to Kafka: <https://docs.confluent.io/platform/current/kafka/introduction.html>
- Docker, Inc. (2021). *docker.com*. Abgerufen am 19. 08 2021 von docker.com: <https://www.docker.com/>
- Docker, Inc. (2021). *Docker Compose*. Abgerufen am 19. 08 2021 von Docker Compose: <https://docs.docker.com/compose/compose-file/compose-file-v3/>
- Fischer, Y. (2016). *Wissensbasierte probabilistische Modellierung für die Situationsanalyse am Beispiel der maritimen Überwachung*. Karlsruhe: KIT Scientific Publishing.

- Fischer, Y., Reisch, A., & Beyerer, J. (2014). Modeling and Recognizing Situations of Interest in Surveillance Applications. (S. S.186–192). IEEE.
- Fischer, Y., Reisch, A., & Beyerer, J. (2014). Modeling and Recognizing Situations of Interest in Surveillance Applications. (S. S.186–192). IEEE.
- IOSB, F. (9. Mai 2020). *Hybrid AI Combining Machine Learning and Dynamic Bayesian Networks for Behavior Recognition*. Von <https://www.iosb.fraunhofer.de/servlet/is/101100/> abgerufen
- Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.
- Mark Grover, T. M. (2015). *Hadoop Application Architectures: Designing Real-World Big Data Applications*. USA: O'Reilly Media.
- MongoDB, Inc. (23. 07 2021). *MongoDB*. Von MongoDB: <https://www.mongodb.com/de-de> abgerufen
- Murphy, K. P. (2002). *Dynamic Bayesian Networks: Representation, Inference and Learning*. Berkeley: University of California.
- NMEA. (2021). *NMEA*. Abgerufen am 23. 08 2021 von NMEA: <http://www.nmea.de/>
- Rammelt, P. (16. Dezember 2019). *Bayes'sche Netze*. Von <https://patricks-seite.de/probnet/index.html> abgerufen
- Salling, T. B. (2021). *GitHub*. Abgerufen am 19. 08 2021 von GitHub: <https://github.com/tbsalling/aismessages>
- Sucar, L. E. (2015). *Probabilistic Graphical Models: Principles and Applications*. Springer.
- U.S. Coast Guard Navigation Center. (17. 04 2019). *AIS Messages*. Abgerufen am 23. 08 2021 von AIS Messages: <https://www.navcen.uscg.gov/?pageName=AIMessages>
- Yuanqiao Wen, u. a. (2019). *Semantic modelling of ship behavior in harbor based on ontology and dynamic bayesian network*. Bd. 8. 3. Multidisciplinary Digital Publishing Institute.

1. Einleitung

1.1. Einführung

Der vorliegende Schlussbericht beschreibt die inhaltlichen Arbeiten und Ergebnisse des Teilprojektes IntelliMar Services, das im Rahmen des Verbundprojektes IntelliMar („Intelligente Assistenz- und Analysesysteme zur Früherkennung und Bewältigung von maritimen Gefahrensituationen“) bearbeitet wurde. Alle Arbeiten zu IntelliMar Services wurden von Airbus Defence and Space GmbH (Airbus DS) in Ulm und Friedrichshafen im vorgegebenen Zeit- und Kostenrahmen des Projektes durchgeführt.

1.2. Definitionen

Begriff	Definition
AIS	Funksystem zum Austausch von Navigations- u. Schiffsdaten
DSG	Data Service Grid, Infrastruktur-Komponente von IntelliMar
GeoJSON	Offenes Datenformat zur Repräsentation und zum Austausch geografischer Daten (siehe auch JSON)
HashMap	Eine Zuordnung aus Schlüssel-Wert-Paaren. Jeder Wert hat einen eigenen Schlüssel und kann über diesen aufgefunden werden.
IMO	Die Internationale Seeschiffahrtsorganisation (International Maritime Organization, IMO) ist eine Sonderorganisation der Vereinten Nationen mit Sitz in London.
JSON	Kompaktes Datenformat in lesbarer Textform zum Datenaustausch
NMEA	Kommunikationsprotokoll der NMEA zum Austausch von Daten zwischen Geräten der Marineelektronik
UTC	Coordinated Universal Time (UTC, dt.: koordinierte Weltzeit) ist die heute gültige Weltzeit
WGS84	Das World Geodetic System 1984 (WGS 84) ist ein geodätisches Referenzsystem als einheitliche Grundlage für Positionsangaben auf der Erde und im erdnahen Weltraum.
YAML	YAML (Yet Another Multicolumn Layout) ist eine vereinfachte Auszeichnungssprache (engl. markup language) zur Datenserialisierung.

Tabelle 1-1: Definitionen/ Glossar

1.3. Abkürzungen

Abkürzung	Definition
ABI	Activity Based Intelligence
Airbus DS	Airbus Defence and Space GmbH
AIS	Automatic Identification System
AP	Arbeitspaket
API	Application Programming Interface (Programmierschnittstelle)
BN	Bayes'sches Netz
CFAR	Constant False Alarm Rate

CLI	Command Line Interface
CONOPS	Concept of Operations
CPD	Conditional Probability Distribution
CPT	Conditional Probability Table
DAG	Direkter Azyklischer Graph
DBN	Dynamisches Bayes'sches Netz
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DRS	Direct Receiving Station
DSG	Data Service Grid
DSGVO	Datenschutzgrundverordnung
ECEF	Earth Centered Earth Fixed
ECGFF	European Coast Guard Functions Forum
EEC	Enhanced Ellipsoid Corrected
EMSA	European Maritime Safety Agency
EU	Europäische Union (European Union)
FN	Friedrichshafen
GEC	Geocoded Ellipsoid Corrected
GIS	Geoinformationssystem
HAP	Hauptarbeitspaket
HDFS	Hadoop Distributed File System
HDP	Hortonworks Data Platform
HMI	Human Machine Interface
IMO	International Maritime Organization
IT	Informationstechnologie
JRC	Joint Research Center
JSON	Java Script Object Notation
MGD	Multi Look Ground Range Detected
MGD_RE	Multilook Ground Range Detected; radiometrically enhanced
MMSI	Maritime Mobile Service Identity
NACGF	North Atlantic Coast Guard Forum
NMEA	National Marine Electronics Association
NoSQL	Not-only SQL
NRT	Near-real-time
SAR	Synthetic Aperture Radar
SQL	Structured Query Language
TCP	Transmission Control Protocol
URL	Uniform Resource Locator
UTC	Coordinated Universal Time
WGS84	World Geodetic System 1984
WSV	Wasserstraßen- und Schifffahrtsverwaltung des Bundes
YAML	Yet Another Multicolumn Layout

Tabelle 1-2: Abkürzungen

1.4. Anwendbare Dokumente

N/A

1.5. Referenzdokumente

- [RD-01] Vorhabenbeschreibung und Arbeitsplan zum Verbundprojekt: „Intelligente Assistenz- und Analysesysteme zur Früherkennung und Bewältigung von maritimen Gefahrensituationen – Services“
Aktenzeichen PtJ: 0303X533, Referenz-Nr.: 015703-000-0000, V1.0, 20.06.2018
- [RD-02] Leitantrag zum Verbundprojekt: „Intelligente Assistenz- und Analysesysteme zur Früherkennung und Bewältigung von maritimen Gefahrensituationen“
Aktenzeichen PtJ: 0303X533, Airbus DS Referenz-Nr.: 17.5.0158.0, V3.0, 21.06.2018

1.6. Autoren

Autor	Abteilung	Kapitel
Christmann, Roland	TCIGS	3.3
Fröhler, Leonhard	TEMDA2	3.2.2, 3.2.4, 3.2.6
Grupp, Cornelius	TCIGS	3.3
Haaga, Steffen	TEMDC3	3.2.7
Hihler, Ralf	TEMDC3	1.1, 2.1 – 2.5
Jochum, Markus	TCIGS	3.3
Link, Norbert	TEMDA2	2.2, 3.1, 3.2.1, 3.2.3, 3.2.5, 3.4, 3.8, 3.9
Schmid, Elke	TEMDA3	3.2.1.5, 3.2.2.8, 3.2.2.9, 3.2.6.18
Schmid, Martin	TEMDA2	1.2 – 1.6
Steybe, Felix	TCIDS1	3.5 – 3.7
von Hassler zu Roseneckh-Köhler, Bastian	TEMDA3	3.2.2, 3.2.4

Tabelle 1-3: Autoren

2. Kurzdarstellung

2.1. Aufgabenstellung

Wesentliche Ziele des Verbundprojektes IntelliMar bestehen darin, durch verbesserten Austausch maritimer Informationen, durch automatisierte Datenanalyse und Anomalie-Detektion sowie durch neue Ansätze der Visualisierung von Anomalien die maritime Lagedarstellung und –beurteilung zu verbessern und damit Möglichkeiten zur Erhöhung der Sicherheit in den deutschen Seegebieten aufzuzeigen.

Die in der Vorhabenbeschreibung [RD-01], Kapitel 2, beschriebenen Aufgaben des Förderprojektes IntelliMar Services sollen einen wichtigen Beitrag zur Erreichung der Verbundziele leisten.

Dabei geht es um die Anbindung von AIS-Datenquellen an einen zentralisierten „Data Lake“ als Teil der maritimen Validierungsumgebung. Unter Nutzung von regelbasierten sowie von Machine Learning Methoden sollen Software-Komponenten („Brains“) entwickelt werden, die auf Basis des Echtzeit-Datenstromes oder der archivierten Daten Anomalien erkennen und melden. Die konkret zu untersuchenden Anomalien werden gemeinsam mit den assoziierten Projektpartnern festgelegt.

Im Rahmen von Validierungskampagnen werden der Bearbeitungsfortschritt, die Integrationsfähigkeit mit den Verbundpartnern sowie die Qualität der Anomalie-Erkennung aufgezeigt und bewertet.

Zusätzlich werden als Beitrag zur Weiterentwicklung operationeller Konzepte Anforderungen an ein System zur intelligenten Datenanalyse durch eine Befragung von internationalen Nutzern erfasst und ausgewertet. Auf Basis dieser Nutzbefragung sollen mögliche Dienstleistungskonzepte beschrieben werden.

2.2. Voraussetzungen

Die Voraussetzungen für die oben beschriebene Aufgabenstellung sind:

1. Die Unterstützung durch die assoziierten Projektpartner hinsichtlich Priorisierung und Auswahl der zu untersuchenden Anomalien und Festlegung der wesentlichen Parameter.
2. Die Verfügbarkeit von AIS Schiffsmeldungen als Datenstrom und in Form von länger dauernden Aufzeichnungen in einem festzulegenden Seegebiet.
3. Die Bereitstellung eines zentralen Data Lakes (Hardware und Basis-Software) durch den Verbundpartner Airbus DS Bremen.
4. Die Bereitstellung von Datenübertragungsfunktionen in einem verteilten Verbund-Netzwerk durch den Verbundpartner Airbus DS Bremen.

5. Die Bereitstellung von Simulationsdaten und aufgezeichneten Manöverdaten eines Forschungsschiffes durch den Verbundpartner Offis in Oldenburg.
6. Der Zugang zu internationalen Nutzern und deren Mitwirkung bei die Durchführung der Nutzerbefragung.

2.3. Planung und Ablauf des Vorhabens

Die Arbeiten im Rahmen von IntelliMar Services sind auf die folgenden Arbeitspakete des Verbundprojektes IntelliMar aufgeteilt (s. [RD-01], Tabelle 1):

HAP 1000 Intelligence Benchmark

AP 1200 Data Lake

AP 1210 Terrestrial Data Sources

AP 1220 Data Fusion Service

AP 1500 Airbus Brain Development

AP 1510 Brain 1 "Rule Based"

AP 1520 Brain 2 "Open Source ML Tools"

AP 1530 Benchmark Support

HAP 2000 Intelligence CONOPS

AP 2100 Service Concepts

AP 2110 User Requirements (International)

AP 2120 Service Requirements (International)

AP 2130 Optimized NRT Processing

Die Arbeiten zu HAP 1000 wurden von Airbus DS am Standort Ulm, die von HAP 2000 in am Standort Friedrichshafen durchgeführt. Die geplanten Meilensteine M1: Validierungskampagne 1, M2: Validierungskampagne 2 und M3: Validierungskampagne 3 und Abschlusspräsentation s. [RD-01], Tabelle 1) wurden eingehalten.

2.4. Ausgehender wissenschaftlicher Stand

Airbus DS wirkte bereits in mehreren Forschungsvorhaben bzw. Projekten im Kontext der maritimen Sicherheit und Überwachung mit.

Daneben hat Airbus DS für Datenprozessierung und Systementwicklung langjährige Sachkenntnis und ausgeprägtes Know-how aufgebaut. Dabei wurden komplexe Software-Systeme mit echtzeitnaher Verarbeitung von Sensordaten zur Erzeugung eines konsistenten taktischen Lagebildes entwickelt. Kernfähigkeiten dabei sind die Multi-Sensor Datenfusion,

die Identifizierung und Klassifizierung von Objekten, ein umfassendes Track Management sowie ein Austausch der Lageinformation mit anderen Einheiten. Dies beinhaltete auch erste Erfahrungen mit der Anbindung von AIS-Sensoren an die bestehenden echtzeitnahen Systeme.

Methoden und Software-Anteile aus der Multi-Sensor Datenfusion konnten bei IntelliMar zur Glättung der AIS-Positionen der unterschiedlichen Schiffe genutzt werden.

Erste Erfahrungen mit Aufbau und Konfiguration eines Big Data Clusters sowie mit der Anwendung eines Big Data Frameworks zur Generierung von Heat Maps zu Positionen und Bewegungen von Objekten lagen zu Beginn des Vorhabens IntelliMar bei Airbus DS bereits vor. Diese bildeten die Grundlage für das Data Lake Konzept, für die Verarbeitung von AIS Datenströmen sowie die Implementierung von Brains zur Anomaliedetektion und deren Einbindung in ein vernetztes Gesamtsystem.

2.5. Zusammenarbeit mit anderen Stellen

Die Arbeitsteilung mit den Projektpartnern erfolgte im Rahmen eines Forschungsverbundes, wie im Leitantrag zum Forschungsprojekt (siehe [RD-02]) IntelliMar festgeschrieben. Hierzu wurde seitens Airbus DS eine entsprechende Kooperationsvereinbarung mit den beteiligten Projektpartnern geschlossen.

Die Koordination der gemeinsamen Tätigkeiten im Forschungsverbund erfolgte durch Airbus DS in Bremen, Abstimmungen hinsichtlich Planung und Technik erfolgten im Rahmen von regelmäßigen Verbundtreffen bzw. Telefon-, Video- oder Web-Konferenzen. Detailabstimmungen zwischen Verbundpartnern waren auch bilateral möglich.

Als assoziierte Partner fungierten die Bundespolizei See sowie die Wasserstraßen- und Schifffahrtsverwaltung des Bundes. Unter Federführung von Airbus DS Bremen war Airbus DS Ulm/FN an mehreren Abstimmungsgesprächen mit den assoziierten Partnern beteiligt, unter anderem auch an einer ausführlichen Besprechung im Maritimen Sicherheitszentrum in Cuxhaven. Dort konnten operationelle Anforderungen und technische Details diskutiert werden.

Im Rahmen der Spezifikation und Priorisierung von Anomalien nahm Airbus DS an Abstimmungsgesprächen mit dem Bundesamt für Seeschifffahrt und Hydrographie teil, um Möglichkeiten der Driftberechnung bei einzelnen Anomalien zu diskutieren.

Für die Nutzerbefragung wurde das European Coast Guard Functions Forum (ECGFF) sowie das North Atlantic Coast Guard Forum (NACGF) einbezogen.

Das European Coast Guard Functions Forum ist ein unverbindliches, freiwilliges, unabhängiges und nicht politisches Forum. Zu den Mitgliedern zählen Küstenwachbehörden von 25 EU-Mitgliedstaaten und assoziierten Schengen-Ländern sowie die Europäische Kommission und ihre Institutionen und Agenturen mit entsprechenden Kompetenzen in Küstenwachfunktionen. ECGFF wurde 2009 ins Leben gerufen und wird von der EU co-finanziert.

Das North Atlantic Coast Guard Forum (NACGF) erweitert die Internationale Zusammenarbeit in Nordamerika und fördert die Zusammenarbeit und den Informationsaustausch zwischen Behörden mit Küstenwacheinsätzen.

3. Ausführliche Darstellung der Ergebnisse

3.1. AP 1200 - Data Lake

Im Rahmen des Teilvorhabens IntelliMar Services wurden aus der Arbeitspaketgruppe AP 1200 „Data Lake“ die Arbeitspakete AP 1210 (Terrestrial Data Sources) sowie AP 1220 (Data Fusion Services) bearbeitet. Dieses Kapitel beschreibt die Ergebnisse und Erfahrungen im Kontext der Arbeitspaketgruppe 1200 und geht an gegebener Stelle auf Teilaspekte der Arbeitspakete 1210 und 1220 ein.

3.1.1. Datenarten

Bevor auf die Architektur und Nutzbarmachung der Data-Lake-Technologie eingegangen werden kann, soll erläutert werden, was im Data Lake gespeichert werden muss. Das Arbeitspaket 1210 „Terrestrial Data Sources“ beschäftigt sich mit diesem Thema.

Zu Beginn der Studie war davon ausgegangen worden, dass alle maritimen Eingangsdaten, die zur Anomalie-Bestimmung dienlich sein können, in einem Data Lake gespeichert werden. Aus der Sicht der terrestrischen Datenquellen sind dies insbesondere AIS-Daten und Radardaten, wie sie z. B. von Küstenüberwachungsradaren erfasst werden können. Aufgrund dauerhafter Speicherung aller Eingangsdaten unter Einbeziehung historischer Daten muss von großen Datenmengen und erheblichem Speicherbedarf ausgegangen werden. Historische Daten werden für Erkennung und Analyse von Anomalien auf einem historischen Datenbestand (Forensik) genutzt. Außerdem können aus historischen Daten Erkenntnisse für Live-Anomalie-Detektionen abgeleitet werden. Weiterhin sollen bestimmte gekennzeichnete (gelabelte) Datenbestände zur Simulation und Validierung hinterlegt werden können. Unter Berücksichtigung dieser Aspekte ist die Nutzung der Data-Lake-Technologie gerechtfertigt.

3.1.2. Datenquellen

3.1.2.1. Radardaten

In der Anlaufphase der Studie stand bezüglich der Bereitstellung der Datenquellen die Akquise im Vordergrund. Schnell konnten jedoch einige Hindernisse in der konkreten Bereitstellung von Daten – sowohl für Live-Betrieb als auch für historische Datenbestände – erkannt werden.

Da vorgesehen war, Erkenntnisse nicht nur aus AIS-Datenbeständen zu gewinnen, sondern auch aus durch Radardaten angereicherten, fusionierten und ergänzten Beständen, wurden Bestrebungen durchgeführt, an geeignete Radardaten zu gelangen. Neben der reinen Verfügbarkeit solcher Daten erwies sich der Umstand als hinderlich, dass Radardaten passend in Zeit und Raum zu einem AIS-Daten-Pendant sein müssen, denn nur so kann eine Fusionierung sinnvoll erfolgen. Diese Bestrebungen blieben erfolglos, geeignete Radardaten konnten weder über einen Verbundpartner noch über einen assoziierten Verbundpartner bereitgestellt werden.

Da sich das Arbeitspaket 1220 „Data Fusion Services“ explizit mit der Thematik der Fusion zwischen Radar- und AIS-Daten beschäftigt, indem Bestandssoftware der Airbus DS in Ulm in den Demonstrator zu IntelliMar einfließen sollte, war an dieser Stelle das Arbeitspaket nur eingeschränkt realisierbar. Anstelle der Fusionierung von Radar- und AIS-Daten beschäftigte sich dieses Arbeitspaket mit der eingehenden Betrachtung von notwendigen Aufbereitungsschritten des AIS-Rohdatenmaterials zu besser verwertbaren Zwischenprodukten, dem AIS-Data-Cleaning.

3.1.2.2. AIS-Daten des WSV

Der assoziierte Projektpartner WSV stellte nach einem WSV-internen Auswahlverfahren für die Studie IntelliMar einen Datensatz basierend auf AIS-Daten bereit.

Die Studie IntelliMar verfügte somit über einen historischen Datenbestand der Jahre 2012 und 2013 mit der Abdeckung der deutschen Küstengebiete an Nord- und Ostsee. Die Auswahl des Zeitraums berücksichtigt die Vorgaben durch die seit Juni 2018 geltende DSGVO. Die Datenmenge der Rohdaten betrug ca. 0,5 GB komprimierter Datensätze im NMEA-0183-Datenformat. Da nicht der gesamte Meldungsvorrat der Rohdaten für die Anomalie-Detektion relevant ist, muss die Datenmenge durch Verarbeitung des von IntelliMar Services bereitgestellten Daten-Importers bestimmt werden. Exemplarisch wurde dem WSV-Datenbestand der gesamte Tagesbestand zum 28. Mai 2012 entnommen und ausgewertet:

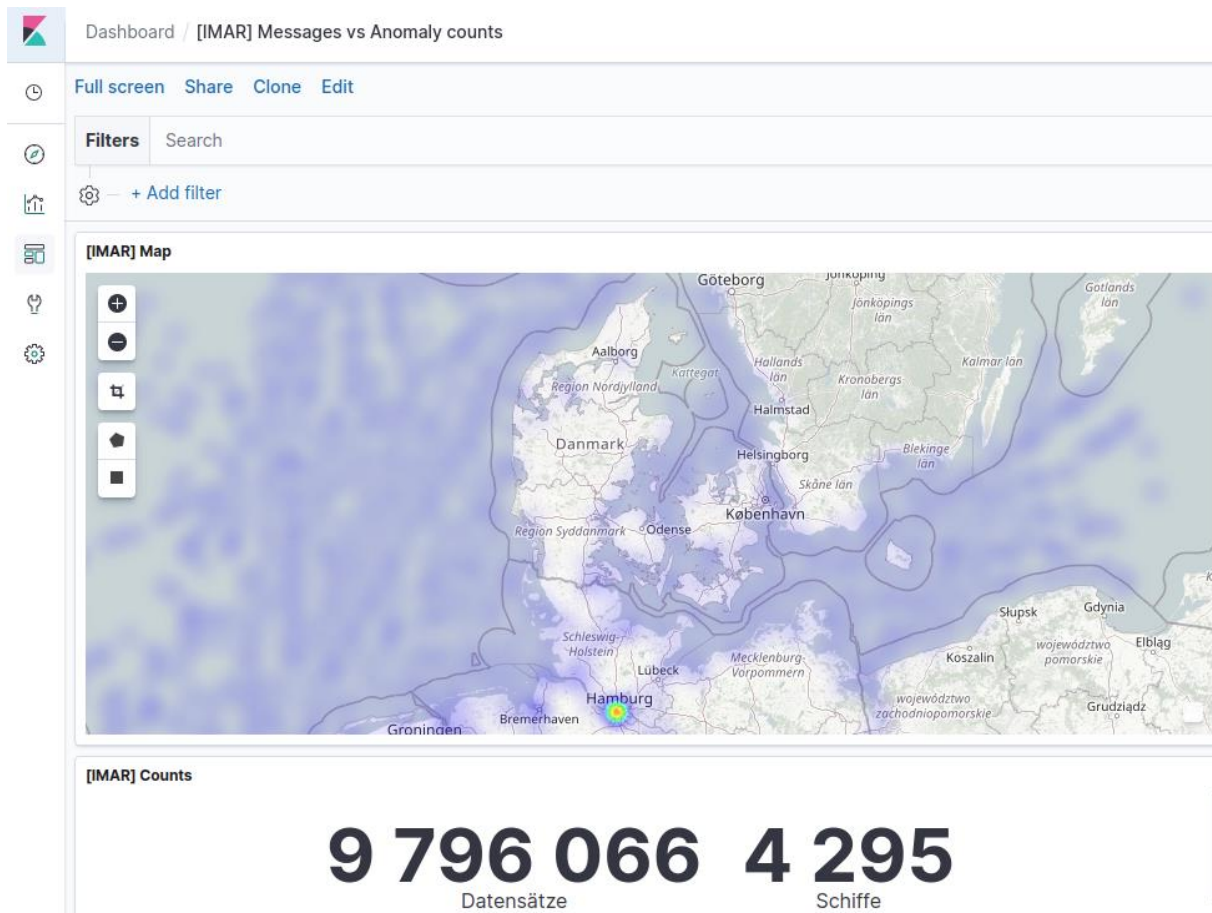


Abbildung 3-1: Verteilung und Anzahl verwertbarer AIS-Daten des WSV-Datensatzes zum 28.05.2012

Die Auswertung ergab für diesen Tag eine Anzahl verwertbarer Datensätze von knapp 10 Millionen. Diese Daten konnten 4295 Schiffen zugeordnet werden. Auffallend ist die sehr homogene Verteilung der Meldungsdaten im Bereich der Deutschen Nord- und Ostseeküste. Dies schließt auf eine sehr hohe Datenqualität. Es wurden mehrere Datenquellen kombiniert und der Datenbestand diesbezüglich bereinigt und verdichtet. Auffällig ist eine sehr hohe Datendichte im Bereich Hamburger Hafen, was auf eine deutlich höhere Dichte an Empfängern und Quellen und das hohe Verkehrsaufkommen in dieser Region zurückzuführen ist.

Diese Originaldaten sind nicht gelabelt, also nicht mit Meta-Informationen über Anomalien versehen.

3.1.2.3. AISHub

AISHub (AIS data sharing and vessel tracking by AISHub, 2021) ist ein weltweit agierender Datenanbieter für AIS-Daten. AISHub stellt eine Internetplattform zum Zusammenführen von AIS-Daten verteilter Empfängerstationen dar. Die Empfängerstationen werden von freiwilligen Unterstützern der Plattform bereitgestellt. Die Anschaffungskosten eines Empfängers sind überschaubar, die zur Verbindung zu AISHub notwendige Software ist kostenfrei. Das Bereitstellen erlaubt dem Unterstützer die Verwertung des gesammelten vollständigen Datenstroms. Die Nutzung der Plattform ist ansonsten kostenfrei. Zum Zeitpunkt der Berichtserstellung verfügte das AISHub-Netzwerk über 620 beitragende Empfangsstationen weltweit.

Für die Studie IntelliMar wurde entschieden, diese Plattform zu nutzen. Zu diesem Zweck wurden vom Studienkoordinator Airbus DS Bremen 2 Antennen errichtet und betrieben. Der dadurch zugängliche Live-Datenstrom der AISHub-Plattform wurde möglichst kontinuierlich empfangen und im Data Lake sowohl in Rohdatenform (NMEA-0183-Sätze) als auch in aggregierter Form hinterlegt (siehe Kapitel 3.2.2.4).

Über dieses System wurden pro Tag ca. 1 GB komprimierte Rohdaten empfangen und gespeichert. Eine Stichprobe über die Dauer des Tages 28. Januar 2021 ergibt folgende Datenverteilung:

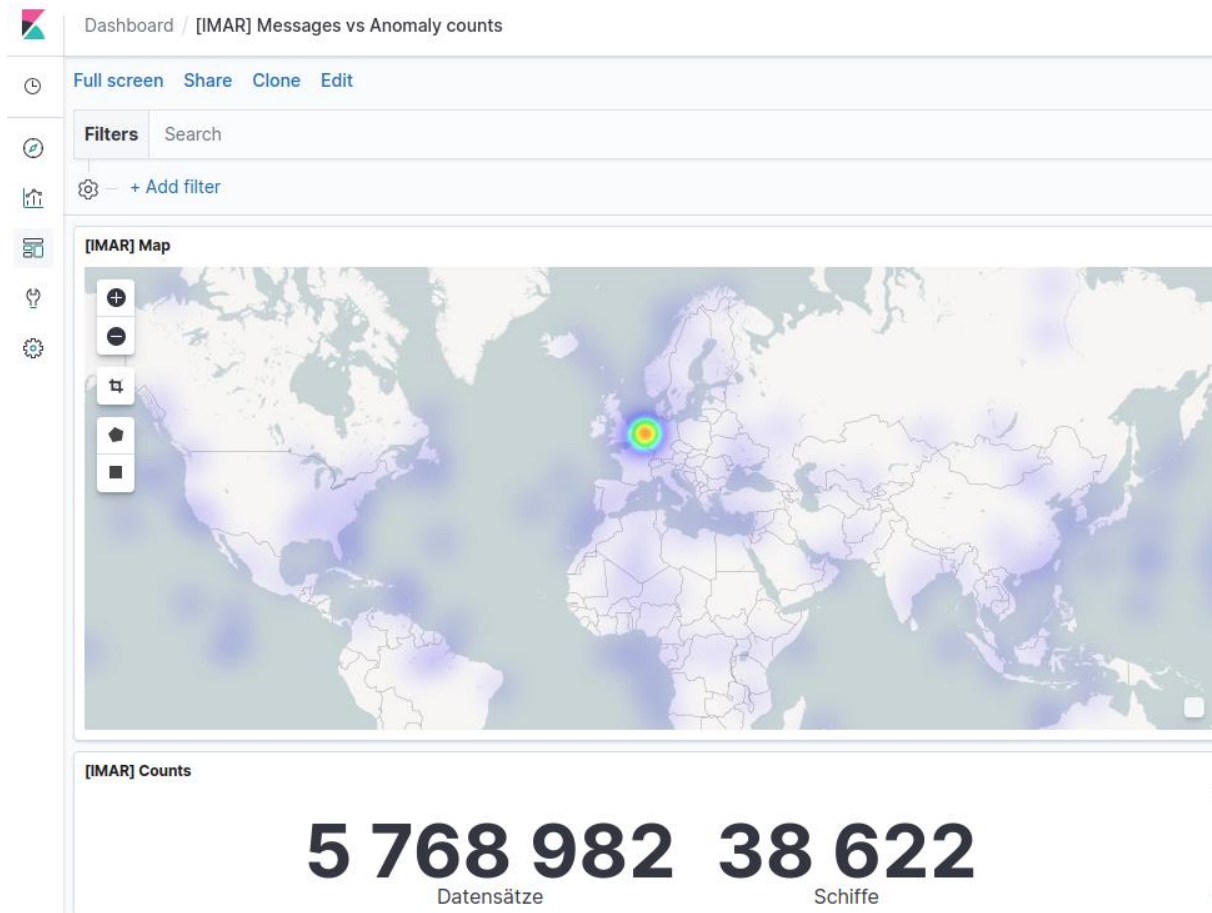


Abbildung 3-2: Verteilung und Anzahl verwertbarer AIS-Daten des Providers AISHub am 28.01.2021

Da die Verteilung der Empfänger des AISHub-Systems als weltweit angenommen werden kann, wurde für die Betrachtung eine weltweite Verteilung erstellt. Für einen Tag wurden ca. 5,8 Millionen Datensätze generiert, die 38622 Schiffen zugeordnet werden konnten. Im Vergleich zu den WSV-Daten ist dies eine geringe Datendichte. Die WSV-Daten erzeugen etwa doppelt so viele Datensätze allein aus dem Gebiet der Deutschen Küste. Wie im Bild zu erkennen, ergibt sich auch hier ein gewisser Hotspot der Datendichte an der Niederländischen Küste. Die Datendichte an der Deutschen Küste ist jedoch ausreichend hoch, um die Daten im Rahmen der IntelliMar-Studie sinnvoll nutzen zu können. Auffallend an dieser Verteilung ist offensichtlich eine beträchtliche Fehlerrate dieser Daten, da verschiedenes Datenaufkommen nicht auf an der Küste oder auf Inseln platzierte Empfänger zurückzuführen ist. Insbesondere zeichnet sich eine Häufung von Daten am Null-Meridian und am Äquator sowie am Schnittpunkt der beiden Linien ab. Dies lässt sich durch nicht funktionierende oder schlecht gewartete Hard- u. Software der Antennenbetreiber erklären. Eine genaue Analyse oblag nicht dem Umfang der IntelliMar-Studie, da sich diese Studie auf den Bereich Deutsche Küste konzentriert. In diesem Bereich wurden auch bei den AISHub-Daten kaum Auffälligkeiten registriert, was auf eine deutlich höhere Empfängerdichte und –qualität schließen lässt.

Diese Daten sind naturgemäß nicht gelabelt.

3.1.2.4. Test- und Validierungsdaten des Verbundpartners OFFIS; Oldenburg

Da der Verbundpartner OFFIS mit Sitz in Oldenburg über ein eigenes Netz aus AIS-Empfängern an Gebieten der deutschen Nordseeküste verfügt, wurde zu Beginn der Studie in Erwägung gezogen, Daten aus diesem System im Demonstrator von IntelliMar zu verarbeiten. Nach Anfrage wurde jedoch schnell klar, dass wegen der zum Startzeitpunkt der Studie 2018 noch neuen Regelungen, die sich aus der seit Mitte 2018 geltenden DSGVO ergeben, zunächst unklar erschien, ob die Daten im gewünschten Rahmen genutzt werden dürften. Der Anfrage dieser Datenquelle wurde letztendlich nicht stattgegeben.

Das OFFIS-Institut hat jedoch entsprechend dem Studien-Auftrag Datenbestände für Test- und Validierungszwecke bereitgestellt. Im Gegensatz zu allen anderen Datenquellen enthalten diese Daten eine Klassifizierung bezüglich der darin enthaltenen Anomalien, sodass ein Test oder eine Validierung einfach möglich sind. Im Vorfeld einer Validierungskampagne wurde ein Datensatz veröffentlicht, der die Aufschlüsselung enthaltener Anomalien beinhaltet, sodass Verbundpartner mit Anomalie-Detektoren in der Entwicklung in der Lage waren, die Güte und Geschwindigkeit ihrer Detektoren zu ermitteln. Zu den Validierungskampagnen selbst wurden Datensätze mit nur dem Verbundpartner OFFIS bekannten Anomalien bereitgestellt, gegen welche die Validierung letztendlich durchgeführt wurde.

Details zum Ablauf der Validierungskampagne aus Sicht des Teilvorhabens IntelliMar-Service sind im Kapitel 3.4 Validierungskampagnen nachzulesen. Weitere Details zu den Test- und Validierungsdaten sind dem Schlussbericht des Verbundpartners OFFIS zu entnehmen.

3.1.3. Data Lake

Hauptaufgabe eines Data Lakes ist das Speichern von Daten. Im Falle eines Data Lakes soll dies für aus heutiger Sicht große Datenbestände erfolgen und Aspekte wie redundante und damit sichere Lagerung der Daten sowie massiv parallele Datenverarbeitung berücksichtigen. Da die Data-Lake-Technologie bereits existiert, sollte im Rahmen der Arbeitspaketgruppe 1200 insbesondere geprüft werden, wie und wofür diese Technologie für den im Rahmen der IntelliMar-Studie bereitzustellenden Demonstrator effektiv genutzt werden kann.

Der Data Lake wurde auf einem vom Verbundkoordinator Airbus DS in Bremen bereitgestellten Hardware-Cluster errichtet. Art und Konfiguration des Hardware-Clusters können dem Schlussbericht der Airbus DS Bremen (Airbus Defence and Space GmbH, 2021) entnommen werden.

3.1.3.1. Data Lake im Projekt IntelliMar

Der Begriff „Data Lake“ entstammt der Welt des Big-Data-Bereichs der IT. Ein Data Lake dient der Speicherung und Verarbeitung großer Datenmengen. Die Verarbeitung wird durch im Big-Data-System bereitgestellte Rechenknoten parallelisiert durchgeführt. Die Ergebnisse dieser Datenoperationen werden wieder im Data Lake abgelegt, sodass ein nachfolgender Arbeitsschritt direkt Zugriff auf die Daten erlangen kann. Zur Realisierung der Parallelisierung

setzt das System auf das Bereitstellen redundanter Kopien auf verteilten Rechenknoten, um den Transportaufwand der Daten von einem entfernten Rechenknoten zu einem andern zu minimieren. (Mark Grover, 2015) Die folgende Abbildung soll dies schematisch darstellen.

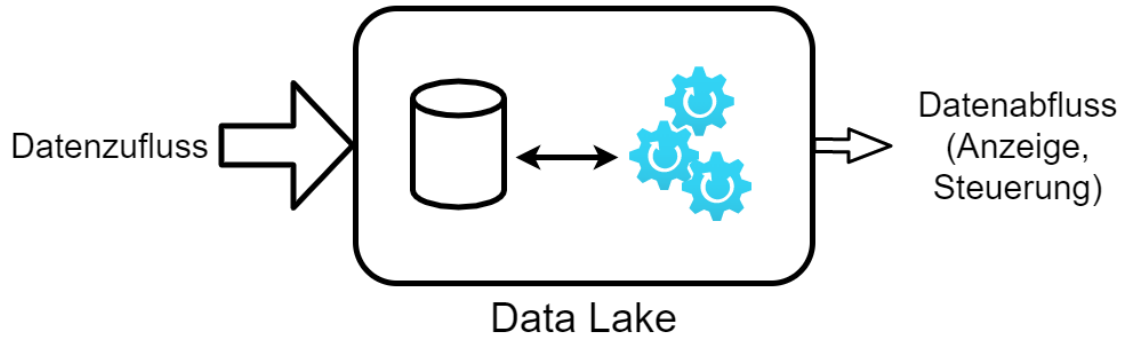


Abbildung 3-3: Schematische Darstellung eines Data Lakes im Big-Data-Umfeld

Hervorzuheben bei diesem Konzept ist die erkennbare Asymmetrie zwischen Datenzufluss und Datenabfluss. Um mit Massendaten arbeiten zu können, benötigt man einen Zulauf in den Data Lake mit sehr hohem Durchsatz. Sind die Daten im Data Lake, werden sie vornehmlich im Data Lake bewegt. Dies wird durch optimierte Datenspiegelung, breite Datenverbindungen unter den Rechenknoten und Parallelisierung erreicht. Da die Verarbeitung bis zum Resultat im Data Lake erfolgt, existiert nur eine sehr schmalbandige Datenausleitung aus dem Data Lake. Sie wird hauptsächlich zur Ergebnisanzeige sowie zur Steuerung und Kontrolle der Prozesse im Data Lake genutzt.

Das Datenverteilungssystem des Demonstrators zur IntelliMar-Studie sieht den Data Lake jedoch in einer anderen Rolle: Der Data Lake wird sehr wohl an einer zentralen Stelle im Datennetzwerk untergebracht, steht den angeschlossenen Projektpartnern jedoch nicht mit breitbandigem Anschluss zur Verfügung. Der Data Lake soll Eingangsdaten wie AIS-Daten und Ergebnisdaten wie Anomalie-Berichte vorhalten. Folgende Abbildung stellt diesen Sachverhalt schematisch dar.

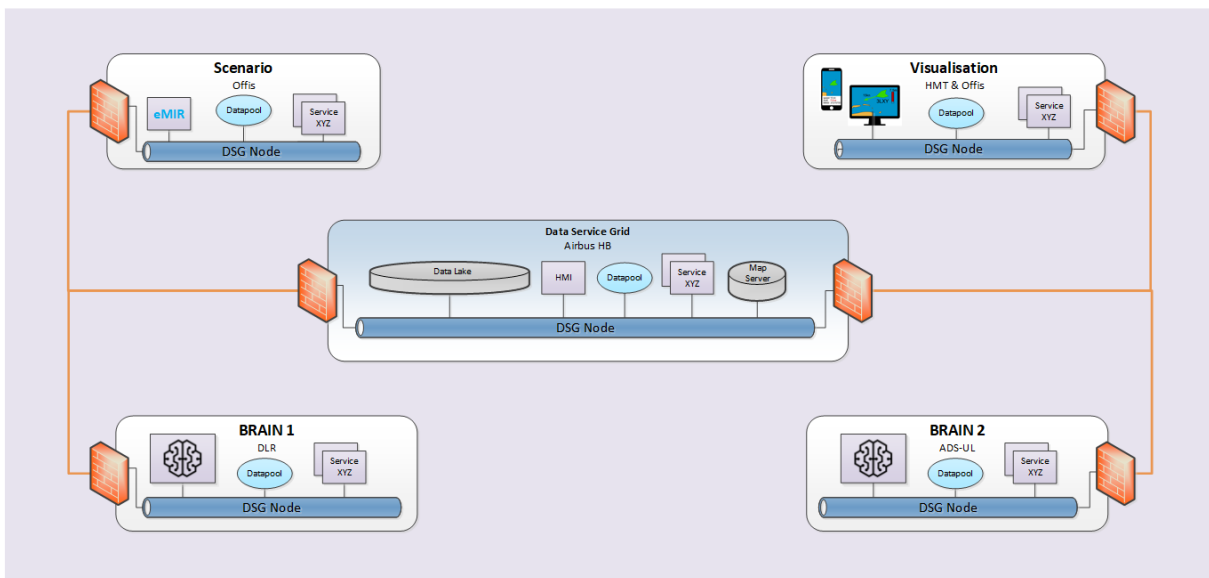


Abbildung 3-4: Gesamtarchitektur IntelliMar-Demonstrator

In der Mitte der Grafik findet sich der Data Lake. Ihm zur Seite gestellt sind weitere zentrale Dienste. Verbunden sind diese Dienste mit einem im Rahmen der IntelliMar-Studie entwickelten Datenbus, dem Data Service Grid, kurz DSG. Dieses DSG unterstützt nicht nur die Verbindung der Dienste an einem Standort, sondern stellt diese auch den Verbundpartnern entfernt bereit. Hierzu sind die entfernten Knoten über eine Internetverbindung unter Einhaltung von Schutzmaßnahmen angebunden. Entfernte Knoten sind in den Standorten Oldenburg (Verbundpartner OFFIS und HuMaTects), Neustrelitz (DLR) und Ulm (Airbus DS) installiert. Der zentrale Knoten befindet sich beim Verbundkoordinator Airbus DS in Bremen. Die Verbundpartner erhalten so Zugang zu den Daten im Data Lake und können Dateneinspeisung, Berechnungen und Anzeige an ihren Standorten durchführen. Weitere Details zum DSG sind dem Schlussbericht des Verbundpartners Airbus DS in Bremen (Airbus Defence and Space GmbH, 2021) zu entnehmen.

Schnell wird klar, dass das Konzept des DSGs optimal für entfernte Kommunikation und Arbeitsteilung ausgelegt ist. Jedoch ist aus dieser Sicht der volle Leistungsumfang eines Data Lakes wie oben definiert nur eingeschränkt nutzbar:

- Das Einspielen von Daten geschieht dann mit geringer Datenrate, wenn es über einen angeschlossenen entfernten Knoten wie dem Knoten des Verbundpartner OFFIS erfolgen muss.
- Die einzelnen Verbundpartner nutzen ihre eigenen Rechenkapazitäten, um ihre Auswertung für Anomalien durchzuführen; sie operieren meist nicht direkt im Data Lake
- Der Data Lake muss nicht nur performant Daten aufnehmen können, sondern den Verbundpartnern wieder bereitstellen können. Es ist also eine symmetrische Auslegung zwischen Zu- und Ablauf zu realisieren.

DSG-seitig wird zur Kompensierung der schmalbandigen Internetverbindung ein Zwischenspeicher bei den Verbundpartnern eingerichtet, der sogenannte Data Pool. Primär bezieht ein entfernt angeschlossener Partner seine Daten aus seinem lokalen Data Pool; sollten die Daten dort nicht vorhanden sein, werden sie transparent aus dem Data Lake nachgeladen.

Zusammenfassend ergibt sich aus der Sicht der Komponente Data Lake folgende schematische Darstellung:

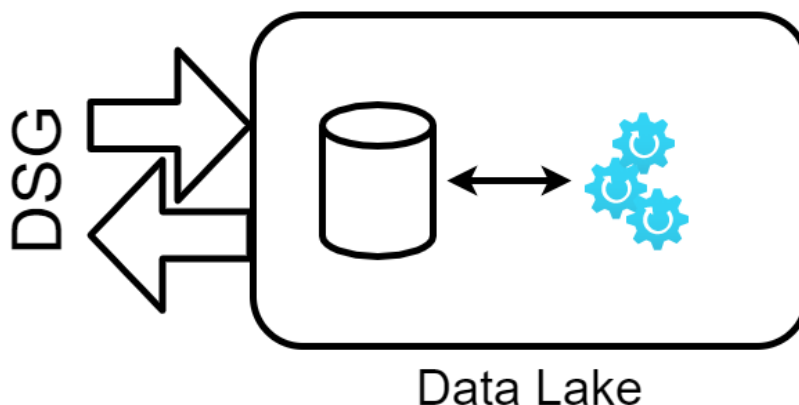


Abbildung 3-5: Schematische Darstellung des Data Lakes bei IntelliMar

Zu erkennen ist, dass der Data Lake von IntelliMar einen symmetrischen Zu- und Ablauf besitzt sowie erwartungsgemäß Daten halten kann und optional auch lokal verarbeiten kann.

Aufgrund dieser Rahmenbedingungen wurde der Data Lake speziell ausgelegt, erweitert und mit eigenen Software-Komponenten zum DSGVO anschlussfähig gestaltet und realisiert.

Im Folgenden wird die innere Architektur dieses Data Lakes detailliert vorgestellt. Die Tätigkeiten hierzu fallen aus Sicht von IntelliMar Services hauptsächlich in das Arbeitspaket 1210, da dort insbesondere die Speicherung und Wiedergabe-Funktionalität für terrestrische Sensordaten niedergeschrieben ist.

3.1.3.2. Architektur des Data Lakes

Unter der Berücksichtigung der im vorigen Kapitel genannten Rahmenbedingungen entsteht für den Data Lake des Demonstrators im Projekt IntelliMar eine Mischung aus verschiedenen Produkten, Werkzeugen und Techniken. Die einzelnen Elemente sollen in diesem Kapitel vorgestellt und ihr Zusammenspiel erklärt werden.

Zunächst zusammenfassend die zu beachtenden Eigenschaften des IntelliMar-Data-Lakes:

- Sichere Speicherung von Rohdaten und Zwischenprodukten
- Sichere Speicherung von Ergebnisdaten
- Bereitstellen von vorhandenen Datenbeständen verschiedenster Quellen wie eingelieferte Rohdaten und Schiffdatenbanken
- Symmetrische Datenbandbreite zum DSGVO
- Wahlfreier Zugriff auf Daten im Data Lake
- Sicherheitskonzept für den Zugang zu Daten

Nachfolgende Abbildung greift die Data-Lake-Struktur von IntelliMar auf und detailliert den internen Aufbau.

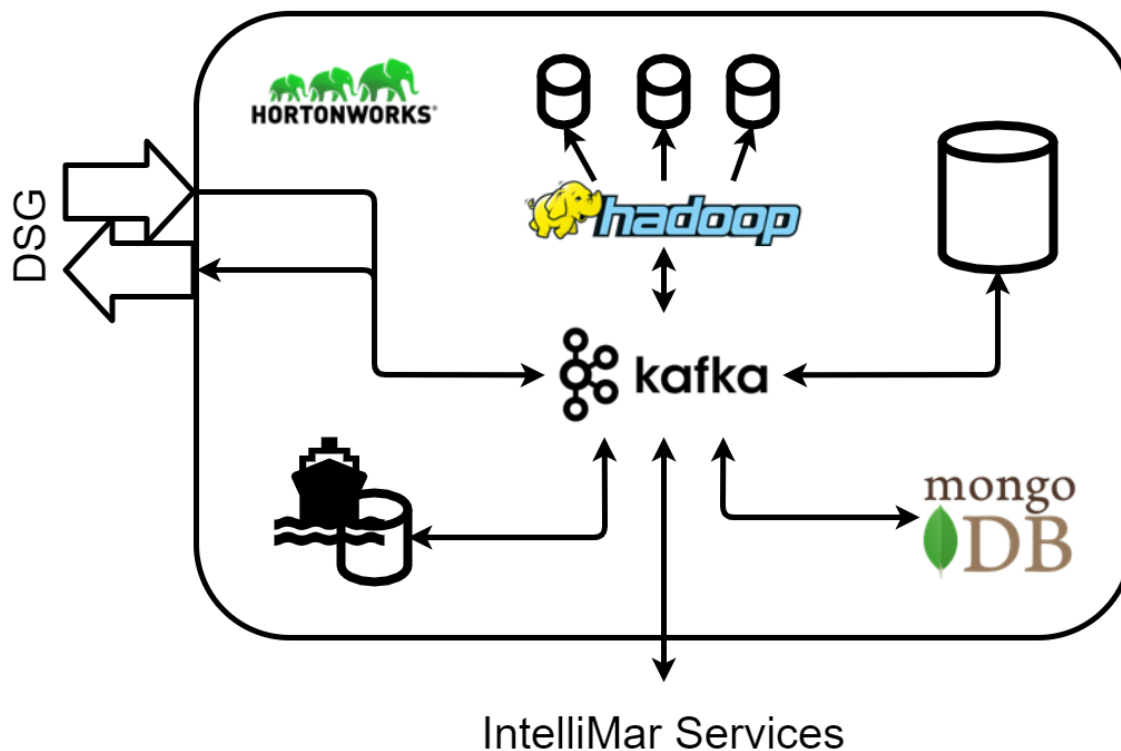


Abbildung 3-6: Architektur des IntelliMar-Data-Lakes

3.1.3.3. Komponentenauswahl

Im Folgenden wird die Komponentenauswahl dargelegt und begründet.

Hadoop Distributed File System (HDFS)

Das Hadoop Distributed File System ist der Standard-Vertreter, wenn es im Big-Data-Kontext um die Sicherung und Verarbeitung großer Datenmengen geht. HDFS ist im Jahr 2005 erstmals erschienen. Seit 2006 wird HDFS unter der Produktbezeichnung Apache Hadoop (Apache Hadoop, 2021) geführt. Es ist ein Top-Level-Projekt der Apache Software Foundation und wird unter der Apache-Lizenz angeboten. Apache Hadoop stellt daher ein von einer breiten Community gefördertes und gepflegtes Produkt dar.

Aus diesem Grund wird auch im Rahmen des IntelliMar-Demonstrators auf das HDFS gesetzt.

Da es sich bei Apache Hadoop nicht ausschließlich um ein Dateisystem im engeren Sinne handelt, besteht Hadoop seinerseits wieder aus vielen verschiedenen Einzelkomponenten. Die Einzelkomponenten wiederum entstammen größtenteils ihrerseits wieder aus Projekten der Apache Software Foundation, werden jedoch nicht kohärent unter dem Mantel von Apache Hadoop zu einem System zusammengeführt. Die Bildung eines konkreten, für die eigenen Zwecke geeigneten Hadoop-Systems obliegt dem Nutzer von Hadoop.

Deswegen bildeten sich Unternehmen außerhalb des Apache-Perimeters, die sich auf die Bündelung passender HDFS-Software-Bestandteile und Erweiterungen spezialisiert haben.

So sind vorkonfigurierte und leicht konfigurierbare Hadoop-Systeme als Software-Distributionen unter verschiedenen Lizenzmodellen verfügbar.

Mit diesem Hintergrund entschied man sich bei der Wahl eines geeigneten Hadoop-Systems für den IntelliMar-Demonstrator für das Produkt Hortonworks Data Platform HDP des Dienstleisters HortonWorks. Im Jahr 2018, zum Startzeitpunkt der IntelliMar-Studie, wurde das Produkt HDP unter einer freien Lizenz zur Verfügung gestellt.

Zu Beginn des Jahres 2019 fusionierten die Unternehmen HortonWorks und Cloudera (Cloudera, Inc., 2021), was sich entscheidend auf die Lizenzpolitik des bis dato freien Produktes HDP auswirkte. Ein Umstieg auf entsprechende Cloudera-Produkte wurde wirtschaftlich verworfen. Stattdessen wurde am Einsatz des immer noch lizenzoffenen, aber nicht weiter gepflegten HDP festgehalten.

Ein nicht zu unterschätzender Vorteil bei der Wahl von Apache Hadoop bzw von HortonWorks HDP ist das Vorhandensein eines Sicherheitsframeworks für Daten. Der Zugriff auf Daten ist konfigurierbar und durch Benutzung von Standardverfahren in weitere Sicherheitskonzepte integrierbar.

Hardware-Dateisysteme

Apache Hadoop stellt an die zugrundeliegende Hardware vergleichsweise hohe Anforderungen. Dies liegt einerseits an der im HDFS vorgehaltenen Redundanz in der Datenhaltung als auch an Hadoops Möglichkeiten, diese Daten zu verarbeiten.

Zur Ressourcenschonung wurde dem angelegten HDFS ein einfaches Dateisystem auf Betriebssystembasis zur Seite gestellt. Aufgabe ist es, Daten zu speichern, von denen ausgegangen wird, dass sie nicht durch Hadoop bearbeitet werden müssen, sondern nur verwahrt werden. Idealerweise sollten die hier gelagerten Daten keine Originale sein, sondern nur eingebrachte Kopien, um nicht durch zusätzliche Datensicherungsmaßnahmen ungewollt weitere Ressourcen zu verbrauchen. Einer Backup-Strategie dieser Daten auf Basis eines externen Band-Backups steht nichts im Wege.

Ein Beispiel für die Nutzung dieses Dateisystems ist die Lagerung der durch die WSV bereitgestellten AIS-Datenbestände. Eine punktuelle Vereinnahmung der Daten in das HDFS erfolgt von der Kopie im Dateisystem des Hardware-Clusters.

Apache Cassandra

Apache Cassandra (Apache Software Foundation, 2021) ist ebenfalls ein prominenter Vertreter des Big-Data-Umfeldes und musste daher ebenfalls für eine Verwendung im IntelliMar-Data-Lake untersucht werden. Bei Cassandra handelt es sich um einen Vertreter der sogenannten NoSQL (Not only SQL) Datenbanken.

Im Gegensatz zu relationalen Datenbanken, die via SQL-Abfragen einen wahlfreien Zugriff auf Daten und Daten-Aggregationen erlauben, arbeitet Cassandra besonders effektiv, wenn die Fragestellungen und Anwendungsfälle im Vorfeld bekannt sind. Die Datenbanken werden beim Anlegen des Datenbestandes auf genau diese Abfragen indiziert. Die Wahl des

Indexes ist entscheidend für die Performance bei der Antwortermittlung und kann nicht leicht geändert werden.

Ein Vorteil des Cassandra-Systems im Gegensatz zu relationalen Datenbanken ist die Skalierbarkeit bei der zu verarbeitenden Datenmenge im Data Lake. Insbesondere wurde Apache Cassandra für IntelliMar erwogen, weil die in der Studie anfallenden Daten durchaus strukturiert sind, also definierte Datenfelder besitzen. AIS-Daten sind ein Vertreter dieser Datengattung.

Letztendlich wurde aber gegen die Aufnahme von Apache Cassandra in den IntelliMar-Data-Lake entschieden, da Cassandra neben Apache Hadoop extrem hohe Ressourcen vom Hardware-Cluster einfordert, die jedoch nur im begrenzten Umfang zur Verfügung stehen. Zudem setzte man auf Wahlfreiheit bei der Gestaltung von Anfragen gegen das System, da zu Beginn der Planung des Data Lakes noch nicht erfasst werden konnte, welche Indizes angelegt werden müssen. Im Hinblick auf die Verwendung des IntelliMar-Demonstrators zu Studienzwecken wollte man sich Wahlfreiheit erhalten.

Für die Architektur eines Produktivsystems hingegen sollte Apache Cassandra als Vertreter des Data Lakes in Betracht gezogen werden. Da im Produktiv-Fall der Fragenkatalog bekannt ist, kann die Datenbank passend angelegt werden, sodass beispielsweise auf gespeicherten historischen Bewegungsdaten schnell und effektiv Bewegungsspuren (Tracks) und Bewegungsprofile (Routen) ermittelt werden können.

Dokumentendatenbank Mongo-DB

MongoDB (MongoDB, Inc., 2021) ist ein dokumentenorientiertes NoSQL Datenbankmanagementsystem. Es erschien im Jahr 2009 und wird vom Unternehmen MongoDB, Inc. entwickelt. Im Big-Data-Umfeld versteht man unter Dokumentenorientierung hauptsächlich die Nutzung von JSON-Dokumenten. Dies sind strukturierte Dateiformate, deren Handhabung nicht nur im Big-Data-Bereich Standard sind. Der IntelliMar-Demonstrator legt beispielsweise Anomalie-Berichte oder Schifflogbucheinträge in diesem Format ab. Aufgrund der auch bei MongoDB üblichen Redundanz durch Auslegung mehrerer Datenbankknoten auf verschiedenen Hardware-Cluster-Einheiten zur Skalierung liegen die Ergebnisdaten wie Anomalie-Berichte in ausreichend gesicherter Form vor.

Dokumentendatenbanken gemein ist die Eigenschaft, Abfragen bezüglich gespeicherter Inhalte bestandsweit durchzuführen, auch wenn die Struktur der Dokumente nicht einheitlich ist. Diese Funktion wird in IntelliMar für Anomalie-Berichte genutzt. Anomalie-Berichte im JSON-Format dürfen entsprechend den Bedürfnissen bezüglich ihrer Dokumentenstruktur variieren. Z. B. enthalten Anomalie-Berichte aus regelbasierten Detektoren nur Felder, die für eine bestimmte Anomalie relevant sind, wohingegen Anomalie-Berichte von KI-Detektoren oft ganze Wahrscheinlichkeitsmatrizen bezüglich der Einordnung zu verschiedenen Anomalie-Kategorien enthalten können. Allen Anomalie-Berichten gemein sind jedoch Felder, die das betroffene Objekt (Schiff), den Zeitpunkt und die Position angeben. Unter der Maßgabe, solche Schlüsselfelder stets gleich zu benennen, können Suchanfragen für zum Beispiel alle Anomalien in einem Seebereich zu einem gewissen Zeitfenster über alle Anomalie-Berichte leicht ermittelt werden.

Neben der Bereitstellung und Anbindung der MongoDB war es Aufgabe der IntelliMar-Studie Datentypen zu entwerfen, die diesen Anforderungen gerecht werden. Die Erläuterung des konkreten JSON-Dokumentenaufbaus findet sich im Kapitel 3.2.2.2 Grundlagen AIS-Daten und internes AVRO-Schema AisMessage in diesem Bericht

Der tatsächliche Ressourcenbedarf dieser Datenbank ist gering oder kann gering dimensioniert werden, da im Vergleich zu den AIS-Daten die Anomalie-Berichtdaten ein vernachlässigbares Volumen haben. Insgesamt ist MongoDB eine nützliche und effektive Bereicherung des IntelliMar-Data-Lakes.

Schiffsdatenbanken und weitere Kundendatenbanken

Der Data Lake soll auch Platz für Produkt-Datenbanken verschiedener Art und Anbieter bereitstellen. Ein Beispiel sind Schiffsdatenbanken. Liegen solche Daten in speicherbarer Form vor, sollen sie im Data Lake so abgelegt werden, dass sie möglichst einfach wieder verfügbar gemacht werden können. Heutzutage stehen solche Datenbanken unter Umständen ausschließlich als gemieteter externer Service zur Verfügung. In so einem Fall kann die Datenbank selbst nicht im Data Lake angelegt werden. Dennoch sind sie Bestandteil der Data-Lake-Architektur, da auch diese Datenbanken eine Schnittstelle zur Abfrage besitzen müssen und zumindest in Interaktion mit dem DSG treten, um Daten auf Anfrage im Verbund zu verteilen. Diese Aufgabe fällt in den Architektur-Bereich des Data Lakes bzw. dessen Anbindung an das DSG.

3.1.3.4. Interaktion der Komponenten

Entscheidend für die Funktion des Data Lakes ist nicht nur die sorgfältige Auswahl beteiligter Komponenten sondern auch deren Verbindung untereinander, sodass die Komponenten insgesamt ein organisiertes Ganzes – den Data Lake – darstellen. Die folgenden Kapitel zeigen die Lösung für IntelliMar.

Apache Kafka

In der Abbildung 3-6 ist die aus der Interaktion entstehende Dynamik durch Pfeile symbolisiert, die zentral auf die Komponente „Kafka“ weisen.

Apache Kafka (Apache Software Foundation, 2021) ist ein Software-Produkt der Apache Software Foundation zur Verarbeitung von Datenströmen. Im Gegensatz zu Apache Hadoop, das auf Bestandsdaten arbeitet, liegt der Schwerpunkt bei Apache Kafka auf bewegten Daten. Bewegte Daten bzw. Datensätze sind im Kontext von Apache Kafka Nachrichten oder Fakten.

Betrachtet man eingehende AIS-Meldungen in diesem Kontext, erschließt sich die Nutzbarkeit von Apache Kafka sofort. AIS-Meldungen werden als Nachrichten empfangen und als solche verarbeitet und zu neuen Nachrichten umgewandelt und gespeichert. Im Rahmen des Data Lakes werden Datenbestände aus (AIS-)Nachrichten einfach mit Kafka bewegt, eine Verarbeitung ist im Kontext Data Lake optional.

Das Kapitel 3.2 AP 1500 - Airbus Brain Development zeigt, dass Apache Kafka nicht nur für den Betrieb als Infrastruktur im Data-Lake geeignet ist, sondern vielmehr eine geeignete

Infrastruktur zum Aufbau der Anomalie-Erkennung an sich darstellt. In der Abbildung 3-6 ist dies durch den Pfeil zwischen „Kafka“ und „IntelliMar Services“ symbolisiert. Anomalie-Detektoren, die Apache-Kafka nutzen erhalten so Zugriff auf den Data Lake, ohne das DSG nutzen zu müssen. Dieser Umstand kann genutzt werden, um Data Lake und Anomalie-Detektoren isoliert vom IntelliMar-Demonstrator-Netzwerk zu betreiben.

Wie auch Apache Hadoop partizipiert Apache Kafka an einem ausgereiften Sicherheitskonzept aus Standardkomponenten, sodass sowohl Zugriff auf Daten als auch deren Integrität im Bewegungsprozess garantiert werden kann.

Apache Kafka unterliegt der Apache-Lizenz und ist frei einsetzbar. Jedoch genauso wie im Fall Apache Hadoop handelt es sich bei Apache Kafka um ein System aus mehreren Komponenten, dessen Installation, Konfiguration und Wartung sich sehr aufwändig gestaltet. Daher hat sich im Umfeld Apache Kafka ebenfalls eine Distribution etabliert: Confluent (Confluent, Inc., 2021) vom gleichnamigen Hersteller. Confluent wird in verschiedenen Lizenzmodellen angeboten, stellt insbesondere aber eine Community-Variante bereit. IntelliMar Services bedient sich dieser Variante. Im Laufe der Studie IntelliMar wechselte das Lizenzmodell jedoch dreimal, was zu einer gewissen Unsicherheit in der Verwendung der Distribution führte. Zum Zeitpunkt dieses Schlussberichtes stellt die Community-Variante ausschließlich einen Single-Node-Betrieb von Kafka bereit. Dieser ist für die Aufgaben der Studie IntelliMar ausreichend, bietet jedoch keine Ausfallsicherheit. Im Rahmen des Demonstrators kann auf Ausfallsicherheit verzichtet werden, dies stellt aber einen Hindernisgrund bei einer operativen Verwendung dar. In diesem Fall ist ein anderes Lizenzmodell zu wählen oder eine angepasste Konfiguration basierend auf Apache Kafka selbst zu erstellen.

Apache Kafka Konnektoren

Die Präsenz von Apache Kafka im IntelliMar Data Lake allein reicht noch nicht aus, Daten tatsächlich zu bewegen. Apache Kafka bietet verschiedene Schnittstellen zur Verbindung von Datenquellen und Datensinken, um Daten in und durch das Kafka-System zu bewegen. Speziell für die Anbindung an Datenquellen und –senken stellt Apache Kafka das Connect-API bereit. Es bietet die Möglichkeit, Softwaremodule zu entwickeln oder zu nutzen, die sich an Gegenstellen verbinden können. Neben der reinen API werden sowohl von der Apache Software Foundation selbst als auch von Drittanbietern fertige Software-Module angeboten. Für die im Umfang des IntelliMar Data Lakes ausgewählten Software-Komponenten stehen solche Module zur Verfügung; eine eigene Entwicklung entfällt und die Nutzbarmachung reduziert sich auf das Bereitstellen einer geeigneten Konfiguration.

Die Erfahrung hat gelehrt, dass das Erstellen solch einer Konfiguration mitunter zeitintensiv sein kann, da man sich mit wechselnden Terminologien und durchaus vorhandenen Unzulänglichkeiten mancher verfügbarer Konnektoren befassen muss.

Nachfolgend ist eine Konnektor-Konfiguration zur Übernahme von NMEA-0183-Datensätzen und Speicherung dieser im HDFS des Data Lakes illustriert:

1. # Common to all connectors
2. name=imar-nmea-from-dsg-to-hdfs-parquet
3. tasks.max=1

```
4. topics=imar-ais-sessions-from-dsg
5. key.converter=org.apache.kafka.connect.storage.StringConverter
6. value.converter=io.confluent.connect.avro.AvroConverter
7. value.converter.schema.registry.url=http://hpc-3.hpc.eso-io.com:8081

8. # Connector
9. connector.class=io.confluent.connect.hdfs.HdfsSinkConnector
10. format.class=io.confluent.connect.hdfs.parquet.ParquetFormat
11. flush.size=5000000
12. # 6 hours flush time
13. rotate.interval.ms=21600000
14. rotate.schedule.interval.ms=-1
15. enhanced.avro.schema.support=true

16. # HDFS
17. hdfs.url=hdfs://hpc-1.hpc.eso-io.com
18. hadoop.conf.dir=/etc/hadoop/conf
19. logs.dir=/staging/imar/nmea-from-dsg

20. # Storage
21. storage.class=io.confluent.connect.hdfs.storage.HdfsStorage
22. topics.dir=/staging/imar/nmea-from-dsg

23. # Partitioner
24. partitioner.class=io.confluent.connect.hdfs.partitionner.FieldPartitioner
25. partition.field.name=source,sessionId,year,month,day
26. #partition.duration.ms=600000

27. #transformations go here:
28. transforms=dropKey
29. transforms.dropKey.type=io.confluent.connect.transforms.Drop$Key
```

Dateiname: imar-nmea-from-dsg-to-hdfs-parquet.properties

Der erste Block in den Zeilen 1 bis 7 gibt an, auf welchem Datenkanal (topic) die zu speichernden Datensätze eintreffen werden und wo Informationen zum verwendeten Format gefunden werden können.

Der zweite Block in den Zeilen 8 bis 15 definiert das zu verwendende Konnektor-Modul und konfiguriert es. Zudem wird festgelegt, dass spätestens nach Ankunft von 5 Millionen Datensätzen diese in HDFS geschrieben werden sollen, spätestens jedoch nach 6 Stunden. Die ideale Einstellung dieser Parameter wurde in der IntelliMar-Studie ermittelt; für die tatsächliche Performance sind dieses Parameter entscheidend. Sie hängen vom verwendeten Datenformat (hier Parquet (Apache Software Foundation, 2021)), seiner zugrundeliegenden Datenkompressionsrate und der optimalen Blockgröße von HDFS (ca. 10 x 64 MB) ab.

Block drei und vier in den Zeilen 16 bis 22 benennen den Ort für die Datenspeicherung. Block fünf mit den Zeilen 23 bis 26 konfiguriert den Konnektor so, dass die Daten in Unterverzeichnissen organisiert werden. Der Pfad zum Unterverzeichnis wird dabei automatisch nach Quelle und Datum eines Datensatzes automatisch angelegt. Es ist leicht erkennbar, dass mit solch einer Organisation Daten mittels Angabe einer Quelle und eines Zeitfensters leicht gefunden werden können, eine spezielle Datenbank zur Organisation der Daten – wie im Big-Data-Umfeld oft nötig - ist nicht notwendig.

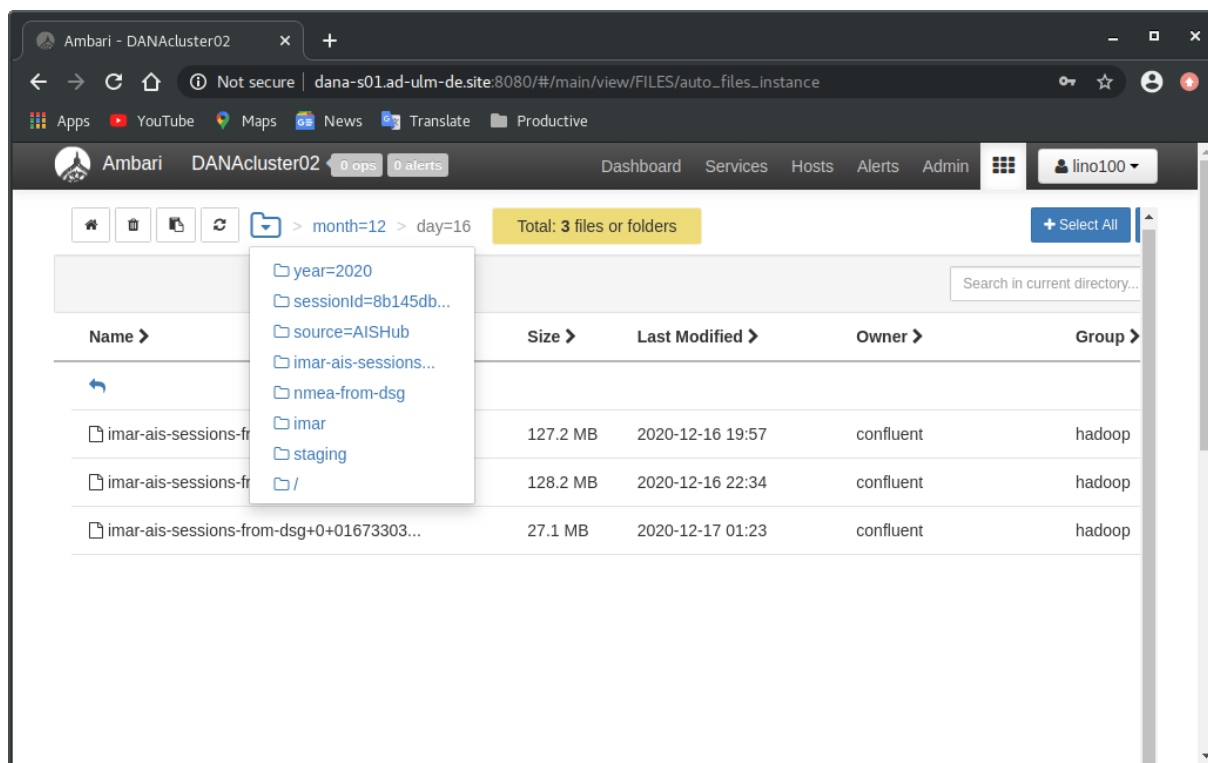


Abbildung 3-7: Beispiel-Auszug aus automatisch angelegter HDFS-Struktur

Der letzte Block mit den Zeilen 27 bis 29 nutzt das im Kafka-Connect-API vorhandene Feature der Transformationen. Datensätze, die ankommen, können vor der tatsächlichen Speicherung in gewissem Rahmen noch verändert oder ergänzt werden. An dieser Stelle wird das Speichern des Schlüssels (obligatorisch für jeden in Kafka bewegten Datensatz) unterdrückt, da er nur internen Zwecken dient.

Interessanterweise stand im Jahr 2019, als die Auslegung des Data Lakes maßgeblich durchgeführt wurde, kein Konnektor-Modul für das Entnehmen von Daten aus dem HDFS-System bereit. Dies ist wie unter Data Lake im Projekt IntelliMar erwähnt darauf zurückzuführen, dass Daten zwar breitbandig in das HDFS eingespielt werden, jedoch nicht mehr breitbandig aus dem HDFS geholt werden. Demnach schien die Implementierung dieses Konnektors überflüssig. Im Jahr 2020 wurde tatsächlich ein Konnektor vorgestellt, der auch dieses erledigen konnte. Dieser Konnektor kam jedoch nicht zum Einsatz, da mit der Nutzung ein Update der gesamten Kafka-Infrastruktur einhergegangen wäre.

Das Problem des Datenexports zur symmetrischen Anbindung an das DSG wurde stattdessen mit einem im Rahmen des Arbeitspaketes 1210 selbst entwickelten Konnektor realisiert. Dieser Konnektor nutzt nicht das Kafka-Connect-API, sondern liest und serialisiert auf Anfrage eines bestimmten Datenbestandes die Daten aus dem HDFS und schreibt sie in Richtung Apache Kafka.

Ein weiterer im IntelliMar-Demonstrator eingesetzter und konfigurierter Kafka-Konnektor ist der Konnektor zur Speicherung von Anomalie-Berichten in MongoDB. Auf eine detaillierte Beschreibung in diesem Bericht wird verzichtet.

Datenvereinnahmung und Datenlieferung

Im Rahmen des Arbeitspaketes 1210 wurden passende Methoden zum Bezug von Daten aus dem DSGVO und zur Wiedergabe von Daten in das DSGVO implementiert und in Betrieb gesetzt. Auch diese Software-Module unterstützen den symmetrischen Betrieb im IntelliMar Data Lake. Weitere Details hierzu finden sich in den Kapiteln 3.2.2.3 Datenvereinnahmungskomponenten und 3.2.2.5 Datenlieferkomponenten.

3.2. AP 1500 - Airbus Brain Development

Dieses Kapitel enthält Ergebnisse und Erfahrungen, die im Zusammenhang mit der Arbeitspaketgruppe 1500 entstanden sind. Der Bereich „Airbus Brain Development“ wurde vollständig im Rahmen des Teilvorhabens IntelliMar Services bearbeitet.

Die einzelnen Arbeitspakete sind „Brain 1 – Rule Based“ für regelbasierte Anomalie-Detektoren, „Brain 2 – Open Source ML Tools“ für einen beispielhaften Anomalie-Detektor aus dem nicht-regelbasierten Umfeld sowie „Benchmark Support“ für die Unterstützung beim Vergleich verschiedener Erkennungsverfahren.

Vor der detaillierten Beschreibung dieser Arbeitspakete werden zunächst wichtige Aspekte von Anomalien in AIS-Daten erläutert.

3.2.1. Architektur der Brains

3.2.1.1. Anomalien im AIS-Datenstrom

Dieses Kapitel stellt zunächst die für die Verarbeitung im Datenstrom wichtigen Charakteristika von AIS-Daten vor, die bei einer Architektur für Anomalie-Detektoren zu berücksichtigen sind.

AIS-Daten in Form von NMEA-0183-Meldungen können in 2 Formen vorliegen: Entweder als historische – also aufgezeichnete - Daten in Dateien oder als Live-Datenstrom, der einer bestimmten Quelle entstammt. Prinzipiell sollen Anomalie-Detektoren mit beiden Daten-Ressourcen umgehen können, ohne angepasst werden zu müssen. Da auch historische Daten zum Zeitpunkt ihrer Aufzeichnung einer Datenstrom-Quelle entstammten, liegt nahe, Anomalie-Detektoren so zu konzipieren, dass sie auf Datenströmen arbeiten. Die Analyse auf historischen Datenbeständen erfolgt dann unter Nutzung der Datenstrom-Analyse-Verfahren durch Einspielen – also dem Re-Serialisieren – der Bestandsdaten. Im Live-Daten-Fall sind die Detektoren direkt mit der Datenquelle verbunden. Einzig die Zuführungsgeschwindigkeit ist unterschiedlich. Die Live-Datenverarbeitung findet in naher Echtzeit statt, die Analyse historischer Daten erfolgt gegebenenfalls in der maximalen Verarbeitungsgeschwindigkeit des Systems. Es kommt zu einer Beschleunigung gegenüber der Echtzeitanalyse.

Ziel einer geeigneten Architektur für Anomalie-Detektion ist nun, ein System zur Anomalie-Erkennung zu schaffen, das gleichzeitig Daten mehrerer Live- und Wiedergabequellen mit

unterschiedlichen Geschwindigkeiten untersuchen kann. Es soll möglichst elastisch auf Lastveränderungen reagieren können, also Live-Quellen hinzunehmen oder ausschließen können, sowie jederzeit Wiedergabedaten in unterschiedlicher Menge verarbeiten können. Bei der Umsetzung des Teilvorhabens IntelliMar Services wurde hierauf geachtet.

Primärer Ansatz zum Entwurf eines Anomalie-Detektors ist die Analyse auf einem Datenstrom von AIS-Meldungen. Die AIS-Meldungen liegen zumeist in NMEA-0183-kodierter Form vor, weshalb im Rahmen der gesamten IntelliMar-Studie von diesem Datenformat ausgegangen wird.

AIS-Meldungsdaten erreichen einen Empfänger (beispielsweise eine UKW-Küstenantenne) zu einem bestimmten Zeitpunkt. Das AIS-System sieht eine Lieferung des Sendezeitpunkts im Datenstrom nur rudimentär vor, insbesondere lässt sich aus diesen Daten der vollständige Zeitstempel nicht ermitteln. Hieraus ergibt sich für den Empfänger die Aufgabe, den Zeitstempel zu ermitteln und zusätzlich zu jedem NMEA-0183-Datenpaket zu notieren. Sowohl für das Aufzeichnen und das spätere Wiedereinspielen der Daten als auch bei einer Nah-Echtzeitverarbeitung ist ein vollständiger Zeitstempel unerlässlich. In der IntelliMar-Studie werden NMEA-0183-Datensätze daher stets in zeitannotierter Form (Zeitstempel in UTC plus NMEA-0183-Satz, timed NMEA sentences) erwartet und verarbeitet.

Jeder AIS-Datensatz ist mit der Adresse des Senders (Transponders), der MMSI, versehen. Hieraus folgt, dass jeder AIS-Datensatz auch nachträglich einem Transponder zugeordnet werden kann. Im Normalfall besitzt ein maritimes Objekt genau einen Transponder, was wiederum die Zuordnung von Transponder-MMSIs und den Objekten ermöglicht. Maritime Objekte werden zumeist mit einer IMO-Nummer identifiziert, die Assoziation zwischen IMO-Nummer und MMSI erfolgt jedoch manuell im Schiffs-AIS-Transponder und ist somit fehleranfällig. Daher wird in der IntelliMar-Studie zumeist von dieser Assoziation abgesehen und ein maritimes Objekt direkt mit der MMSI assoziiert. Die MMSI eignet sich daher als Schlüssel in einem Detektionssystem. Maritime Objekte entsprechen im Zusammenhang mit Anomalie-Detektion den Schiffen, also denjenigen Objekten, deren Verhalten untersucht werden soll.

Charakteristisch für AIS-Meldungen ist weiterhin deren Trennung in Positionsreports sowie statische und reisebezogene Daten. Das heißt, eine AIS-Meldung für sich betrachtet ist niemals vollständig in dem Sinne, dass Sender (das Schiff), Mission und Position in einem AIS-Telegramm enthalten sind. Diese Daten sind aber für eine Anomalie-Erkennung erforderlich und in der Architektur zu berücksichtigen. Die Anomalie-Detektoren werden daher nicht direkt auf AIS-Telegrammen, sondern auf einem eigenen einheitlichen Datentypen arbeiten. Daten dieses Datentyps sollen allerdings synchron zu AIS-Telegrammen weitergeleitet werden, jedoch stets alle Schiffsdaten vorhalten. Dieser Vorgang wird als Aggregation bezeichnet.

Diese aggregierten Daten dienen als Eingangsprodukt für einen Anomalie-Detektor, dessen Ausgangsprodukt sind Anomalie-Berichte. Ziel des Anomalie-Detektors ist es, den Eingangsdatenstrom auf Besonderheiten hin zu analysieren. Da das Eingangsmaterial AIS-Datensätze sind, werden Anomalien zum Zeitpunkt der Verarbeitung einer gewissen AIS-Nachricht festgestellt. Eine solche Nachricht selbst kann keine Anomalie darstellen; zur

Ermittlung eines abnormalen Verhaltens eines maritimen Objektes, also Schiffs muss der Datenstrom über einen Zeitraum hinweg beobachtet werden. Stellt der Detektor zu einem Zeitpunkt eine Unregelmäßigkeit fest, ist selbst dies noch keine nennenswerte Anomalie, denn diese wird sich im weiteren Verlauf entweder auflösen oder erhärten. Verdeutlichen kann man sich das anhand der Anomalie „Drastic ($> 30^\circ$) change of course at open seas“ wie sie im Rahmen dieses Teilvorhabens untersucht und im Detail im Kapitel 3.2.4.2 beschrieben ist. Im Datenstrom wird zunächst eine geringfügige Kursänderung festgestellt, die sich zu 30° akkumulieren kann oder dies nicht erfüllt. Ferner ist möglich, dass die Kursänderung in eine Schlingerfahrt übergeht, was ebenfalls eine Anomalie darstellt.

Folgender Architektur-Ansatz zum Umgang mit Detektionen im Datenstrom wurde umgesetzt: Wenn ein abnormales Verhalten aus den bisher gesammelten AIS-Daten erkennbar oder absehbar wird, generiert der Detektor eine Nachricht hierzu. Nachrichten werden hier als Ereignisse oder Fakten verstanden. Eine Nachricht wird beim Gewinnen einer neuen Erkenntnis – beispielsweise bei der nächsten Positionsänderung – aktualisiert. Die Aktualisierung enthält dann eine neue Bewertung der untersuchten Anomalie. Sollte sich der Verdacht für eine Anomalie nicht erhärten, wird auch dieser Umstand als Nachricht des Detektors ausgegeben.

Aus dieser zeitlich geordneten Abfolge von Nachrichten oder Ereignissen kann eine tatsächlich relevante Anomalie abgeleitet werden. Die Relevanz aus Sicht des Detektors wird als Signifikanz in der Nachricht angegeben und wird sich in einer Berichtsfolge ändern. Nachgelagerte Analysen können aus dem Anomalie-Nachrichtenstrom unter Zuhilfenahme des Signifikanzwertes ein komplexeres Bild erschließen.

Die Datenstromanalyse erzeugt somit aus Nachrichten oder Ereignissen eine Sammlung von Anomalie-Bericht-Updates, die sowohl einer bestimmten Anomalie als auch einem bestimmten Objekt, also Schiff zugeordnet sind. Diese Berichte werden gesammelt und stehen tiefergehenden Analysen zur Verfügung. Zudem ergibt sich aus der Tatsache, dass diese Berichte Schiffen zugeordnet sind, die Möglichkeit, Schiffslogbücher zu schreiben. Es wird also vorgesehen, die Sammlung von Anomalie-Berichten nicht nur zu Anomalien zu gruppieren, sondern auch effizient beteiligten Schiffen zuzuordnen, um Schiffslogbücher zu erstellen. Schiffslogbücher können zudem einfach aus weiteren Datenfeldern des AIS-Datenstroms wie Zielhafen und Ladungsart angereichert werden. So können die generierten Schiffslogbücher in forensischer Datenanalyse mit berücksichtigt werden. Schiffslogbücher werden aus allen anfallenden Berichten erhoben, auch aus jenen, die letztendlich keiner Anomalie zugeordnet werden.

Setzt man die AIS-Datenstrombasierte Analyse-Methode für Anomalie-Detektion voraus, erkennt man, dass Detektoren mit verschiedenen komplexen Zusammenhängen umgehen müssen:

Der einfachste Fall sind alle Anomalien, die sich direkt aus dem Datenstrom eines einzelnen Schiffes ergeben. Erkannte Ereignisse bei der Betrachtung der Bewegung eines einzelnen Schiffes resultieren in einer Anomalie. Vertreter dieser Einzelereignis-Anomalien, die im Zuge der IntelliMar-Studie umgesetzt wurden, sind

- Small fast boat proceeding at high speed in open sea

- Drastic (> 30°) change of course at open seas
- The new position / travelled distance after an AIS signal loss does not appear consistent to the previous route (probably a hidden maneuver).

Sind 2 oder eine bestimmte Anzahl von Schiffen an einem Ereignis beteiligt, steigt die Komplexität im Detektor an, da hier gegebenenfalls Ereignisse einzelner Schiffe in Raum und Zeit korreliert betrachtet werden müssen. Die IntelliMar-Studie untersuchte folgende Gruppenereignisse:

- Two or more vessels approach unusually close to each other in international waters
- Double IMO-number in the situation picture
- Double international call sign in the situation picture

Die Komplexität steigt weiter an, wenn an Anomalien ganze Gruppen von Schiffen beteiligt sind, so genannte Cluster-Ereignisse. Bei Cluster-Ereignissen kann gegebenenfalls die Rolle eines einzelnen Schiffs relativ unbedeutend sein. Im Rahmen des Teilvorhabens IntelliMar Services wurde keine Anomalie dieser Gattung untersucht. Ein Beispiel dieses Vertreters ist die Stauererkennung wie zum Beispiel der aus der im Suezkanal querstehenden „Ever Green“ resultierende Stau ab dem 23 März 2021.

Es gibt Anomalie-Szenarien, deren Erkennung nicht durch eine einfache Betrachtung des AIS-Meldungsstroms abgeleitet werden können. Die Komplexität solcher Anomalien übersteigt die Fähigkeiten eines regelbasierten Detektors, sodass hier auf modernere Verfahren gesetzt werden muss. Das Teilvorhaben IntelliMar Services untersuchte als Vertreter dieser Kategorie die Anomalie

- The same two vessels are sailing a route / maneuver (repeatedly within X months) that indicates a delayed drug delivery (considering the timeframe and drift corridor).

Das Kapitel 3.2.7.3 widmet sich dieser Anomalie.

3.2.1.2. Architektur der Datenstromanalyse

Dieses Kapitel stellt die im Rahmen des Teilvorhabens IntelliMar Services entwickelte Architektur für regelbasierte Anomalie-Detektoren vor. Zunächst werden die Bedürfnisse der Anomalie-Detektoren auf Basis der Datenstromanalyse wie oben beschrieben erfasst, um daraus Werkzeuge und Techniken abzuleiten.

Regelbasierte Anomalie-Detektoren gehören seit Beginn der computergestützten Lage-Erfassung zum Repertoire der Datenverarbeitungsmethoden. Regeln lassen sich sehr leicht in Algorithmen fassen und gelten daher in der computergestützten Analyse als elementar. Sie sind und bleiben Bestandteil auch moderner Systeme. Sie gelten als effizient, robust, wartbar, testbar und aufgrund ihrer Vorhersagbarkeit sind sie bei Bedarf audittierbar und zertifizierbar. Somit können sie bis hin zur beweisfesten Strafermittlung eingesetzt werden.

Die zunehmende Leistungssteigerung bei Computern unterstützt das Erarbeiten von komplexeren regelbasierten Methoden unter Berücksichtigung immer weiterer Analyse-Aspekte. Mit zunehmender Komplexität verlieren sie jedoch die Eigenschaft Testbarkeit. Es ist dann nicht mehr möglich, regelbasierte Algorithmen vollständig zu testen. Im Zuge dessen verlieren sie ihre Robustheit und Wartbarkeit. Letztendlich muss damit gerechnet werden, dass sie fehleranfällig und nicht mehr für alle Fälle vorhersagbar werden. Diese Problematik betrifft insbesondere die Entwicklung von regelbasierten Monolithen, also Systemen, deren Komponenten stark voneinander abhängen und nicht lose gekoppelt sind.

Durch Zerlegung kann dem Trend der Nicht-Testbarkeit entgegengewirkt und somit die Wiedererlangung der positiven Eigenschaften erreicht werden. Dieser Trend endet in der Bildung von Microservices, also kleinen, überschaubaren Funktionsblöcken, die wiederum zu komplexen Systemen zusammengestellt werden können. Die Microservices sind unabhängig testbar, ihre Funktion in einem System aus Microservices wird wiederum auf entsprechend höherer Ebene nachgewiesen.

Weitere Gründe für die Zergliederung in Microservices ist die Integrationsfähigkeit in moderne, hardware-agnostische Frameworks (Cloud Computing) und die hieraus resultierende Robustheit gegen Hardware-Ausfall sowie eine Skalierbarkeit zur Lastverteilung. Auch die Nutzung moderner, transparenter Sicherheits-Frameworks für Verschlüsselung und Datenschutz wird so erschlossen.

Die Architektur der IntelliMar Services nutzt daher diese Methoden, um

- ein robustes, wartbares und funktionssicheres System zu realisieren
- sich auf die Kernaufgabe zu konzentrieren (Realisierung der so genannten Business Logik)
- von Skalierungsmöglichkeiten zur Leistungssteigerung zu profitieren
- ein datenschutzkonformes System zu garantieren.

Bei der Realisierung der Business-Logik wurde auf konsequente Umsetzung des Microservice-Schemas geachtet: Regelbasierte Anomalie-Detektoren nutzen oft die gleichen Regeln, jedoch in einem anderen Kontext. Ein Beispiel ist ein Geschwindigkeitsfilter, das bei fast jeder Anomalie-Erkennung sinnvoll eingesetzt werden kann.

Designgrundlage ist daher die Schaffung von einfachen Regel-Komponenten

- die jeweils leicht getestet werden können
- die konfigurierbar bezüglich ihrer Eigenschaften sind (z. B. die Geschwindigkeit eines Geschwindigkeitsfilters)
- die aneinander zu reihen sind, um komplexe Detektoren zu bilden
- die steuerbar im System sind
- die überwachbar sind (Monitoring).

Insgesamt soll die Grundlage geschaffen werden, regelbasiert Anomalie-Detektor-Systeme auf die Ebene moderner Infrastrukturen zu heben und Nachteile klassischer Entwurfsmethoden entgegenzuwirken.

Werkzeuge

Die Möglichkeiten zur Realisierung Microservice-basierter Systeme sind vielfältig, da eine unüberschaubare Anzahl möglicher Werkzeuge in diesem Themenfeld existiert. Dies liegt zum einen an der fortlaufenden Neuentwicklung solcher Werkzeuge und Frameworks sowie an einer ausstehenden Konsolidierung der Tool-Landschaft. Daher wurden im Rahmen des Teilvorhabens IntelliMar Services zunächst Ressourcen zur Auswahl der zu nutzenden Frameworks und Tools aufgewendet.

Programmiersprache Java:

Als Implementierungsgrundlage und Programmiersprache dient Java in einer Version ab mindestens 8. Java gilt als robust, etabliert und zukunftssicher. Es stellt zudem oft die Grundlage der gewählten Werkzeuge dar.

Build-Generator Maven:

Als Werkzeug zum Zusammenstellen von ausführbaren Java-Produkten wurde Maven ab einer Version 3.6 gewählt. Maven ist ebenfalls ein Standard-Werkzeug in der Software-Entwicklung und bietet zudem die Möglichkeit, durch Bereitstellen von eigenen Schablonen, den Maven Archetypes, die Implementierung ähnlicher Funktionseinheiten zu vereinfachen und zu homogenisieren. Hiervon wurde Gebrauch gemacht.

Kernkomponente Apache Kafka:

Apache Kafka ist ein Werkzeug zur Verarbeitung von Datenströmen. Wie in Kapitel 3.1.3.4 bereits dargelegt, wird es ebenfalls zum Bewegen von Daten im Data Lake genutzt. Tatsächlich sind die Einsatzzwecke reichhaltiger. Apache Kafka selbst ist ein verteiltes System, das skalierbar und fehlertolerant ist. Zudem unterstützt es die gängigen Sicherheitsstandards zur Verschlüsselung und zur Zugriffskontrolle. Zur Realisierung der Verteilung setzt es auf Instanzierung mehrerer Serverknoten, den so genannten Brokern, die in einem Netzwerk von Brokern organisiert werden.

Daten werden auf Nachrichtenkanälen, den so genannten Topics, ausgetauscht. Interessenten eines bestimmten Nachrichtentyps abonnieren sich auf ein Topic und erhalten gemäß Konfiguration alle noch nicht empfangenen Nachrichten oder die neuesten Nachrichten. Produziert ein im Kafka-Verbund angeschlossener Prozess Nachrichten, sendet er sie auf das entsprechende Topic und vertraut die Verteilung dem Kafka-Broker-Netzwerk an. Das Kafka-Broker-Netzwerk sieht zudem Maßnahmen zur Rückstaufestigkeit in den Nachrichtenkanälen vor, ein Nachrichtenverlust kann vernachlässigt werden. Solche Systeme bezeichnet man als Publish/Subscribe-Systeme.

Die logischen Nachrichtenkanäle in Kafka können in mehreren realisierten Subkanälen transportiert werden. Die Subkanäle werden Partitionen genannt und ermöglichen so eine Lastverteilung im Realsystem. Von dieser Möglichkeit wurde auch im Rahmen der IntelliMar

Services Gebrauch gemacht. Für die korrekte Nutzung von Partitionen der Topics sei auf die Literatur verwiesen. (Confluent, Inc., 2021) Wichtig für das Verständnis dieses Dokumentes ist folgender Umstand: In Kafka werden Nachrichten stets in Schlüssel-Wertpaaren organisiert. Der Wert-Beitrag einer Nachricht enthält die tatsächlich zu betrachtende Nachricht, der zugeordnete Schlüssel bestimmt die Auswahl der zu nutzenden Partition. Dies heißt, alle Nachrichten, die denselben Schlüssel tragen, werden in ihrer Eingangsreihenfolge in eine Partition geschrieben. Die Empfangsreihenfolge ist nur innerhalb einer Partition konsistent. Sollte ein Empfänger sich mehrerer Partitionen eines Nachrichtentyps bedienen, ist die Reihenfolge nur noch blockweise und nicht mehr vollständig konsistent.

Nutzbar gemacht wird das Kafka- Broker-Netzwerk durch Bereitstellen verschiedener APIs. Datenverarbeitende Prozesse werden nicht Bestandteil des Broker-Netzwerks, sondern nutzen es lediglich. Hierdurch wird eine hohe Stabilität des Broker-Netzwerks gewährleistet. Die Nutzer oder Klienten können sich folgender APIs bedienen:

Kafka Connect:

Kafka Konnektoren sind im Kapitel 3.1.3.4 erläutert. Sie dienen hauptsächlich dazu, Daten aus Fremdsystemen in das Kafka-Netzwerk zu befördern oder wieder an Fremdsysteme abzugeben. Sie stellen demnach Endpunkte für Verarbeitungseinheiten oder –ketten dar. Neben der Möglichkeit, Konnektoren selbst zu implementieren, kann eine reichhaltige Auswahl frei verfügbarer Konnektoren genutzt werden. Eine besondere Stärke von Konnektoren ist deren Nutzbarmachung durch Konfiguration, nicht durch Implementierung. Konnektoren lösen zudem den Dualismus zwischen Bestandsdatenverarbeitung und dem Prozessieren von Live-Daten: Es gibt Konnektoren, die beispielsweise aus einem Dateisystem oder dem HDFS Dateien lesen und in das Kafka-System hinein serialisieren. Live-Daten werden durch einen geeigneten Consumer oder Konnektor in das System gebracht. Ab diesem Zeitpunkt erübrigt sich eine unterschiedliche Betrachtung von Dateien und Live-Datenströmen.

Kafka Consumer und Producer:

Dies entspricht dem klassischen Ansatz eines Publish/Subscribe-Systems zum Nachrichtenaustausch. Es bietet maximale Flexibilität bei der Applikationsentwicklung und stellt gegebenenfalls schnell Kompatibilität zu anderen Publish/Subscribe-Lösungen her, da sich das Entwicklungsmuster solcher Applikationen über Anbieter-Grenzen hinweg stets ähnelt.

Kafka Streams:

Dieses API ermöglicht eine elegante Implementierung im modernen funktionalen Stil. Der Entwicklungsoverhead zum Anschluss an Apache Kafka ist gering und Parallelität zur Leistungserhöhung wird durch Mehrfachinstanzierung eines Kafka-Streams-Prozesses erreicht. Dieses API ermöglicht die maximale Verlagerung auf Anwendungslogik-Aspekte.

Von all diesen APIs wurde bei der Realisierung im Rahmen der IntelliMar Services Gebrauch gemacht; sie wurden entsprechend ihrer Stärken für bestimmte Fälle ausgewählt.

Docker und Docker Compose:

Docker legt den Grundstein für das Erschließen der cloudbasierten Technologien. Es ist ein Containerisierungswerkzeug und stellt Applikationen in abgeschlossenen Ausführungscontainern so bereit, dass sie auf beliebigen Systemen instanziiert und betrieben werden können. Cloud-Technologien stehen im Vorhaben IntelliMar nicht im Vordergrund, eine prinzipielle Nutzung sollte jedoch ermöglicht werden, da das Vorhaben IntelliMar durchaus einem Anspruch an Zukunftsorientiertheit gerecht werden sollte. So wurde ein minimaler Aufwand für Cloud-Technologien berücksichtigt.

Es ist davon auszugehen, dass containerisierte Applikationen in einer cloudbasierten Umgebung profitieren können, wenn sie so ausgelegt werden, dass sie durch simple Mehrfachinstanzierung der Container Leistungssteigerung durch Parallelisierung realisieren. Dem Aspekt der Parallelisierung wird in der Studie zu den IntelliMar Services an mehreren Stellen Rechnung getragen, sowohl seitens der Architektur als auch seitens der Implementierung.

Um im Rahmen der IntelliMar Services den Aufwand der Realisierung einer gesamten cloudbasierten Technologie zu vermeiden, wurde auf das Toolkit Docker Compose gesetzt. Damit können containerisierte Applikationen organisiert, gestartet, gestoppt und überwacht werden. Konfiguriert wird das System durch eine Textdatei, betrieben wird es auf einer Hardware, die lediglich die Docker-Infrastruktur mitbringt. Die konkrete Nutzung wird im Kapitel 3.2.2.7 vorgestellt.

3.2.1.3. Verarbeitungsknoten und Verarbeitungsketten

Basierend auf den beschriebenen Rahmenbedingungen an die Architektur von Anomalie-Detektoren – anwendbar auf regelbasierte und weiterführende Methoden – ergibt sich ein Entwicklungsmuster aus Verarbeitungsknoten und Verarbeitungsketten.

Hintergrund ist die Schaffung eines Vorrats an möglichst einfachen Regelknoten, die durch Verkettung und passende Konfiguration zu einer Regelkette gruppiert werden. Eingangsdaten in die Regelkette sind AIS-Meldungen, der Ausgang einer Kette ist eine Ereignisnachricht, die in das Schiffslogbuch eingetragen wird oder der Darstellungseinheit des IntelliMar-Demonstrators zugeführt wird.

Folgende Abbildung zeigt eine mögliche Realisierung einer Verarbeitungskette:

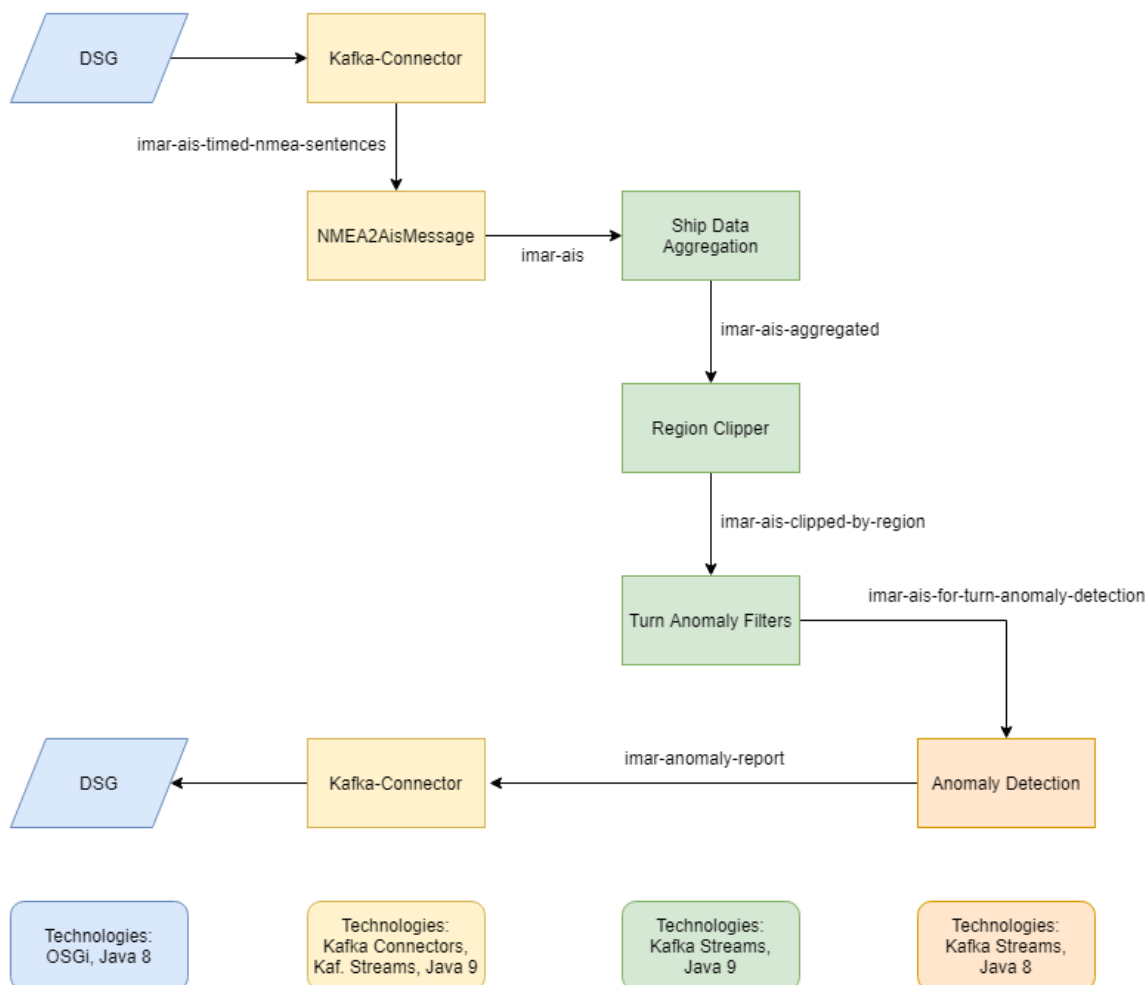


Abbildung 3-8: Verarbeitungskette aus Verarbeitungsknoten

Aufgezeichnet ist der Weg von AIS-Nachrichten durch verschiedene Verarbeitungsknoten hin zur Rückführung eines Ereignisberichts. Eine Kette dieser Art wird für den IntelliMar-Demonstrator verwendet.

Die AIS-NMEA-0183-Meldungen erreichen die Kette über das DSG des Demonstrators (blau). Sie werden von einem speziellen Konnektor in Empfang genommen, zu aggregierten AIS-Meldungen zusammengestellt (gelb) und durch verschiedene Verarbeitungsknoten vorgefiltert (grün), bis sie im Anomalie-Detektionsknoten intensiv analysiert werden (orange). Ausgabe des Detektors ist ein Ereignisbericht, der von einem weiteren Konnektor (gelb) wieder an das DSG übermittelt wird (blau).

Die Pfeile symbolisieren Kafka-Topics, die zum Nachrichtentransport genutzt werden. Hierbei ist die Methode der Verkettung der Verarbeitungsknoten zu erkennen: Die Ausgabe des Vorgängerknotens bildet die Eingabe des Folgeknotens.

Im Rahmen des Teilvorhabens IntelliMar Services sind zahlreiche dieser Verarbeitungsknoten entstanden, sowohl einfache Filter wie auch komplexere Analyse-

Knoten. Die Komplexität der Analyse-Knoten wird dabei durch die Vorfilterung auf das notwendige Maß reduziert; eine beliebige Reduzierung ist nicht möglich, da ein Analyse-Knoten unter Umständen komplexere Zustände für verschiedene Objekte vorhalten muss. Im Kaptiel 3.2.2 werden die realisierten Verarbeitungsknoten vorgestellt.

Ein einzelner Verarbeitungsknoten ist nach folgendem Muster aufgebaut:

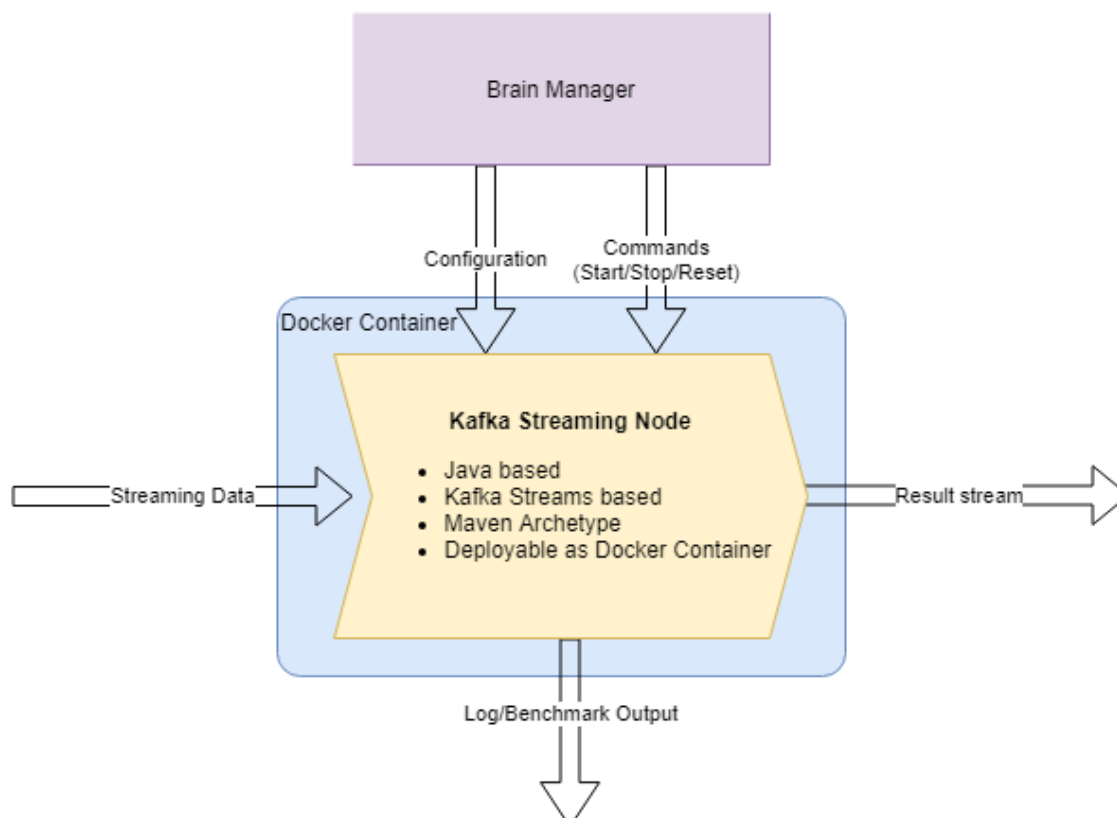


Abbildung 3-9: Innere Architektur eines Verarbeitungsknotens

Der Verarbeitungsknoten (gelb) ist Java-basiert implementiert. Er nutzt das Kafka-Streams-API und kann als einzelner Docker-Container ausgerollt werden. Zur Unterstützung der Gleichförmigkeit für Verarbeitungsknoten steht eine Realisierungsschablone als Maven Archetype bereit. Dieses Konstrukt wird im IntelliMar-Services-Umfeld auch Kafka Streaming Node oder nur Streaming Node genannt.

Ein Kafka Streaming Node erhält seine Eingangsdaten aus einem Kafka-Topic und schreibt seine Ergebnisse auf ein Ergebnis-Topic. Kernstück eines Kafka Streaming Nodes ist die Realisierung einer simplen Funktion nach dem Schema:

$$m' = f(m)$$

wobei m' die Ergebnismnachricht und m die Eingangsmnachricht ist. Die Ergebnismnachricht m' darf auch leer sein. Diese Eigenschaft wird für das Filtern benötigt. Auch der finale Analyse-

Knoten macht hiervon überwiegend Gebrauch, da hier die Eingangsnachricht m zwar noch gefilterten AIS-Nachrichten entspricht, die Ergebnissnachricht m' jedoch ein Ergebnis- oder Anomalie-Bericht ist, dessen Häufigkeit im Verhältnis zu eingehenden AIS-Nachrichten gering ist.

Eine weitere Eigenschaft jedes Verarbeitungsknotens ist das Vorhandensein eines Log- und Monitoring-Ausgangs sowie eines Befehls- und Status-Kanals. Alle Kanäle sind als Kafka-Topics ausgelegt. Zudem enthält jeder Knoten eine generische Funktionalität zum Auswerten von Kommandos und Senden von Statusinformationen sowie zum Bereitstellen einfacher Statistiken für das Benchmarking und zum Loggen weiterer Informationen.

Diese Maßnahmen garantieren eine Konzentration auf die Implementierung der Business-Logik, zumeist durch Realisierung einer Funktion. Die Funktion kann unabhängig von Kafka-Kenntnissen entwickelt werden.

In der Abbildung 3-9 violett dargestellt befindet sich ein Verwaltungssystem für Verarbeitungsknoten (Brain Manager). Es ist zentral ausgelegt und selbst als Verarbeitungsknoten implementiert. Die Architektur und Funktionsweise des Brain Managers wird im Kapitel Management von Verarbeitungsketten erläutert.

Folgende Abbildung zeigt eine Verarbeitungskette nun im Kontext mit Konfigurations- und Logging-Funktionalität:

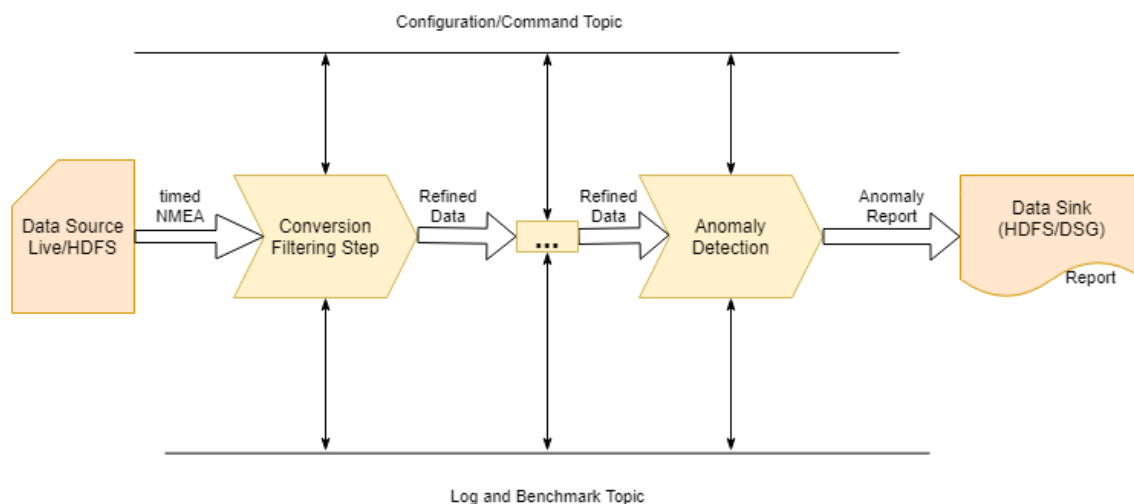


Abbildung 3-10: Verarbeitungskette mit Steuer- und Log-Integration

Man erkennt, dass alle Verarbeitungsknoten der Kette an einem gemeinsamen Strang (Kafka-Topic) zur Kommandierung und für Statusmeldung sowie an einem Strang für Log- und Benchmark-Ergebnisse hängen.

Formal besteht für eine Verarbeitungskette folgender Zusammenhang:

$$r = f_n(f_{n-1}(\dots f_1(m)))$$

mit

- r als Ergebnisbericht,
- f_n als Anomalie-Erkennungsknoten,
- f_{n-1} und weitere als vorgelagerte Filterknoten und schließlich
- f_1 als Eingangsknoten zur Verarbeitung der
- Eingangsnachricht m .

3.2.1.4. Auslegungsmodelle für Verarbeitungsketten

Die bisher dargelegte Architektur ermöglicht es nach den Designvorgaben eine Erkennungskette und somit einen Anomalie-Detektor zu realisieren. Tatsächlich soll die Architektur nicht auf eine Verarbeitungskette beschränkt sein, sondern die Koexistenz sowohl mehrerer verschiedener als auch mehrerer gleicher Verarbeitungsketten ermöglichen.

Verschiedene Verarbeitungsketten realisieren hierbei Anomalie-Detektoren unterschiedlichen Anomalie-Typs. Da 5 grundsätzlich verschiedene Detektoren implementiert wurden, enthält der IntelliMar-Demonstrator mindestens 5 Ketten. (*Hinweis:* Die Anomalien vom Typ „Double IMO-number in the situation picture“ und „Double international call sign in the situation picture“ sind ausreichend deckungsgleich, sodass sie innerhalb eines Anomalie-Detektor-Knotens implementiert wurden.)

Zur Mehrfachauslegung derselben Verarbeitungskette kommt es in folgenden Fällen:

- Skalierbarkeit: Die Leistung des Detektors wird erhöht, indem er mehrfach instanziiert wird.
- Verarbeitung verschiedener Quellen: Der IntelliMar-Demonstrator ist nicht auf die Verarbeitung einer einzigen AIS-Nachrichtenquelle beschränkt. Im Mindestumfang des Demonstrators sind die Verarbeitung von Live-AIS-Daten aus der Quelle AISHub und die Analyse in Echtzeit oder Über-Echtzeit wiedergegebener AIS-Daten aus vorgefertigten Szenarien vorgesehen.

Aus Sicht von IntelliMar Services ergibt sich folgende Gesamtarchitektur:

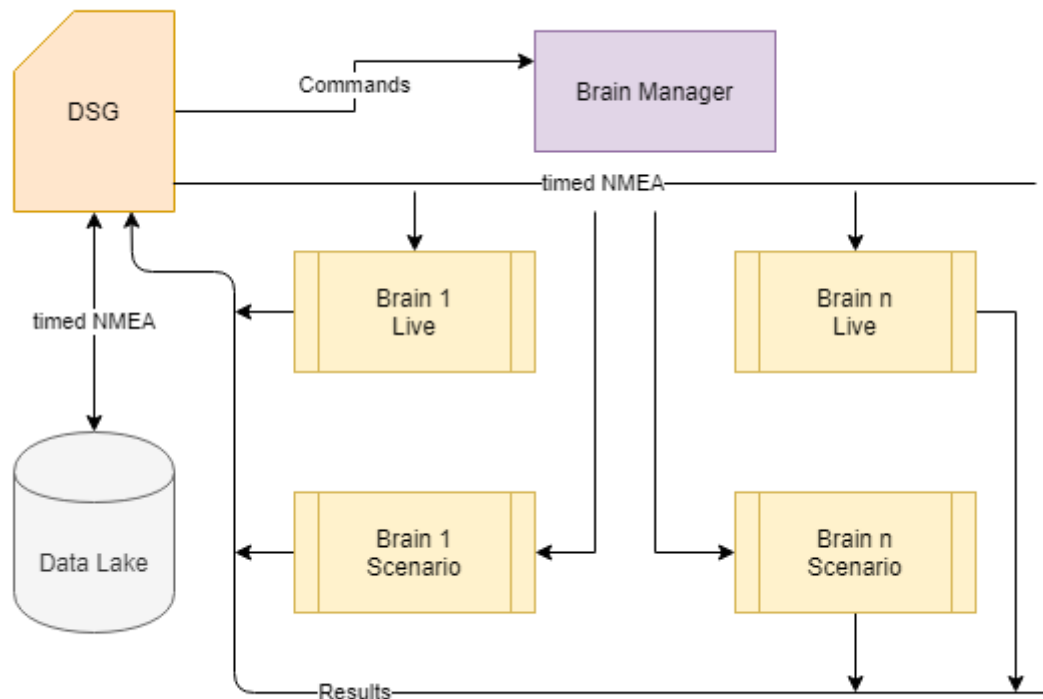


Abbildung 3-11: Gesamtarchitektur IntelliMar Services

Datenquelle und –senke des IntelliMar-Demonstrators ist stets das DSG (orange). Über das DSG werden AIS-Nachrichten mit Zeitannotation empfangen, die Ereignisberichte versendet und Befehle zur Steuerung der Verarbeitungsketten entgegengenommen. Das DSG unterhält auch eine Verbindung zum Data Lake (grau), um Zugriff auf z. B. hinterlegte historische AIS-Daten zu erhalten.

Die einzelnen Verarbeitungsketten (gelb) abonnieren sich auf die für sie relevanten Datenströme, die vom DSG bereitgestellt werden. Die Ereignisberichte werden auf einem gemeinsamen Kanal dem DSG zur Verteilung im DSG-Netzwerk zur Verfügung gestellt. Nicht symbolisiert ist das direkte Zuliefern der Ergebnisberichte in eine Datenbank des Data Lakes, um dort Schiffslogbücher zu führen.

Der Brain Manager (violett) empfängt Befehle vom DSG und setzt sie in Befehle zur Steuerung der einzelnen Verarbeitungsketten um. Das DSG wiederum dient in der Befehlskette lediglich als Übermittler. Erteilt wird ein Befehl durch eine im DSG-Netzwerk angeschlossene Stelle, wie zum Beispiel eine Benutzeroberfläche zur Konfiguration. Die Kommando-Verbindung zwischen DSG und Brain Manager unterhält ebenfalls einen Rückkanal zur Statusübermittlung der Verarbeitungsketten.

Aus Gründen der Übersichtlichkeit ist in der Abbildung das Sammeln von Log-Ausgaben und Benchmarking-Ergebnissen aus den Verarbeitungsketten nicht dargestellt.

3.2.1.5. Management von Verarbeitungsketten

Die Erkennung einer einzelnen Anomalie ist Aufgabe eines Anomalie-Detektors in Form einer Verarbeitungskette von Kafka Streaming Nodes. Diese werden im IntelliMar-Umfeld „Brains“ genannt. Gesteuert und konfiguriert werden die Brains und ihre Komponenten - den einzelnen Kafka Streaming Nodes durch den Brain Manager.

Die Kommunikation zwischen Streaming Nodes und Brain Manager erfolgt über Kafka Topics (Abbildung 3-12).

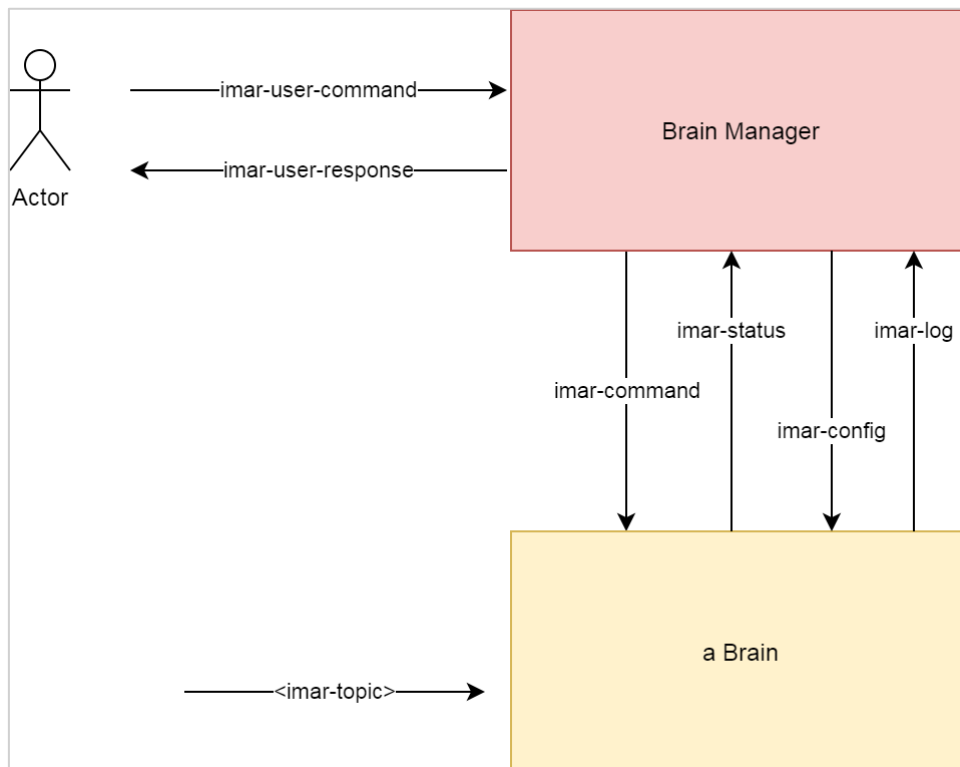


Abbildung 3-12: Kommunikation zwischen Streaming Nodes und Brain Manager

Empfangen wird ein Kommando dabei immer von allen Streaming Nodes. Verarbeitet werden jedoch nur an die jeweilige Streaming Node (<Brain Name>:<Node Name>) oder das jeweilige Brain (<Brain Name>:*) adressierte Befehle.

Auch der Brain Manager selbst kommuniziert über Kafka mit dem Brain Manager HMI (Kapitel 3.2.2.8 Das Brain Manager HMI) oder dem Kommandozeilentool Brain Manager CLI (Kapitel 3.2.2.9 Das Brain Manager CLI Kommandozeilentool) und erhält so Befehle von außerhalb.

Die Streaming Nodes selbst sind zustandslos, lediglich Brain Name und Node Name sind fest konfiguriert. Sie starten zunächst im angehaltenen Zustand und warten auf ihre Konfiguration durch den Brain Manager. Dafür teilen sie ihren Zustand sowie unterstützte Parameter über das imar-status Topic mit. Der Brain Manger hat die Aufgabe, die Konfiguration zu persistieren und dem jeweiligen Streaming Node über das imar-config Topic wieder zur Verfügung zu stellen. Anschließend wird der Brain Manager das Brain

entsprechend dem vorherigen Zustand über das imar-command Topic starten (siehe Abbildung 3-13: Interaktion zwischen Brain Manager und Brain (li) und Zustandsübergänge der Brains (re)).

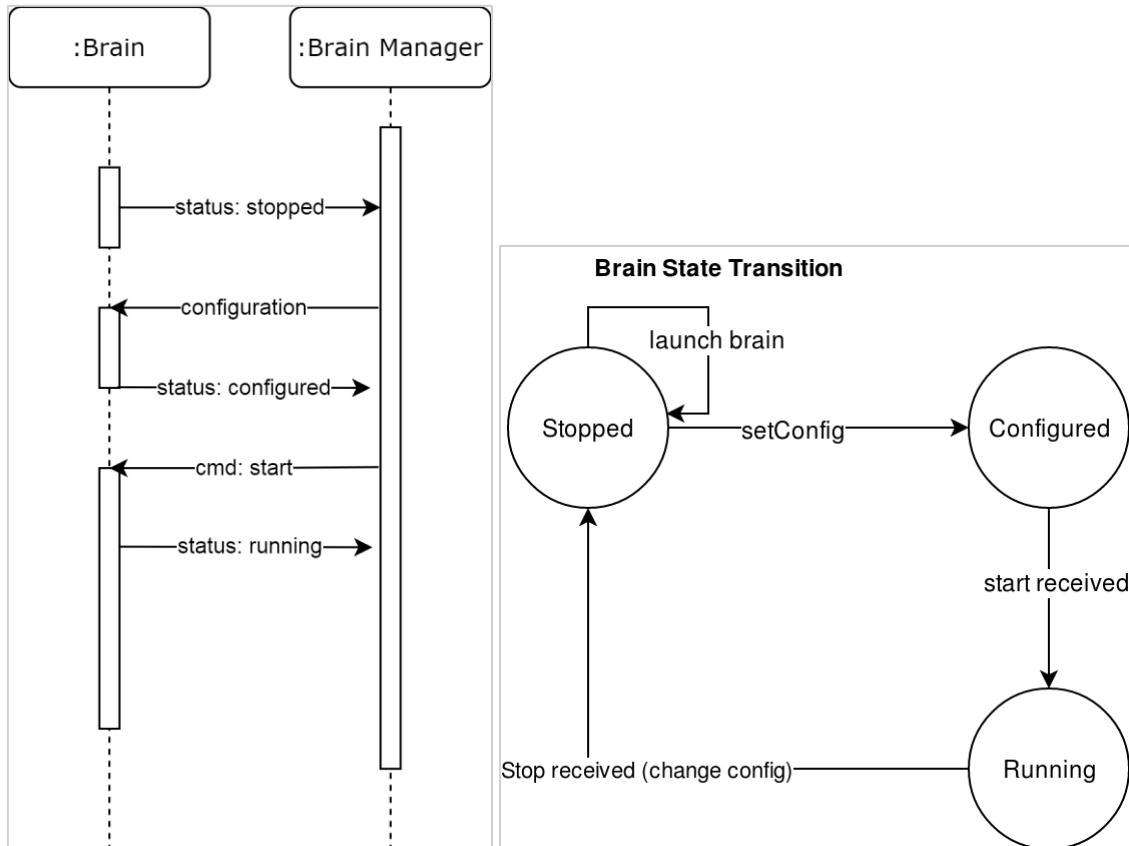


Abbildung 3-13: Interaktion zwischen Brain Manager und Brain (li) und Zustandsübergänge der Brains (re)

Über das imar-command Topic werden folgende Befehle von jeder Streaming Node unterstützt:

Befehl	Effekt
START	Verarbeitung starten
STOP	Verarbeiten anhalten
RESET	Statistiken auf 0 und Konfiguration auf Anfangswerte zurücksetzen
ENABLE_FILTER	Normale Funktion
DISABLE_FILTER	Im Falle eines Filter: alle Daten ungefiltert durchleiten
READ_STATISTICS	Statistiken auf dem imar-log Topic ausgeben

Tabelle 3-1: Befehle zur Steuerung der Streaming Nodes

Das imar-log Topic dient zur Übertragung von Statistiken zur Anzeige im Brain Manager HMI.

Über das imar-user-command Topic werden folgende Befehle vom Brain Manager unterstützt:

Befehl	Effekt
LIST	Alle verfügbaren Ketten auf dem imar-user-

	response Topic ausgeben
START <Streaming Node Adresse>	Befehl an Streaming Node(s) weiterleiten
STOP <Streaming Node Adresse>	Befehl an Streaming Node(s) weiterleiten
SET_CONFIG <Streaming Node Adresse>	Angegebene Streaming Node(s) konfigurieren
GET_CONFIG <Streaming Node Adresse>	Konfiguration der angegebenen Streaming Node auf imar-user-response ausgeben
READ_STATISTICS <Streaming Node Adresse>	Statistiken der angegebenen Streaming Node abfragen und auf imar-user-response ausgeben
ENABLE_FILTER <Streaming Node Adresse>	Befehl an Streaming Node(s) weiterleiten
DISABLE_FILTER <Streaming Node Adresse>	Befehl an Streaming Node(s) weiterleiten
RESET <Streaming Node Adresse>	Befehl an Streaming Node(s) weiterleiten
CLEAR <Streaming Node Adresse>	Persistierte Konfiguration für Streaming Node(s) löschen

Tabelle 3-2: Brain Manager Benutzerkommandos

3.2.1.6. Performance-Steigerung

Die Architektur der IntelliMar Services bietet den Kafka Streaming Nodes und damit den Verarbeitungsketten Raum zur Performance-Verbesserung, sobald es notwendig wird. Erreicht wird dies durch Mehrfach-Instanziierung von Kafka Streaming Nodes. Wie in **Kernkomponente Apache Kafka** dargelegt, unterstützt Kafka diese Art der Parallelisierung.

Wird Parallelisierung im Kafka-Umfeld gewünscht, erfordert dies die Auslegung der Kafka-Topics in mehreren Partitionen. Partitionen werden den Kafka Streaming Nodes gleichverteilt zugeordnet; die an einem Streaming Node anliegende Datenlast reduziert sich auf

$$l = \frac{m}{n}$$

wobei l die Last in Nachrichten m je Streaming Node ist. Existiert nur eine Instanz n eines Streaming Nodes, so wird er auch weiterhin die gesamte Nachrichtenlast tragen müssen. Existieren idealerweise so viele Partitionen wie instanziierte Streaming Nodes, teilt sich die Last wie oben beschrieben. Gibt es mehr Partitionen als parallele Streaming Nodes, werden mehrere Partitionen auf je einen Streaming Node gelegt, sodass obige Formel im Falle eines teilbaren Verhältnisses von Streaming Nodes zu Partitionen erfüllt bleibt. Ist das Verhältnis nicht teilbar, kommt es zu einer unterschiedlichen jedoch gleichmäßigen Lastverteilung auf die Streaming Nodes. Ist die Anzahl der instanziierten Streaming Nodes höher als die Anzahl der verfügbaren Partitionen, kommt es zu nicht genutzten Streaming Nodes und die Last l berechnet sich zu

$$l_p = \frac{m}{p}$$

mit p als Anzahl der zur Verfügung stehenden Partitionen. Da in diesem Fall gilt:

$$p < n \Rightarrow l_p > l$$

Die Last je aktivem Knoten steigt an, ruhende Knoten führen zu keiner Leistungssteigerung. Dieser Umstand sollte bei der Auslegung der Partitionen und parallelen Streaming-Node-Instanzen berücksichtigt werden.

Ein praktisches Beispiel im IntelliMar-Demonstrator ist die Konvertierung von NMEA-0183-Sätzen in das AIS-Nachrichtenformat, welches in den Folgeschritten genutzt wird. Jede Verarbeitungskette nutzt diesen vorgeschalteten Konvertierungsknoten. Im Fall einer Über-Echtzeitverarbeitung von archivierten AIS-Daten zeigt sich dieser Konvertierungsknoten als Flaschenhals, hier ist eine parallele Auslegung der Konverter indiziert. Realisiert wird dies mittels eines speziell hierfür implementierten Kafka Streaming Nodes, dem AIS-Message-Partitioner oder kurz Partitioner. Aufgabe des Partitioners ist es, eingehende NMEA-0183-Sätze vor der eigentlichen Konvertierung effizient zu analysieren und bezüglich eines Schlüssels implizit einer bestimmten Partition zuzuweisen. Als Schlüssel wurde die MMSI gewählt. Das heißt, jede Nachricht einer bestimmten MMSI wird stets der gleichen Partition zugeordnet und dorthin verteilt. Da die Vergabe der MMSI als gleichverteilt bezüglich des Modulo-Verfahrens eines niedrigen ganzen Teilers bezogen auf die Integer-Darstellung der MMSI als gleichverteilt angesehen werden kann, ergibt sich hierdurch eine homogene Lastverteilung auf die beteiligten Partitionen.

Folgende Abbildung zeigt die Auslegung in n Partitionen.

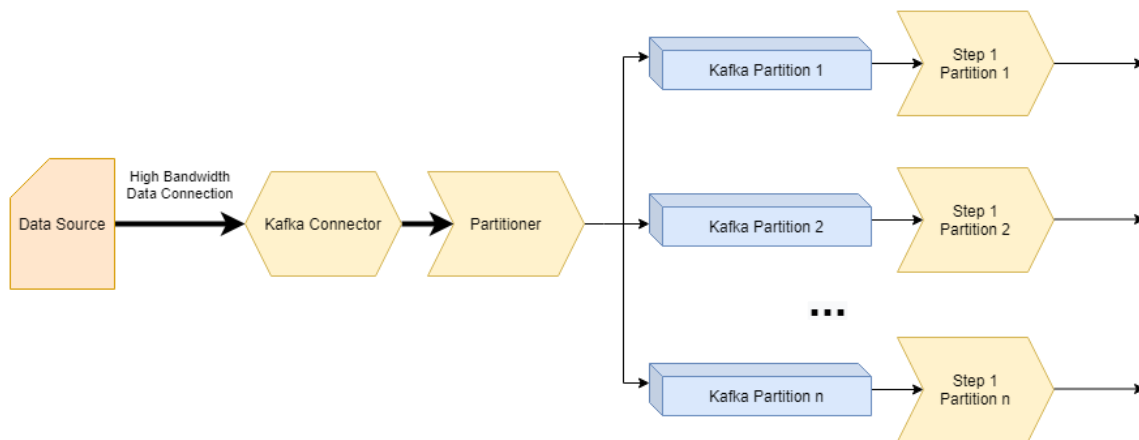


Abbildung 3-14: Der Partitioner verteilt die Datenlast auf Partitionen

Aus einer Datenquelle mit hoher Datenrate empfängt ein Kafka Connector zunächst die unkonvertierten NMEA-0183-Daten. Der Partitioner ermittelt aus den NMEA-0183-Sätzen effizient die MMSI und legt sie als Schlüssel der Kafka-Nachricht fest, was zu einer Zuteilung zu einer nach dem Modulo-Verfahren bestimmten Partition führt. Der eigentliche Konverter „Step 1“ ist hier n -fach instanziiert. Dies führt zu einer Lastreduktion auf $1/n$. Es konnte für den IntelliMar-Demonstrator nachgewiesen werden, dass eine Auslegung mit 4 Partitionen die Prozessorlast einer einzelnen Konverter-Instanz auf unter 100 % reduziert; es kommt zu einer Sättigung. Eine weitere Partitionierung ist zwar möglich, jedoch in Bezug auf Performance-Steigerung in diesem Schritt nicht zielführend. Die tatsächlich zu wählende Anzahl an Partitionen hängt von der Eingangsdatenrate sowie der Rechenleistung eines

einzelnen Streaming Nodes in einer konkreten Umgebung ab. Diesen Parameter gilt es daher manuell zu ermitteln oder durch geeignetes Logging zu bestimmen. Eine Anpassung zur Laufzeit ist für Cloud-Installationen von Kafka in der Confluent-Distribution möglich, jedoch nicht im Rahmen des IntelliMar-Demonstrators umgesetzt worden.

Diese Art der Performance-Steigerung bringt jedoch Nachteile mit sich: Es ist zwar möglich, nach einer Lastverteilung die Ergebnisse wieder in einer Partition zu vereinigen, jedoch geht hierbei die strenge Ordnung der Eingangsnachrichten bezüglich Zeit in einem gewissen Maße verloren. Nur noch Nachrichten, die derselben MMSI zugeordnet sind, also aus einer bestimmten Partition stammen, erfüllen nach wie vor das Kriterium der zeitlichen Ordnung. Für Nachrichten verschiedener MMSIs stimmt dies unter Umständen nicht mehr. Dieser Umstand muss gegebenenfalls in einem Anomalie-Detektor-Knoten kompensiert werden. So erhöht sich die Komplexität ungewollt; dies sollte durch die Wahl der Architektur eigentlich vermieden werden. In den meisten Fällen reicht bei genauer Analyse hingegen die eingeschränkte zeitliche Abfolge der Nachrichten ohne Anpassung des Detektors aus.

Werden Anomalien vom Typ Einzelereignisse (s. Anomalien im AIS-Datenstrom) detektiert, stellt eine Parallelisierung anhand der MMSI keine Einschränkung dar. Die Detektoren ermitteln eine Anomalie für jedes Schiff – also MMSI – unabhängig. In diesem Falle kann die Parallelisierung entweder auf der gesamten Kette durchgeführt werden oder nach der Konvertierung bei ausreichender Rechenleistung der beteiligten Streaming Nodes in eine Partition zurückgeführt werden. Das Ergebnis der Detektion ist stets das gleiche.

Folgende Darstellung zeigt den Effekt durch die Partitionierung für 3 Objekte:

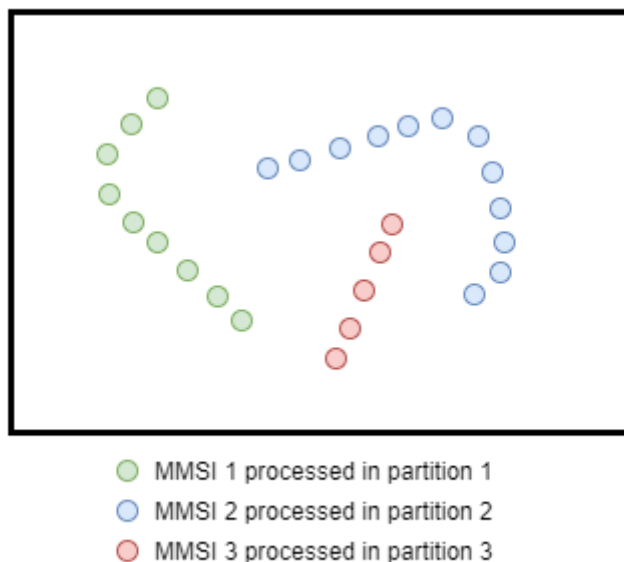


Abbildung 3-15: Partitionierung nach MMSIs

Sowohl für die Objekte der grün markierten Nachrichten-Spur als auch für jene der blauen Nachrichten werden Abbiege-Manöver erkannt; es sind Einzelereignisse, der Bezug zu je einer MMSI ist gegeben.

Sobald mehrere Objekte an einem Ereignis beteiligt sind, treten ungewollte Effekte auf. Unter der Vorgabe, dass die Verarbeitungsketten jeweils vollständig parallel ausgelegt sind, also Nachrichten einer bestimmten MMSI durch genau eine dieser Ketten laufen, kann folgendes Szenario eintreten:

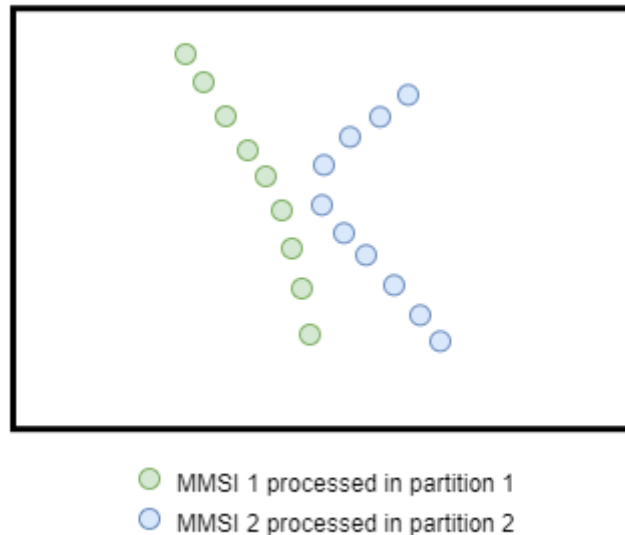


Abbildung 3-16: Partitionierung nach MMSI verhindert Erkennung einer Schiffsbegegnung

Die grüne Nachrichtenspur wird von Kette 1 verfolgt, die blaue Spur von Kette 2. Kette 1 und 2 arbeiten unabhängig voneinander; sie tauschen keine Daten aus. Im Ergebnis werden sie die Anomalie der Annäherung der 2 Schiffe nicht feststellen können. Dies funktioniert nur, wenn beide Spuren in ein und derselben Kette bearbeitet werden.

Ist die Nutzung einer parallelen Auslegung solcher Detektoren aus Performance-Gründen notwendig, muss zu Beginn der Ketten ein anderer Partitioner implementiert werden: In diesem Fall bietet sich eine Partitionierung im Raum an. Die NMEA-0183-Sätze müssen zunächst effizient nach ihrer Position analysiert werden. Danach werden sie Partitionen gleichen Raums zugeordnet. Die Bestimmung kann z. B. mittels Bildung des Geohashes vorgenommen werden.

Hieraus ergibt sich dann beispielsweise folgende Darstellung für den Annäherungsdetektor:

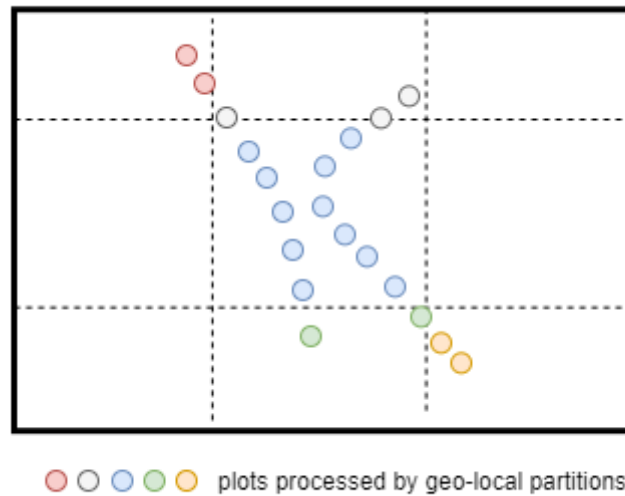


Abbildung 3-17: Partitionierung je Gebiet: Schiffsbegegnung wird erkannt.

Die 2 Schiffe werden dem gleichen örtlichen Bereich zugeordnet und deswegen der gleichen Partition zugeführt. Die Verarbeitungskette hinter der Partition sieht beide beteiligten Schiffe und kann so zuverlässig die Annäherung erkennen. Kritisch ist jedoch das Bilden scharfer Ortsgrenzen der Gebiete. Die vollständige Spur eines Schiffes wird bei dieser Art der Partitionierung von verschiedenen Verarbeitungsketten bearbeitet. Auch das Bearbeiten von Cluster-Anomalien kann mit dieser Partitionierung nicht vollständig abgedeckt werden:

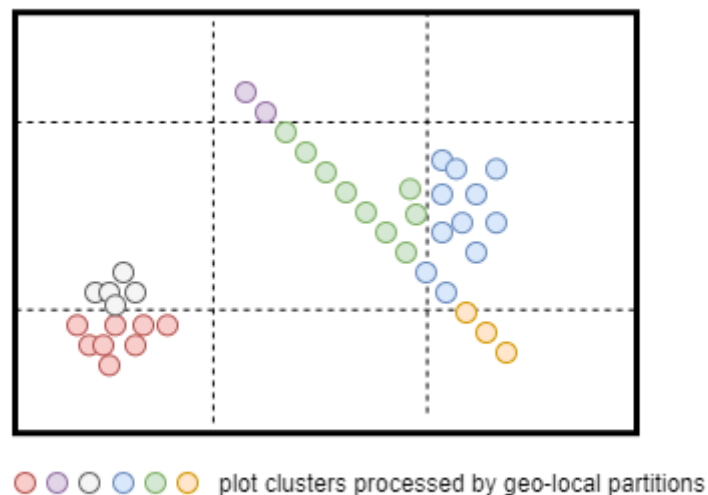


Abbildung 3-18: Cluster-Teilung bei Partitionierung nach Gebieten

Es ist nicht sichergestellt, dass ein Cluster vollständig in einem Geobereich zu liegen kommt, somit wird das Cluster von mehreren Verarbeitungsketten erkannt oder nicht entdeckt.

Gegebenenfalls wandert das Cluster, was zu fließenden Übergängen zwischen Verarbeitungsketten führt.

Diese Probleme können durch folgenden Ansatz gelöst werden: Sämtliche Verarbeitungsketten für Ereignisse mit räumlichem Bezug melden ihre Ergebnisse an eine einfach ausgelegte Instanz, die eine abschließende Beurteilung der Lage vornimmt. Dies scheint dem Parallelisierungsansatz zu widersprechen, jedoch macht man sich tatsächlich einen Effekt der Anomalie-Detektoren zu Nutze: Sie reduzieren die eigentliche Meldungslast aus AIS-Meldungen auf eine handhabbare Menge an Ereignisberichten. Es kann davon ausgegangen werden, dass diese geringe Last von einer finalen Einzelinstanz bewältigt werden kann, insbesondere unter der Vorgabe, dass bestimmte Anomalie-Detektoren nur auf einem abgegrenzten Gebiet eingesetzt werden. Im Beispiel IntelliMar ist dies das Gebiet der deutschen Küste.

Das Verfahren der örtlichen Partitionierung wurde im Rahmen der IntelliMar-Studie nicht umgesetzt.

Abschließend erfolgt eine Performance-Betrachtung des IntelliMar-Demonstrators ohne Nutzung der Partitionierung und damit unter Inkaufnahme des Flaschenhalses zu Beginn der Verarbeitungsketten. Diese Auslegung wurde gewählt, um den gesamten Anomalie-Detektor-Umfang des Demonstrators mit in die Performance-Betrachtung einzubeziehen.

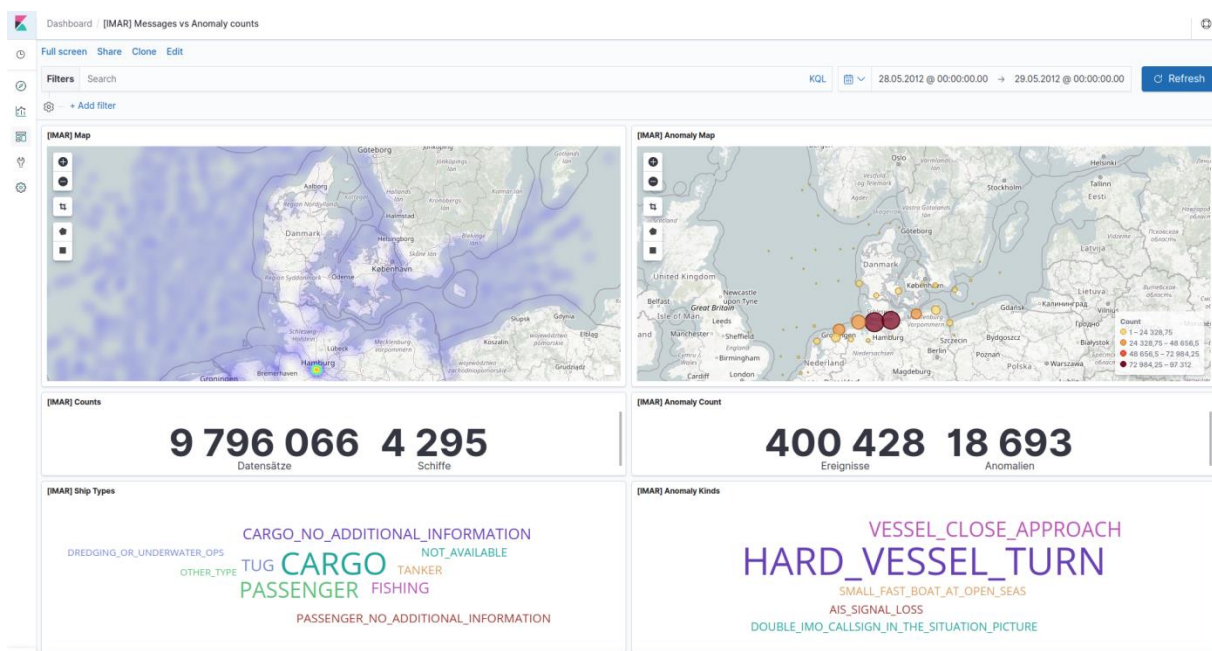


Abbildung 3-19: Anomalie-Scan des WSV-Datensatzes vom 28. Mai 2012

Zugrunde liegt ein historischer AIS-Datensatz des WSV, hier die gesammelten Daten des 28. Mai 2012. Es ist der selbe Datensatz, der auch schon im Kapitel 3.1.2.2 vorgestellt wurde. Ausgewertet wurden knapp 10 Millionen Datensätze, die 4295 Schiffen zugeordnet werden konnten. Links unten ist eine grobe Darstellung der Schiffsklassenzuordnung zu sehen. Überwiegend wurden Fracht- und Passagierschiffe erfasst. Rechts oben ist die Verteilung der detektierten Ereignisse zu sehen. Die Häufung im Bereich des Hamburger Hafens lässt

sich auf zahlreiche Wendemanöver und Schiffsbegegnungen im Hafengebiete zurückführen. Zu Lastbestimmungszwecken wurde absichtlich auf eine Filterung dieser Gebiete für die entsprechenden Detektoren verzichtet. So lässt sich ebenfalls die Darstellung der grob quantifizierten Verteilung der Anomalien mit der Häufung für Wendemanöver (Hard Vessel Turn) und Begegnungen (Vessel Close Approach) erklären.

Entdeckt wurden im Datensatz ca. 400 Tausend Einzelereignisse, die 18693 verschiedenen Anomalien zugeordnet werden konnten.

Die Verarbeitung des Datensatzes erfolgte in ca. 15 Minuten, also etwa 0,01 Tagen. Hieraus ergibt sich eine Beschleunigung gegenüber einer Echtzeit-Verarbeitung um den Faktor 100. Im Gegenschluss heißt das, dass ein auf eine Partition ausgelegtes Erkennungssystem wie dieses ca. 400 Tausend Schiffe (4295 x 100) gleichzeitig verfolgen kann, um an seine Lastgrenze zu stoßen. Im Einsatzgebiet, das für die IntelliMar-Studie analysiert wird, ist der Wert lediglich theoretischer Natur. Es wurde erkannt, dass eine einfache Auslegung der Anomalie-Detektionsketten für die Belange des IntelliMar-Demonstrators ausreichend ist.

3.2.2. Regelbasierte Detektoren

3.2.2.1. Allgemeine Komponenten und ihr Zusammenspiel

Der Weg von den einzelnen rohen Schiffsdaten hin zu einer gefunden und kompletten Anomalie ist ein sehr langer, der zwar schnell bewältigt wird aber auch aus vielen kleinen Schritten besteht. Werden die AIS-Daten in ihrer Rohform in das regelbasierte Anomalieerkennungssystem eingeführt, so müssen sie erstmal auf das vorhandene System angepasst werden. Diesen Schritt nennt man die Datenvorprozessierung. Hier müssen die Daten vorsortiert werden und vor allem auf das interne Format angepasst werden. Das übernehmen mehrere Streaming-Nodes, die miteinander verkettet werden. Sind die Daten bereit und korrekt vorbereitet, so folgt der nächste Schritt: Die Datenverarbeitung und die eigentliche Anomalieerkennung. Dies wird mithilfe verschiedenster Streaming-Node-Filter und Anomaliedetektoren gemacht, die ebenfalls in Reihe miteinander verkettet sind und für jede Anomalie unterschiedlich sind. Da dies der Hauptkern des kompletten Systems ist, wird dazu ein eigenes Kapitel beschrieben, welches im Anschluss an dieses steht. Wurde eine Anomalie gefunden, so muss diese weitergeleitet oder gespeichert werden. Diesen abschließenden Schritt nennt man Datenlieferung. Dabei werden die Anomalien entweder auf ein externes Interface gemappt, mit dem die nachfolgenden Partner arbeiten können oder sie werden im Data Lake gespeichert. Für jede dieser Optionen sind wiederum weitere Streaming-Nodes von Nöten. Wie all diese Komponenten funktionieren, miteinander verknüpft oder benutzt werden, wird im Folgenden gezeigt; dazu werden zunächst die wichtigsten Begriffe, Definitionen, interne Schemata und Datenquellen erläutert.

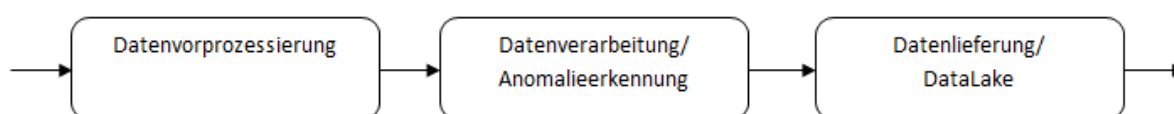


Abbildung 3-20: Schematische Darstellung des Datenflusses

Die hier dargelegten Aktivitäten und Ergebnisse sind im Rahmen des Arbeitspaketes 1510 entstanden und decken dieses nach Vorhabenbeschreibung, Kapitel 4.1.1, Arbeitspaketbeschreibung AP-Nr. 1510 vollständig ab. [RD-01]

3.2.2.2. Grundlagen AIS-Daten und internes AVRO-Schema AisMessage

Die zu prozessierenden AIS-Daten werden als **NMEA-Satz** empfangen. Diesem wird ein **Zeitstempel** in einem Format vorangestellt, welches zuvor mit den Studienpartnern abgesprochen wurde. Dieser Zeitstempel ist nötig um die Schiffsdaten zeitlich zuordnen zu können. Die Schiffsdaten mit all ihren Information stecken in den NMEA-Sätzen. NMEA-Sätze basieren auf dem NMEA-0183-Standard, welcher für die Kommunikation zwischen Navigationsgeräten fungiert und von der National Marine Electronics Association (NMEA) definiert wurde. Außerdem wird eine eindeutige Kennung, ähnlich der MMSI, den Schiffen zugewiesen, die über alle Studienpartner hinweg genutzt werden soll, diese ist die **EntityId** vom Typ UUID. Somit werden die Schiffsdaten folgendermaßen im System ankommen:

2020-05-29T09:17:25+00:00 **8c988418-0000-037f-0000-01725fa6e668**

!AIVDM,1,1,,A,B39R4600lp9G;IWbL:?NGwU1I86<,0*16

Wie genau diese NMEA-Sätze dekodiert werden, kann unter <http://www.nmea.de> (NMEA, 2021) nachgelesen werden.

NMEA-Datensätze können in dieser Form nicht effizient verarbeitet werden, sie müssen in ein für die Informationsverarbeitung handhabbares Format konvertiert werden. Hierzu gibt es viele Möglichkeiten. Neben proprietären Ansätzen bieten sich jedoch auch standardisierte Formate an, deren Unterstützung in der modernen Applikationsentwicklungswelt deutlich breiter ist. Im Rahmen von IntelliMar wurde hier das modellierbare und versionssichere Format AVRO gewählt. Es werden im Folgenden Grundlagen der AIS-Daten und des intern genutzten AVRO-Schema AisMessage erläutert, welche für das Verständnis von hoher Wichtigkeit sind.

Der erste Schritt der Datenvereinnahme wird daher die Konvertierung der NMEA-Sätze in das AVRO-Format sein. Es gibt verschiedene Typen von AIS-Daten, nämlich zum einen Bewegungsdaten (z.B. Position, Kurs, etc.) und zum anderen statische Daten (z.B. Schiffslänge, IMO-Nummer, etc.). Die einzige Information, die in jeder AIS-Nachricht enthalten ist, ist die MMSI-Nummer. Diese Nummer weist die Informationen eindeutig diesem Schiff (eigentlich AIS-Transponder) zu. Außerdem wird noch zwischen Schiffen mit Klasse-A-Transpondern und mit Klasse-B-Transpondern unterschieden. Die Zuweisung der Klasse erfolgt aufgrund der Größe und der Art der Schiffe. Die Zusammenführung von Bewegungsdaten und statischen Daten erfolgt in der Datenvorprozessierung und zwar im Aggregator.

Da für die regelbasierte Anomalieerkennung bestimmte Informationen wichtiger sind als andere und manche sogar nicht von Bedeutung sind, gibt es ein internes AVRO-Schema mit Namen AisMessage, in dem die benötigten Daten zusammengefasst sind. So entsteht ein Mapping zwischen NMEA-AIS-Feldern und Datentypen auf diese des relevanten AVRO-Schemas. Dieses Schema wird durchgehend bis zur Anomalieerkennung genutzt. Wird eine

Anomalie erkannt, so werden die erkannten und berechneten Werte der jeweiligen Anomalie in einem anderen, speziell für lesbare Anomalie-Berichte angepassten AVRO-Schema mit Namen AisAnomalyReport gespeichert. Diese internen AVRO-Schemata werden nun erklärt, zuerst das AVRO-Schema AisMessage. (Referenz: (U.S. Coast Guard Navigation Center, 2019))

AIS Message ID	Feldname	AVRO-Schemafeld	Feldtyp	Bemerkung
1, 2, 3 (CLASS A AIS POSITION REPORT)	Message ID	message Id	ENUM: Messageld	Optional: Kennnummer zur Identifizierung des Messagetypes
	User ID	mmsi	int	Die MMSI des Schiffes
	Navigational status	navState	ENUM: Naviga- tionState	Optional: Der Navigationsstatus des Schiffes
	Rate of turn	turnRate	float	Optional: Zeitliche Kursänderung des Schiffes in Rad pro Sekunde
	SOG	CourseSpeed. SpeedOverGro und	float	Geschwindigkeit über Grund des Schiffes in Meter pro Sekunde
	Position accu racy	positionAccura cy	boolean	Optional: Genauigkeit der Positionsbestimmung (True entspricht einer Genauigkeit von unter 10 Meter)
	Longitude	Position.Longit ude	Winkel in [- π ,+ π]	Geographische Länge der Position
	Latitude	Position.Latitu de	Winkel in [- π ,+ π]	Geographische Breite der Position
	COG	CourseSpeed. CourseOverGr ound	float	Kurs über Grund des Schiffes in Rad
	True heading	trueHeading	float	Optional: Schiffskurs in Rad
	Time stamp	year	int	Jahr in dem die AIS-Nachricht verschickt wurde
		month	int	Monat in dem die AIS-Nachricht verschickt wurde
		day	int	Tag an dem die AIS-Nachricht verschickt wurde
		timeStamp	long	Zeitstempel der AIS-Nachricht in Millisekunden
special maneuver indicator	maneuverIndic ator	ENUM: Maneuver- Indicator	Optional: Gibt an, ob das Schiff in einem speziellen Manöver involviert ist	
5 (AIS CLASS A SHIP STATIC	Message ID	message Id	ENUM: Messageld	Optional: Kennnummer zur Identifizierung des Messagetypes

AND VOYAGE RELATED DATA)	User ID	mmsi	int	Die MMSI des Schiffes
	IMO number	imo	int	Optional: IMO-Nummer des Schiffes
	Call sign	callSign	string	Optional: Call Sign des Schiffes
	Name	shipName	string	Optional: Schiffsname
	Type of ship and cargo type	shipType	ENUM: ShipType	Optional: Schiffstyp
	Overall dimension/ reference for position	vesselLength	int	Optional: Schiffslänge in Meter
		vesselWidth	int	Optional: Schiffsbreite in Meter
	Type of electronic position fixing device	epfd	ENUM: Epfd	Optional: Typ des elektronischen Positionsendesystems
	ETA	timeStampEstimatedTimeOfArrival	long	Optional: Zeitstempel der voraussichtlichen Ankunftszeit in Millisekunden
	Maximum present static draught	draught	float	Optional: Maximaler Tiefgang des Schiffes in Meter
Destination	destination	string	Optional: Ankunftsort des Schiffes	
18 & 19 AIS STANDARD (EXTENDED) CLASS B EQUIPMENT POSITION REPORT	Message ID	message Id	ENUM: Messageld	Optional: Kennnummer zur Identifizierung des Messagetyps
	User ID	mmsi	int	Die MMSI des Schiffes
	SOG	CourseSpeed.SpeedOverGround	float	Geschwindigkeit über Grund des Schiffes in Meter pro Sekunde
	Position accuracy	positionAccuracy	boolean	Optional: Genauigkeit der Positionsbestimmung (True entspricht einer Genauigkeit von unter 10 Meter)
	Longitude	Position.Longitude	Winkel in $[-\pi, +\pi]$	Geographische Länge der Position
	Latitude	Position.Latitude	Winkel in $[-\pi, +\pi]$	Geographische Breite der Position
	COG	CourseSpeed.CourseOverGround	float	Kurs über Grund des Schiffes in Rad
	True heading	trueHeading	float	Optional: Schiffskurs in Rad

	Time stamp	year	int	Jahr in dem die AIS-Nachricht verschickt wurde
		month	int	Monat in dem die AIS-Nachricht verschickt wurde
		day	int	Tag an dem die AIS-Nachricht verschickt wurde
		timeStamp	long	Zeitstempel der AIS-Nachricht in Millisekunden
	Name	shipName	string	Optional: Schiffsname
	Type of ship and cargo type	shipType	ENUM: ShipType	Optional: Schiffstyp
	Overall dimension/reference for position	vesselLength	int	Optional: Schiffslänge in Meter
		vesselWidth	int	Optional: Schiffsbreite in Meter
Type of electronic position fixing device	epfd	ENUM: Epfd	Optional: Typ des elektronischen Positionssendesystems	
24 (STATIC DATA REPORT)	Message ID	message Id	ENUM: MessageId	Optional: Kennnummer zur Identifizierung des Messagetyps
	User ID	mmsi	int	Die MMSI des Schiffes
	Name	shipName	string	Optional: Schiffsname
	Type of ship and cargo type	shipType	ENUM: ShipType	Optional: Schiffstyp
	Vendor ID	vendorId	string	Optional: Schiffsidentifikationsnummer des Herstellers
	Call sign	callSign	string	Optional: Call Sign des Schiffes
	Dimension of ship/reference for position.	vesselLength	int	Optional: Schiffslänge in Meter
		vesselWidth	int	Optional: Schiffsbreite in Meter
Type of electronic position fixing device	epfd	ENUM: Epfd	Optional: Typ des elektronischen Positionssendesystems	
Externe Information		entityId	UUID	Projektinterne Referenznummer des Schiffes

Tabelle 3-3: Mapping der externen Informationsfelder ins interne AVRO-Schema AisMessage

Verwendete Aufzählungstypen (Enums)

Aufzählungstyp	Konstanten
ShipType	NOT_AVAILABLE, RESERVED_FUTURE_USE1, RESERVED_FUTURE_USE2, RESERVED_FUTURE_USE3, RESERVED_FUTURE_USE4, RESERVED_FUTURE_USE5, RESERVED_FUTURE_USE6, RESERVED_FUTURE_USE7, RESERVED_FUTURE_USE8, RESERVED_FUTURE_USE9, RESERVED_FUTURE_USE10, RESERVED_FUTURE_USE11, RESERVED_FUTURE_USE12, RESERVED_FUTURE_USE13, RESERVED_FUTURE_USE14, RESERVED_FUTURE_USE15, RESERVED_FUTURE_USE16, RESERVED_FUTURE_USE17, RESERVED_FUTURE_USE18, RESERVED_FUTURE_USE19, WING_IN_GROUND, WING_IN_GROUND_HAZARDOUS_CAT_A, WING_IN_GROUND_HAZARDOUS_CAT_B, WING_IN_GROUND_HAZARDOUS_CAT_C, WING_IN_GROUND_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE25, RESERVED_FUTURE_USE26, RESERVED_FUTURE_USE27, RESERVED_FUTURE_USE28, RESERVED_FUTURE_USE29, FISHING, TOWING, TOWING_LENGTH_200M_OR_BREADTH_25M, DREDGING_OR_UNDERWATER_OPS, DIVING_OPS, MILITARY_OPS, SAILING, PLEASURE_CRAFT, RESERVED_FUTURE_USE38, RESERVED_FUTURE_USE39HIGH_SPEED_CRAFT, HIGH_SPEED_CRAFT_HAZARDOUS_CAT_A, HIGH_SPEED_CRAFT_HAZARDOUS_CAT_B, HIGH_SPEED_CRAFT_HAZARDOUS_CAT_C, HIGH_SPEED_CRAFT_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE45, RESERVED_FUTURE_USE46, RESERVED_FUTURE_USE47, RESERVED_FUTURE_USE48, RESERVED_FUTURE_USE49, PILOT_VESSEL, SEARCH_AND_RESCUE_VESSEL, TUG, PORT_TENDER, ANTI_POLLUTION_EQUIPMENT, LAW_ENFORCEMENT, SPARE_LOCAL_VESSEL_1, SPARE_LOCAL_VESSEL_2, MEDICAL_TRANSPORT, NONCOMBATANT_SHIP_ACCORDING_RESOLUTION_NO_18, PASSENGER, PASSENGER_HAZARDOUS_CAT_A, PASSENGER_HAZARDOUS_CAT_B, PASSENGER_HAZARDOUS_CAT_C, PASSENGER_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE65, RESERVED_FUTURE_USE66, RESERVED_FUTURE_USE67, RESERVED_FUTURE_USE68, PASSENGER_NO_ADDITIONAL_INFORMATION, CARGO, CARGO_HAZARDOUS_CAT_A, CARGO_HAZARDOUS_CAT_B, CARGO_HAZARDOUS_CAT_C, CARGO_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE75, RESERVED_FUTURE_USE76, RESERVED_FUTURE_USE77, RESERVED_FUTURE_USE78, CARGO_NO_ADDITIONAL_INFORMATION, TANKER, TANKER_HAZARDOUS_CAT_A, TANKER_HAZARDOUS_CAT_B, TANKER_HAZARDOUS_CAT_C, TANKER_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE85, RESERVED_FUTURE_USE86, RESERVED_FUTURE_USE87, RESERVED_FUTURE_USE88, TANKER_NO_ADDITIONAL_INFORMATION, OTHER_TYPE, OTHER_TYPE_HAZARDOUS_CAT_A, OTHER_TYPE_HAZARDOUS_CAT_B,

	OTHER_TYPE_HAZARDOUS_CAT_C, OTHER_TYPE_HAZARDOUS_CAT_D, RESERVED_FUTURE_USE95, RESERVED_FUTURE_USE96, RESERVED_FUTURE_USE97, RESERVED_FUTURE_USE98, OTHER_TYPE_NO_ADDITIONAL_INFORMATION
NavigationState	UNDER_WAY_USING_ENGINE, AT_ANCHOR, NOT_UNDER_COMMAND, RESTRICTED_MANEUVER_ABILITY, CONSTRAINED_BY_DRAUGHT, MOOREDAGROUND, ENGAGED_IN_FISHING, UNDER_WAY_SAILING, HSC_FUTURE_AMENDMENT, WIG_FUTURE_AMENDMENT, POWER_DRIVEN_VESSEL_TOWING_ASTERN, POWER_DRIVEN_VESSEL_PUSHING_AHEAD_OR_TOWING_ALONGSIDE, RESERVED_FUTURE_USAGE, AIS_SART_ACTIVE, NOT_DEFINED
ManeuverIndicator	NOT_AVAILABLE, NO_SPECIAL_MANEUVER, SPECIAL_MANEUVER
MessageId	POSITION_REPORT_CLASS_A_SCHEDULED, POSITION_REPORT_CLASS_A_ASSIGNED_SCHEDULE, POSITION_REPORT_CLASS_A_RESPONSE_TO_INTERROGATION, BASE_STATION_REPORT, SHIP_AND_VOYAGE_RELATED_DATA, ADDRESSED_BINARY_MESSAGE, BINARY_ACKNOWLEDGEMENT, BINARY_BROADCAST_MESSAGE, STANDARD_SAR_AIRCRAFT_POSITION_REPORT, UTC_AND_DATE_INQUIRY, UTC_AND_DATE_RESPONSE, ADDRESSED_SAFETY_RELATED_MESSAGE, SAFETY_RELATED_ACKNOWLEDGEMENT, SAFETY_RELATED_BROADCAST_MESSAGE, INTERROGATION, ASSIGNMENT_MODE_COMMAND, DGNSS_BINARY_BROADCAST_MESSAGE, STANDARD_CLASS_B_EQUIPMENT_POSITION_REPORT, EXTENDED_CLASS_B_EQUIPMENT_POSITION_REPORT, DATALINK_MANAGEMENT_MESSAGE, AIDS_TO_NAVIGATION_REPORT, CHANNEL_MANAGEMENT, GROUP_ASSIGNMENT_COMMAND, STATIC_DATA_REPORT, SINGLE_SLOT_BINARY_MESSAGE, MULTIPLE_SLOT_BINARY_MESSAGE, LONG_RANGE_BROADCAST_MESSAGE, NOT_DEFINED
Epdf	UNDEFINED, GPS, GLONASS, COMBINED_GPS_GLONASS, LORAN_CCHAYKA, INTEGRATED_NAVIGATION_SYSTEM, SURVEYED, GALILEO, INTERNAL_GNSS
AnomalyKind	MILITARY_AIRCRAFT_CLASSICATION, VESSEL_SPEED_HEATMAP, VESSEL_COUNT_HEATMAP, HARD_VESSEL_TURN, AIS_SIGNAL_LOSS, VESSEL_CLOSE_APPROACH, SMALL_FAST_BOAT_AT_OPEN_SEAS, STATIC_DATA_CHANGE, DOUBLE_IMO_CALLSIGN_IN_THE_SITUATION_PICTURE, POTENTIAL_SMUGGLING
AnomalyState	NEW, UPDATED, OBSOLETE

Tabelle 3-4: Verwendete Aufzählungstypen (Enums)

Mehrfach genutzte Datentypen

Feldname	Feldtyp	Werte	Bemerkung
Position	lat	Winkel in $[-\pi, +\pi]$	Geographische Länge der Position

	lon	Winkel in $[-\pi, +\pi]$	Geographische Breite der Position
CourseSpeed	CourseOverGround	float	Kurs über Grund des Schiffes in Rad
	SpeedOverGround	float	Geschwindigkeit über Grund des Schiffes in Meter pro Sekunde

Tabelle 3-5: Mehrfach genutzte Datentypen

Das für die Ausgabe von Anomalie-Berichten notwendige AVRO-Schema ist das AisAnomalyReport-Schema, welches nach einer Anomalieentdeckung genutzt und belegt wird:

Feldname	Feldtyp	Bemerkung
anomalyKind	AnomalyKind	Typ der Anomalie
aisSignalLoss	AisSignalLoss	Optional: Zusätzliche Informationen, falls es eine Ais-Signal-Loss-Anomalie ist
AisSignalLoss.lastPos	Position	Optional: Letzte Position des Schiffes vor dem Signalabbruch
AisSignalLoss.lastCourseSpeed	CourseSpeed	Optional: Letzter CourseSpeed des Schiffes vor Signalabbruch
AisSignalLoss.duration	long	Optional: Länge des Signalabbruchs in Sekunden
AisSignalLoss.expectedPos	Position	Optional: Erwartete berechnete Position des Schiffes bei erneuter Signalsendung
AisSignalLoss.expectedDistance	float	Optional: Erwartete berechnete Distanz des Schiffes zur letzten Position bei erneuter Signalsendung in Meter
AisSignalLoss.distanceToCurrentPos	float	Optional: Distanz des Schiffes von letzter Position vor Signalabbruch zur aktuellen Position bei erneuter Signalsendung in Meter
AisSignalLoss.courseSpeedOffset	CourseSpeed	Optional: Veränderung des CourseSpeed von Anfang bis Ende des Signalabbruches
hardVesselTurn	HardVesselTurn	Optional: Zusätzliche Informationen, falls es eine Hard-Vessel-Turn-Anomalie ist
HardVesselTurn.turnRate	float	Optional: Turnrate des Schiffes in Rad pro Sekunde
HardVesselTurn.totalCourseChange	float	Optional: Gesamte Veränderung des Kurses während der gefahrenen Kurve in Rad
HardVesselTurn.startCourse	float	Optional: Kurs des Schiffes zu Beginn der gefahrenen Kurve in Rad
HardVesselTurn.endCourse	float	Optional: Kurs des Schiffes am Ende

e		der gefahrenen Kurve in Rad
vesselCloseApproach	VesselCloseApproach	Optional: Zusätzliche Informationen, falls es eine Vessel-Close-Approach-Anomalie ist
VesselCloseApproach.vesselCloseApproachList	array<VesselCloseApproachInfo>	Liste an Schiffsinformationen von den Schiffen, die an einer Vessel-Close-Approach-Anomalie beteiligt sind
VesselCloseApproach.VesselCloseApproachInfo.mmsi	int	Die MMSI des nahen Schiffes
VesselCloseApproach.VesselCloseApproachInfo.distance	float	Abstand dieses Schiffes zum anomaliauslösenden Schiff
VesselCloseApproach.VesselCloseApproachInfo.position	Position	Position des nahen Schiffes
VesselCloseApproach.VesselCloseApproachInfo.shipType	ShipType	Optional: Schiffstyp des nahen Schiffes
VesselCloseApproach.VesselCloseApproachInfo.entityId	string	Optional: Projektinterne Referenznummer des nahen Schiffes
smallFastBoat	SmallFastBoat	Optional: Zusätzliche Informationen, falls es eine Small-Fast-Boat-Anomalie ist
SmallFastBoat.courseSpeed	CourseSpeed	Optional: CourseSpeed des Schiffes
SmallFastBoat.vesselLength	int	Optional: Länge des Schiffes in Meter
SmallFastBoat.vesselWidth	int	Optional: Breite des Schiffes in Meter
doubleImoCallsign	DoubleImoCallsign	Optional: Zusätzliche Informationen, falls es eine Double-Imo-Callsign-Anomalie ist
DoubleImoCallsign.duplicatedValue	string	Optional: Gibt an ob die IMO-Nummer oder das Call-Sign oder beides doppelt vorkommt
DoubleImoCallsign.imo	int	Optional: Doppelte IMO-Nummer der Schiffe
DoubleImoCallsign.callSign	string	Optional: Call-Sign der Schiffe
DoubleImoCallsign.doubleImoCallsignList	DoubleImoCallsignInfo	Liste an Schiffsinformationen von den Schiffen, die an einer Double-Imo-Callsign-Anomalie beteiligt sind
DoubleImoCallsign.DoubleImoCallsignInfo.mmsi	int	Die MMSI des beteiligten Schiffes

DoubleImoCallsign. DoubleImoCallsignInfo.m msi	Position	Position des nahen Schiffes
DoubleImoCallsign. DoubleImoCallsignInfo.m msi	string	Optional: Projektinterne Referenznummer des nahen Schiffes
smuggling	Smuggling	Optional: Zusätzliche Informationen, falls es eine Smuggling-Anomalie ist
Smuggling.hasSuspiciousS hiptype	boolean	Angabe, ob das Schiff einen verdächtigen Schiffstyp besitzt
Smuggling.isInSuspiciousA rea	boolean	Angabe, ob sich das Schiff in einem verdächtigen Gebiet befindet
Smuggling.vesselCloseApp roachList	array<VesselCloseApproachInfo >	Liste an Schiffsinformationen von den Schiffen, die an Smuggling-Anomalie beteiligt sind
timeOfOccurrence	long	Zeitpunkt zu dem die Anomalie passiert ist
positionOfOccurrence	Position	Position an der die Anomalie passiert ist
timeOfReport	long	Zeitpunkt zu dem der Anomalybericht erstellt wird
mmsi	int	MMSI des Schiffes
imo	int	Optional: Die IMO-Nummer des Schiffes
shipName	string	Optional: Name des Schiffes
uuid	string	Eindeutige Referenznummer des Anomalieberichtes
state	AnomalyState	Status der Anomalie
significance	float	Optional: Signifikanzwert der Anomalie
source	string	Eindeutiger Bezeichner der Datenquelle
entityId	string	Optional: Projektinterne Referenznummer des Schiffes
nmeaSentences	array<string>	Optional: Liste aller dazugehörigen NMEA-Sätze

Tabelle 3-6: Internes AVRO-Schema AisAnomalyReport

3.2.2.3. Datenvereinbarungskomponenten

In diesem Abschnitt wird dargestellt, aus welchen Quellen die Schiffsdaten ausgelesen werden können, in welchem Format diese Daten vorliegen und wie sie in das Anomalieerkennungssystem eingespeist werden können.

Wie in Grundlagen AIS-Daten und internes AVRO-Schema AisMessage dargelegt, ist der erste Schritt der Datenvereinbarung die Konvertierung von NMEA-Sätzen in das gewählte AVRO-Schema. Im Umfeld der regelbasierten Anomalieerkennung in IntelliMar wird dies

durch die externe, lizenzierte Software-Bibliothek aismessages, Version 3.0.2 durchgeführt (Salling, 2021).

Datenvereinnahmung als Text

Nun stellt sich die Frage, auf welchen Wegen diese NMEA-Sätze kommuniziert werden. Da gibt es zum einen den simplen Ansatz mit Textdateien, in denen zeilenweise solch ein Tupel aus Zeitstempel und NMEA-Satz stehen. Diese Dateien müssen eingelesen werden und via eines Kafka-Consumers in Apache Kafka eingespielt werden, womit die Vorprozessierung der Daten beginnt. Solche Kafka-Consumer können entweder selbst implementiert werden oder durch eine schon vorhandenes Tool von Kafka, den Kafka-Console-Consumer, benutzt werden. Neben dem manuellen Einspielen der Daten gibt es auch eine andere Variante, nämlich die des Datenstromes. Dabei müssen Daten nicht zeilenweise eingegeben und ins System gebracht werden, sondern werden direkt von einem Konsumenten übernommen, d. h. man spart sich den Weg, die Daten aus einer Datei erst in einen Datenstrom zu verwandeln. Die Daten, die als Datenstrom vorliegen, können über zwei Wege erhalten werden. Der eine Weg um diese Daten zu empfangen ist über einen TCP-Port (Transmission Control Protocol). Mithilfe dieses Protokolls können Daten einfach zwischen verschiedenen Rechnern ausgetauscht werden. Mittels eines selbst geschriebenen Connectors wird der Datenstrom aus dem TCP-Port konsumiert und direkt in ein Kafka-Topic geliefert.

Datenvereinahmung aus proprietären Schnittstellen

Die Architektur zum Datenaustausch in IntelliMar sieht vor, Daten über das DSG unter den Netzwerkteilnehmern zu verteilen. Das DSG setzt bei der Datenübermittlung nicht auf das Prinzip einer einfachen Distribution der Information in zeilenorientierten Textdaten, sondern bietet ein API, welches Informationsinhalte gemäß einer wohldefinierten Objektstruktur dem Empfänger übermittelt. Voraussetzung für die Nutzung eines APIs ist das Schaffen und Implementieren eines Endpunktes zu dieser API. Naturgemäß definieren APIs nicht nur den genauen Inhalt der Daten sondern auch eine Vorgehensweise – ein Protokoll. Das Protokoll regelt die Initialisierung, den Empfang, die Quittierung und die Terminierung des Empfangs. Die Spezifikation des API ist Bestandteil der Studienleistung des Verbundleiters, Airbus DS in Bremen und wird hier nicht weiter detailliert.

Im Rahmen des Arbeitspaketes 1510 wurde Implementierung, Test und Integration durchgeführt. Aufgrund des komplexen Charakters des APIs und dem zu implementierenden Paradigmenwechsel zwischen einerseits Einzelvereinnahmung von NMEA-Datenobjekten und andererseits Rückführung in das Apache-Kafka-System, erfordert diese Art der gewählten Integration einen deutlich höheren Aufwand als der Rückgriff auf standard- und textbasierte Verfahren. Im Rahmen von IntelliMar sind demnach beide Connector-Typen implementiert, die DSG-Variante für den operationellen integrierten IntelliMar-Betrieb im Verbund und die rein textbasierte Variante zum Test sowie zur Validierung mit historischen Datenvorräten.

3.2.2.4. Datenvorprozessierung

Damit die rohen hereinkommenden AIS-Daten bearbeitet und auf Anomalien geprüft werden können, müssen diese erstmals gefiltert, aufbereitet werden (Data Cleaning). Dazu werden im Folgenden die dafür wichtigen Streaming-Nodes beschrieben.

AIS-MMSI-Partitioner

Dieser Streaming-Node ist dazu da, die empfangenden AIS-Daten, welche als NMEA-Sätze dargestellt sind, zu überprüfen, zu filtern und auszusortieren. Denn die Schiffsdaten kommen ungefiltert im System an, lediglich nach der Zeit sortiert. Dazu kommt nicht nur der **NMEA-Satz** sondern auch ein **Zeitstempel** und eine **EntityId**, die in das System geschickt werden. Die EntityId ist eine eindeutige Kennung für einen Datensatz ähnlich der MMSI. Diese EntityId wurde sehr spät während der Studie eingefügt und für alle Studienpartner verpflichtend als Kennung jedes Schiffes eingesteuert. Die komplette Eingangsnachricht sieht dann folgendermaßen aus:

2020-05-29T09:17:25+00:00 8c988418-0000-037f-0000-01725fa6e668

!AIVDM,1,1,,A,B39R4600lp9G;IWbL:?NGwU1I86<,0*16

Da das regelbasierte Anomalieerkennungssystem nicht alle Daten verwertet bzw. viele Arten an AIS-Daten überflüssig sind, müssen die Rohdaten erst gefiltert werden. Zudem spielt der AIS-MMS-Partitioner eine wichtige Rolle bei der parallelisierten Datenprozessierung: Er verteilt (partitioniert) die Daten sinnvoll auf mehrere Verarbeitungskanäle (siehe Kapitel 3.2.1.2). Zunächst sortiert dieser Streaming-Node alle NMEA-Sätze aus, welche nicht von der Anomalieerkennung unterstützt werden. Für die regelbasierte Anomalieerkennung werden die AIS-Typen 1, 2, 3, 5, 18, 19 und 25 unterstützt, bei denen es sich um die statischen Daten und um die Bewegungsdaten handelt. Diese werden benötigt, um Felder wie die MMSI, die Geschwindigkeit oder die Position des Schiffes zu erhalten. Außerdem prüft der AIS-MMSI-Partitioner auch die syntaktische Korrektheit und Vollständigkeit der Zeitstempel, EntityId und NMEA-Satzkombination, da bei der Datenübertragung Fehler oder Probleme auftreten können.

In Kafka besteht jeder Datensatz aus einem Wert und aus einem Schlüssel. Als Wert wird der Zeitstempel plus dem NMEA-Satz verwendet. Doch der Schlüssel muss ein eindeutiger Schlüssel sein, daher bietet sich in diesem Fall die MMSI an, da diese eine eindeutige Zuweisung zu einem Schiff gibt. Somit wird in diesem Streaming-Node zusätzlich zur Aussortierung fehlerhafter, unvollständiger und inkompatibler Datensätze noch die Extraktion der MMSI durchgeführt, um diese dann als eindeutigen Schlüssel für den Kafka-Datensatz zu setzen. Dieser Streaming-Node muss sehr flexibel sein, da sich während der Studie IntelliMar häufig das Format der eingehenden Daten verändert hat. Somit muss es möglich sein, verschiedenste Eingangsformate zu verstehen und genauso filtern und prozessieren zu können.

Sind all diese Schritte absolviert, so werden die gefilterten Tripel aus Zeitstempel, EntityId und NMEA-Satz in das Output-Topic in Kafka geschickt, wo sie dann vom AIS-In-Stream konsumiert werden.

AIS-In-Stream

Der AIS-In-Stream ist der nächste Streaming-Node nach dem AIS-MMSI-Partitioner in der Datenvorprozessungskette. Hier werden die vorgefilterten Tripel aus Zeitstempel, EntityId und NMEA-Sätze dekodiert und auf das interne AVRO-Schema gemappt. Das Dekodieren wird hier von der lizenzierten aismessages-Bibliothek von T. B. Salling (Salling, 2021) absolviert; auf eine aufwändige Eigenimplementierung wurde verzichtet. Die Datentypen der externen Bibliothek passen jedoch nicht direkt zu den gewählten Datentypen des AisMessage-AVRO-Schemas und müssen daher gemappt werden. Unter anderem müssen Einheiten umgerechnet werden, fehlende Daten ergänzt werden oder auch die Zeitstempel verstanden und umgewandelt werden. Die EntityId wird direkt durch den Streaming-Node geschleust und dann im AVRO-Schema gesetzt. Der AIS-In-Stream dient somit der Kommunikation mit der externen Bibliothek und als Mapper auf das AisMessage-AVRO-Schema.

Aggregator

Im Aggregator kommen alle AIS-Daten an, sowohl die statischen als auch die Bewegungsdaten. Diese müssen nun zusammengeführt werden, so dass es nur noch eine AisMessage zu jedem Schiff gibt, die alle Informationen vorhält. Die jeweiligen AisMessages aus der vorigen Konvertierung sind also nur zur Hälfte belegt und müssen nun mit ihrem Pendant vereinigt werden. Der Schlüssel, der zur Vereinigung dient, ist die MMSI. Dies passiert im Aggregator, indem dieser zwei Caches bzw. Zwischenspeicher anlegt, in dem zum einen die statischen Daten und zum anderen die Bewegungsdaten gespeichert werden und darauf warten, dass der passende andere Datentyp von diesem Streaming-Node konsumiert wird. Ändern sich bei den statischen Daten keine Werte, so gibt es kein Update des bereits vorhandenen Datensatzes. Kommt es jedoch zu einer Änderung, so wird der alte Datensatz durch den zeitlich neueren ersetzt. Bei den Bewegungsdaten wird jedes Mal ein Update durchgeführt; die Schiffposition hat sich geändert. Sind nun Daten von beiden Typen konsumiert und haben die gleiche MMSI, dann wird eine AisMessage daraus fusioniert und an das Kafka-Output-Topic gesendet. Um einen möglichst großen Datenbestand im Data Lake aufzubauen, wird in diesem Streaming-Node mithilfe eines weiteren Kafka-Producer sowohl die statischen als auch die Bewegungsdaten an ein weiteres Topic gesendet, das dann von einem Connector konsumiert wird und der die Daten dann im HDFS des Data Lakes abspeichert. Bei Schiffen mit Transponder-Typ A werden die Daten mit AIS Message ID 1, 2, 3 (Positionsdaten) mit den Daten mit AIS Message ID 5 (statische Daten) aggregiert. Bei Schiffen mit Transponder-Typ B werden die Daten mit AIS Message ID 18 und 19 (Positionsdaten) mit den Daten mit AIS Message ID 24 (statische Daten) aggregiert.

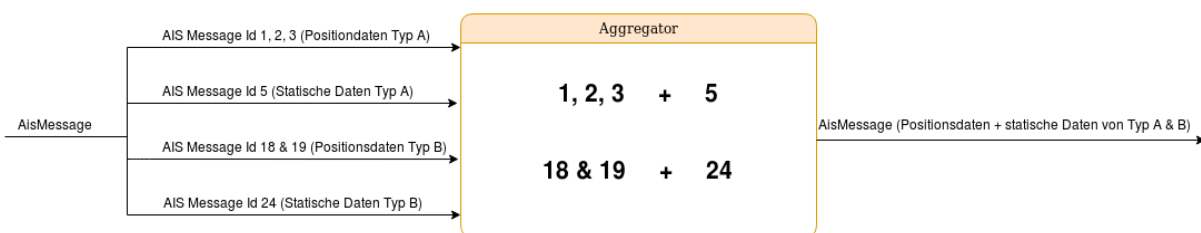


Abbildung 3-21: Schematische Darstellung der Funktionsweise des Aggregators

Data Cleaner

Aufgrund von Unzulänglichkeiten in den AIS Daten wie Positionssprünge oder inkonsistente volatile Geschwindigkeits- und Kursdaten, werden die AIS Daten zunächst gesäubert und geglättet hauptsächlich bezogen auf kinematische Daten. Dies wird für jedes Objekt durchgeführt, wobei das MMSI-Attribut als eindeutige Zuordnung der AIS Messages zu Objekten dient. Die AIS Nachrichten werden als Plot-Messungen mit Position, Geschwindigkeit und geschätzter Genauigkeit behandelt und werden durch einen Tracking-Algorithmus (Kalman-Filter) prozessiert, um die entsprechenden Zustandsvariablen und damit insgesamt das Trackingergebnis zu verbessern.

Das Kalman-Filter wird genutzt, um verrauschte oder anderweitig ungenaue Daten zu verbessern, indem die Zustandsvariablen wie Position und Geschwindigkeit präzise geschätzt werden und dadurch die Abweichungen zur Originalspur des Ziels reduziert werden. In jedem Zeitschritt werden die bisherigen Schätzwerte mit den neuen Messungen kombiniert und der daraus resultierende Fehler neu bewertet und darüber hinaus werden auch Korrelationszusammenhänge zwischen den verschiedenen Variablen deutlich. (Wie stark ändert sich eine Variable in Bezug auf eine andere.)

Dafür wird für jede empfangene AIS Nachricht die vorhergehende Track-Position auf den aktuellen Zeitstempel extrapoliert, d.h. es wird berechnet, wo sich das Ziel auf Grund des bisherigen Spurverlaufs und dem Zeitabstand seit dem letzten Update befinden müsste. Falls der Plot mit dem Erwartungsgebiet übereinstimmt (siehe nachfolgende Abbildung), wird dieser dem Track zugeordnet und die neue Position berechnet, indem die gemessene Position und die erwartete extrapolierte Position unter Verwendung des konfigurierten dynamische Modells gewichtet und verrechnet werden. In der folgenden Abbildung ist der Zusammenhang zwischen Plot, aktuellem Track-Zustand und anderen Einflussgrößen vereinfacht dargestellt.

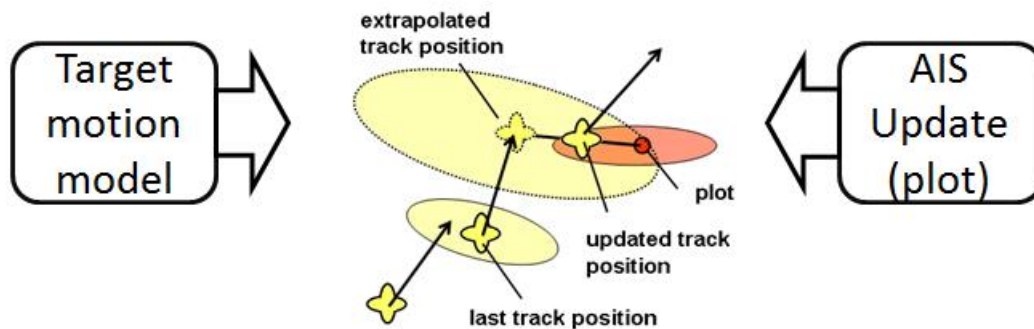


Abbildung 3-22: Vereinfachte Darstellung des Prinzips des Trackings

Die Größe des Erwartungsgebiets hängt vor allem ab von

- der Agilität des Ziels (Manövrierfähigkeit) und
- dem Zeitraum zwischen aufeinanderfolgenden Plots

Was die Agilität des Ziels betrifft, so ist das System aktuell für maritime Ziele optimiert, die typischerweise eine lineare Bewegung aufweisen.

Allerdings fiel in der Datenanalyse auf, dass lineare Trajektorien nicht immer auch konsistent in den Daten abgebildet werden. Konkret fiel auf, dass der Kurs teilweise sehr stark schwankte, obwohl anhand der geographischen Positionen eine nahezu lineare Bewegung des Objekts nachzuvollziehen war. In der folgenden Abbildung ist eine Zeitreihe des Kurses (in Grad) eines Schiffes zu sehen, die den Unterschied zwischen gegebenem und durch den Algorithmus errechneten Kurs zeigt.

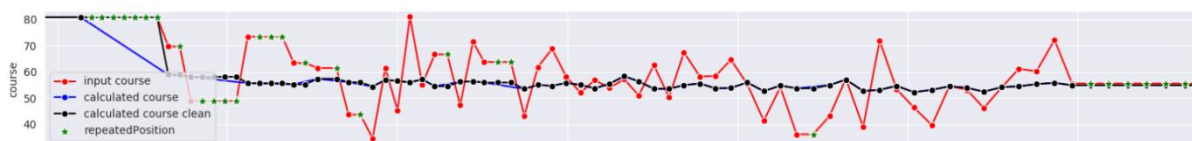


Abbildung 3-23: Unterschied zwischen gegebenem und durch den Algorithmus errechneten Kurs.

Auch unrealistische Positionssprünge, die auf fehlerhafte Daten hinweisen, werden durch einen entsprechenden Algorithmus ausgefiltert.

Für die Detektion von Anomalien ist es entscheidend, die verrauschten Daten wenn möglich zu verbessern, um die Erzeugung von falschpositiven Ergebnissen zu verhindern, die möglicherweise durch die Ungenauigkeit der empfangenen Daten entstehen.

3.2.2.5. Datenlieferkomponenten

Wurden Anomalien gefunden, so müssen diese zur weiteren Verarbeitung, Speicherung und Darstellung an Interessenten gesendet werden. Interessenten sind hier der Data Lake sowie

von Projektpartnern bereitgestellte Software-Komponenten wie beispielsweise das Anomlaie-Display. Die Anomalie-Berichte sollen in dafür geeigneter Form ausgeliefert werden. Um Vielfalt zu gewährleisten, übernehmen verschiedene sogenannte Report-Formatter diese Aufgabe. Report-Formatter dürfen gleichzeitig laufen und liefern eingehende Berichte in ihrem Zielformat aus.

JSON-Report-Formatter

Eine dieser Varianten ist der JSON-Report-Formatter. Für die Validierungskampagnen I und II wurde in diesem Schritt aus dem internen AVRO-Schema AisAnomalyReport ein JSON-Objekt realisiert. Dafür muss je Anomalie-Bericht ein JSON-Objekt erstellt und entsprechend mit den Daten aus den Anomalien befüllt werden. Das JSON-Format war in diesen Validierungskampagnen sehr hilfreich, da es sehr leicht und schnell konfigurierbar bzw. veränderbar ist.

Realisiert ist folgendes JSON-Schema

Feldname	Feldtyp	Bemerkung
Brain	string	Gibt an von welchem Projektpartner die Anomalie entdeckt und gesendet wird
Identifier	string	Projektinterne Abkürzung der Anomalie
OccurenceTime	string	Zeitpunkt zu dem die Anomalie passiert ist, in lesbarem Format
Title	string	Titel der Anomalie
Message	string	Ausführlicher Text, der die Anomalie genau beschreibt
Ship	array<string, int, int>	Liste an Informationen des Schiffes
Ship.Name	string	Name des Schiffes
Ship.MMSI	int	MMSI des Schiffes
Ship.IMO	int	Die IMO-Nummer des Schiffes
Position	Position	Position an der die Anomalie passiert ist
State	AnomalyState	Status der Anomalie
Significance	float	Signifikanzwert der Anomalie
Source	string	Eindeutiger Bezeichner der Datenquelle

Tabelle 3-7: JSON-Schema des JSON-Report-Formatters

```
{
  "Brain": "UL - Turn (AISHUB-LIVE)",
  "Identifier": "H208",
  "OccurenceTime": "2019-09-13T09:52:25Z",
  "Title": "Signal loss",
  "Message": "signal loss for vessel named ZUSE (IMO: 0, MMSI: 211724970) at position (53° 43.355' N, 008° 04.088' E) and time 2019-09-13T09:52:25Z, with the last known position (53° 37.432' N, 008° 08.165' E), the last known course over ground was 331.4°, the speed over ground was 15.38 m/s (29.9 kn), the duration of the signal loss was 874 s, the ship is expected at position (53° 43.792' N, 008° 02.314' E), with an expected distance of 13443.77 m (7.26 NM), and the distance to the current position is 11870.4 m (6.41 NM), the course over ground offset is -22.0°, the speed over ground offset is 17.23 m/s (33.5 kn), the state of the report is new and the significance of the anomaly is",
  "Ship": [
    "ZUSE",
    211724970,
    0
  ],
  "Ship.Name": "ZUSE",
  "Ship.MMSI": 211724970,
  "Ship.IMO": 0,
  "Position": "53.72888888888889, 8.048888888888889",
  "State": "NEW",
  "Significance": 0.5,
  "Source": "UL - Turn (AISHUB-LIVE)"
}
```

```
100.0.", "Ship": {"Name": "ZUSE", "MMSI": "211724970", "IMO": "0"}, "Position": "(53°  
43.355' N, 008° 04.088' E)", "lastPos": "(53° 37.432' N, 008° 08.165'  
E)", "lastcourseOverGround": 331.4, "lastspeedOverGround": 29.9, "duration": 874, "expecte  
dPos": "(53° 43.792' N, 008° 02.314'  
E)", "expectedDistance": 7.26, "distanceToCurPos": 6.41, "courseOverGroundOffset": -  
22.0, "speedOverGroundOffset": 33.5, "State": "new", "Significance": 100.0, "Source": "ASIH  
UB-LIVE"}
```

Beispiel Anomalie "AIS-Signal-Loss" im JSON-Format

In diesem Beispiel kann man eine AIS-Signal-Loss“-Anomalie sehen, welche von dem Report-Formatter ausgegeben wird. Dabei gibt es Felder, die direkt mit Werte gefüllt werden, aber auch das Feld „Message“, welches automatisch aus den vorhandenen Informationen gebaut werden muss, um dem Leser bzw. Operator eine lesbare Form der Anomalie zu geben. Neben den oben in der Tabelle angegebenen Feldern dieses JSON-Formates gibt es noch die Anomalie-spezifischen Felder. Diese sind genauso aus dem AVRO-Schema AisAnomalyReport übernommen, wie es dort beschrieben ist.

Diese Anomalie-Berichte werden ebenfalls via Apache Kafka über ein Topic verschickt und werden dann über eine TCP-Verbindung zu den Projektpartnern weitergeleitet.

Common-Values-Report-Formatter

Speziell zur Einlagerung im Data Lake oder zur externen Verarbeitung wurde dieser Streaming Node geschaffen. Er unterstützt durch Bereitstellen eines angepassten JSON-Formates. Insbesondere werden die bisherigen Positionen im GeoJSON-Format dargestellt, welches von MongoDB direkt als geolokales Suchfeld genutzt werden kann.

"Position": "(53° 43.355' N, 008° 04.088' E)"
wird zu einem

```
GeoPoint:  
" Position ": {"type": "Point", "coordinates": [53.8097222, 8.091111111111111]}.
```

Dies ist zudem hilfreich, um die Anomalien in weiteren Standard-Statistikwerkzeugen wie-Elasticsearch zu laden, um dort Auswertungen, Statistiken und Grafiken erstellen zu lassen. Im Zuge dieser Umformatierung werden auch die Einheiten konvertiert. Intern werden mit SI-Einheiten gearbeitet, wie z.B. Meter pro Sekunde als Einheit für die Geschwindigkeit. Da diese im maritimen Umfeld unüblich sind, wird diesem Umstand so für eine weitere externe Verarbeitung Rechnung getragen. Der Common-Values-Report-Formatter konvertiert die Geschwindigkeit in Knoten und die Entfernungen in nautische Meilen. Sind die Positionen und die Einheiten umgeformt, so werden die neuen Anomalie-Berichte in ein Apache Kafka-Topic geschickt und verteilt. In diesem Falle werden die Berichte von einem Kafka-Connector für MongoDB empfangen und so im Data Lake gesichert.

DSG-Datenlieferung

Im integrierten und operationellen Einsatz für IntelliMar soll wie für die **Datenvereinbarung aus proprietären Schnittstellen** das von Airbus DS Bremen bereitgestellte DSG genutzt werden. Auch hierzu stellt das DSG ein spezielles API zur Verfügung. Im Falle der Übermittlung von Anomalie-Berichten gestaltet sich dieses Interface gegenüber dem zum Dateneingang deutlich komplexer. Neu ist hier, dass die interne Struktur im DSG sich von

Anomalie-Update zu Anomalie-Update weiterentwickelt, um den Meldungsverlauf aus Anomalie-Updates möglichst bandbreiteneffizient abbilden zu können. Hierbei entsteht ein starker Paradigmenwechsel: Während das Bereitstellen von Anomalie-Updates aus den Anomalie-Detektoren stets umfänglich ist – also mit jeder Meldung eine aktuelle und vollständige Schiffs- und Positionsmeldung ist, soll beim Versand durch das DSG die Übermittlung auf jeweils die Änderungen zwischen 2 Meldungen reduziert werden.

Die Aufgabe dieses Report-Formatters ist daher nicht die simple Abbildung der Datenfelder in ein anderes Format, sondern umfasst folgende programmtechnisch umzusetzende Aspekte:

- Empfang und Pufferung von Anomalie-Berichten (eigentlich Anomalie-Updates)
- Notwendige Konvertierungen zwischen Anomalie-Detektor-Informationen- und DSG-Feldtypen
- Verfolgen von Änderungen zwischen sequenziellen Anomalie-Updates
- Bedienen des DSG-APIs zur Erstellung und Versand von Bericht-Updates unter Bereitstellung von Änderungstelegrammen
- Nutzen der DSG-eigenen Services zur Ermittlung von Schiffs- und Positionsinformationen anstelle der direkten Verwendung solcher in den Anomalie-Updates aus den Brains.

Details zum verwendeten DSG-API sind der Dokumentation im Rahmen des Schlussberichtes der Airbus DS Bremen zu entnehmen, sodass hier auf Darlegung verzichtet wird.

Da der Anschluss dieses DSG-Report-Formatters an das DSG-System die Nutzung eines von Airbus DS Bremen bereitgestellten DSG-Endknotens inklusive bereitgestellten Implementierungsframeworks erfolgen musste, erhöhte sich der Aufwand der Implementierung gegenüber den anderen Report-Formattern durch folgende Tätigkeiten:

- Iteratives Abstimmen der DSG-APIs zwischen den Studienpartnern
- Erlernen der speziellen Fähigkeiten zur Nutzung des API
- Bugtracking und Bugfixing
- Test
- Integrationsrunden

3.2.2.6. Filter

Eine wichtige Komponente bei der Anomalieerkennung ist die Filterung der Daten. Dies trägt zum einen dazu bei, die Datenlast der zu verarbeitenden Daten schrittweise zu minimieren und zum anderen ist sie auch zur Variierung der Anomalieerkennung nötig.

Welche Filter es für die regelbasierten Anomalieerkennung gibt, welche Funktionen diese haben und welche Parameter bestimmt werden können, wird im Folgenden beschrieben.

Speed-Filter (imar-speed-filter)

Dieser Streaming-Node-Filter wurde realisiert, um die durchlaufenden Daten der Schiffe auf deren Geschwindigkeit zu filtern. Dies wird vor allem gemacht, um stehende Schiffe aussortieren zu können. Dies ist für einige Anomalieerkennung durchaus sinnvoll, da man z.B. für einen 30°-Turn kein stehendes Schiff überprüfen muss. In bestimmten Fällen will man aber auch nur Schiffe betrachten, die nicht schneller als eine bestimmte Geschwindigkeit fahren. Daher gibt es in diesem Filter die Möglichkeit zwei Parameter zu setzen:

Parameter	Bedeutung
speedOverGround	Hier kann man den Grenzwert der Geschwindigkeit, nach der gefiltert werden soll, bestimmen. Dieser Wert wird in Knoten angegeben, also z.B. 2,5 Knoten. Die Kommandozeilenoption für diesen Parameter ist: -c
typeOfLimit	Hier kann man zwischen zwei Werten auswählen, zum einen der Wert „MIN“ und zum anderen der Wert „MAX“. Dies gibt an, ob der oben gewählte Geschwindigkeitsparameter SpeedOverGround als Minimumswert („MIN“) oder als Maximumswert („MAX“) genutzt wird. Die Kommandozeilenoption für diesen Parameter ist: -l

Tabelle 3-8: Parameter des Speed-Filters und deren Bedeutung

NavState-Filter (imar-navstate-filter)

Ein weiterer Streaming-Node zum Filtern der Schiffsdaten ist der NavState-Filter. Jedes Schiff sendet seinen Navigationsstatus beim Versenden seiner AIS-Nachricht mit. Dieser Status beschreibt mehr oder weniger genau, was das Boot gerade macht, z.B. gibt es die Navigationsstatus „UNDER_WAY_USING_ENGINE“, „AT_ANCHOR“, „SAILING“, etc. Dieser Status beschreibt, ob das Schiff gerade fährt, in einem Hafen vor Anker liegt, einfach am Segeln ist. Mit diesem Filter kann man also wieder recht schnell bestimmte Schiffe aussortieren, die für eine Anomalieerkennung nicht von Belang sind. Dies wird in diesem Streaming-Node so realisiert, indem man als Parameter eine Liste der (für die Anomalie wichtigen) Navigationsstatus angibt und somit die irrelevanten Schiffsdaten entfernt werden. Ein Beispiel ist die AIS-Signal-Loss-Anomalie, bei der Schiffe erkannt werden die eine bestimmte Zeit kein Signal mehr senden, weil sie es absichtlich ausgeschaltet haben oder etwas kaputt gegangen ist. Bei dieser Anomalie sind Schiffe, die vor Anker (also „AT_ANCHOR“) liegen sehr uninteressant, da diese nach einer Weile ihr AIS-Signal nicht mehr senden. Würde man für diese Boote trotzdem eine Anomalie erkennen, wäre dies offensichtlich eine False-Positiv-Erkennung und somit nicht erwünscht.

Option	Bedeutung
navigationState	Dieser Parameter mit einer Liste an Navigationsstatus gibt an, welche Navigationsstatus die durchlaufenden Schiffsdaten haben dürfen, um nicht aussortiert zu werden. Die Kommandozeilenoption für diesen Parameter ist: -n

Tabelle 3-9: Parameter des NavState-Filters und deren Bedeutung

ShipType-Filter

Dieser Streaming-Node filtert die eingehenden Daten nach dem Schiffstyp und gibt nur Schiffsdaten mit den gewollten Schiffstypen aus. Jedes Boot, das eine AIS-Nachricht verschickt, gibt auch den Schiffstyp bei der Übertragung des Signals mit an. Von diesen Typen gibt es sehr viele verschiedene, wie z.B. „TANKER“, „CARGO“, „PASSENGER“, „PLEASURE_CRAFT“, etc. Betrachtet man den Anomalie-Detektor zur Erkennung von schnellen, kleinen Booten auf offener See, dann ist z.B. ein „PLEASURE_CRAFT“ sehr verdächtig bzw. besorgniserregend, wenn dieses alleine und schnell auf hoher See kreuzt, denn dies ist abnormal für ein Freizeitboot. Daher ist dieses Filter auch ein wichtiger Bestandteil der Anomalieketten. Als Parameter wird dem ShipType-Filter genau wie beim NavState-Filter eine Liste von Werten übergeben, in diesem Fall eine Liste von Schiffstypen:

Option	Bedeutung
shipType	Diesem Parameter wird eine Liste an Schiffstypen übergeben. Die genannten Typen werden zum Filtern benutzt und die Schiffs, welche von einem dieser Typen sind, werden von dem Streaming-Node ausgegeben. Die Kommandozeilenoption für diesen Parameter ist: -t

Tabelle 3-10: Parameter des ShipType-Filters und deren Bedeutung

Region-Filter (imar-region-filter)

Wie der Name schon verrät, ist dieser Filter dazu geeignet, die Schiffsdaten auf deren Position also auf eine bestimmte Region zu filtern. Dies ist einer der wichtigsten Filter für die Anomalieerkennung, denn bestimmte Regionen bzw. Seegebiete sind interessanter oder auch gefährlicher als andere. Aber auch ein „Area of interest“ ist einer der Hauptgründe für diesen Streaming-Node. Betrachtet man z.B. die Regierung eines Landes, dann wird sich diese vor allem um ihr persönliches Hoheitsgewässer kümmern. Somit kann einfach bestimmt werden, in welcher Region die Schiffe weiterhin betrachtet werden sollen und alle Schiffe, die nicht darin wässern, werden automatisch aussortiert. Als Parameter benötigt der Region-Filter ein Quadrupel an Koordinaten, denn es wird mit meinem Viereck gearbeitet. Somit braucht man zum einen den Längen- und Breitengrad des nordwestlichen Eckpunktes des Vierecks und zum anderen die beiden Gradangaben des südöstlichen Eckpunktes. Dann wird automatisch vom Streaming-Node das Viereck berechnet und als einzugrenzendes Gebiet behandelt. Es reicht also ein Parameter:

Option	Bedeutung
region	Dieser Parameter gibt die Region an, in der die Schiffe fahren dürfen, um nicht aussortiert zu werden. Dazu werden zwei Punkte mit Längen- und Breitengrad angegeben, getrennt durch Schrägstriche („/“). Die Kommandozeilenoption für diesen Parameter ist: -r

Tabelle 3-11: Parameter des Region-Filters und deren Bedeutung

Polygon-Filter (imar-polygon-filter)

Ein Polygon ist ein Vieleck, welches aus mehreren Streckenzüge besteht. So sind z.B. ein Dreieck, ein Hunderteck oder auch ein „Donut“-Eck ein sogenanntes Polygon. Dieser Filter ist somit eine Verfeinerung des Region-Filters, welcher vorher erläutert wurde, da hier nicht

nur ein Viereck sondern ein Vieleck für eine genauere Region verwendet werden kann. Der Region-Filter kann zur Effizienzsteigerung sehr gut als Vorsortierung vor den Polygon-Filter geschaltet werden. Daher sollte immer ein Region-Filter vor einem Polygon-Filter benutzt werden.

Als Parameter wird ein Polygon benötigt, welches aus mindestens drei Punkten besteht. Es werden alle Eckpunkte des Polygons angegeben, wobei der erste und letzte Eckpunkt nicht identisch sein müssen. Der Streaming-Node spannt automatisch dieses Polygon auf und schließt es mit der Streck vom letzten zum ersten Eckpunkt ab. Somit hat man ein Polygon, bei dem das Innere zum Datenfiltern herangezogen wird, nur indem man den Parameter mit den einzelnen Eckpunkten angibt. Via Brain-Manager ist es möglich, vorgefertigte Polygone, welche Namen besitzen (z.B. „Nordsee“ oder „Hohe See“) und im Streaming-Node encodiert sind, auszuwählen und sofort zu benutzen. Dies erleichtert den Umgang mit Polygonen, welche sehr viele Eckpunkte haben. Dazu muss man im Brain-Manager-HMI bei dem Feld „regionLabel“ aus einer vorgefertigten Liste aussuchen, welches Gebiet erwünscht ist. Außerdem ist es mit diesem Filter möglich, ein Polygon zu erstellen, das alle Schiffe darin aussortiert, d.h. es ist möglich sowohl das Polygoninnere als auch das Polygonäußere zu benutzen, um Schiffe nach ihrem Aufenthaltsort zu filtern. Dies ist ebenfalls via Brain-Manager-HMI möglich, indem man das Feld „regionInbound“ mit einem Häkchen markiert und den Wert auf „True“ setzt, um das Innere des Polygons als Akzeptanzbereich zu wählen. Somit ergeben sich folgende Parameter:

Option	Bedeutung
polygon	Der Polygon-Parameter besteht aus einer endlich langen Aufzählung von Eckpunkten, welche als Tupel von Breitengrad und Längengrad des Punktes angegeben werden. Die einzelnen Eckpunkte werden durch einen Komma getrennt und die Koordinaten der jeweiligen Punkte durch einen Schrägstrich („/“). Die Kommandozeilenoption für diesen Parameter ist: -p
regionLabel	Liste an vordefinierten Gebieten, bei denen das Polygon schon vorgefertigt ist und die Eckpunkte dementsprechend für den Parameter Polygon genutzt werden.
regionInbound	Dieser Boolean-Parameter gibt an, ob das Innere oder das Äußere des Polygons als Akzeptanzbereich genutzt wird. Wird der Parameter auf „True“ gesetzt, so wird das Polygoninnere als Bereich benutzt.

Tabelle 3-12: Parameter des Polygon-Filters und deren Bedeutung

3.2.2.7. Aufbau, Betrieb und Verwaltung von Erkennungsketten

Eine Anomalie-Erkennungskette entsteht durch gleichzeitigen Betrieb der beteiligten Streaming-Nodes. Die systematische Verkettung von Streaming-Nodes wird durch geeignete Parametrisierung der einzelnen Streaming-Node-Prozesse erreicht. Als Betriebsplattform wird das Docker-System (Docker, Inc, 2021) genutzt. Der zum Docker-System gehörende Docker-Compose-Mechanismus bietet eine Möglichkeit zur Konfiguration – also dem Beschreiben einer Kette aus Streaming-Nodes - sowie eine Plattform für den parallelen Start und Stop der beteiligten Streaming-Node-Prozesse.

Zur Verwaltung während des Betriebs einer oder mehrerer Erkennungsketten dient das im Rahmen des Teilvorhabens IntelliMar Services entwickelte Werkzeug „Brain Manager“.

Im Folgenden werden die Komponenten Docker-Compose, das Erstellen von Kettenkonfigurationen und der Brain Manager vorgestellt.

Aufbau und Betrieb: Docker-Compose

Das Docker-Compose Tool wird dazu verwendet, um mehrere Docker-Container zu erstellen, zu verwalten und auszuführen. Das Tool dient vor allem der Erleichterung beim Managen vieler einzelner Docker-Container. Wie im Fall von IntelliMar können dadurch mehrere Container zusammengefasst und in einer Kette zusammengeführt werden.

Als Konfigurationsdatei wird eine sogenannte YAML-Datei genutzt. Diese beinhaltet alle nötigen Informationen, welche zur erfolgreichen Erstellung so einer multi-container Kette benötigt werden. Wurde so eine YAML-Datei korrekt gefertigt, so kann diese durch einfache Befehle gestartet, gestoppt oder geloggt werden, aber auch der Status der Kette kann angezeigt werden. Diese Befehle werden mittels Kommandozeile der Docker-Ausführungseinheit mitgeteilt, z.B. wird zum Starten solch einer Docker-Compose-Kette folgender Befehl genutzt:

```
docker-compose up
```

Dadurch werden alle in der YAML-Datei aufgeführten Docker-Container instanziiert und gestartet. Es können also durch einen einzelnen Befehl beliebig viele Container aktiviert werden.

Darstellung von Ketten in der Compose-Datei

Wie bereits erklärt, ist es mit einer Docker-Compose-Datei möglich eine Anomaliekette, welche aus mehreren einzelnen Docker-Container besteht, leicht und schnell zu erstellen und zu verwalten (Docker, Inc., 2021). Wichtig zu erwähnen ist, dass bei einer YAML-Datei die Einrückungen und das Layout syntaxrelevant sind, d.h. es kommt exakt darauf an, wie die einzelnen Zeilen eingerückt sind und das alles korrekt formatiert ist. Die YAML-Datei kann man in drei wichtige Sektionen unterteilen: „version“, „services“ und „networks“. Unter „version“ wird die genutzte Dockerversion geschrieben, in diesem Fall Version „3.1“. Unter „services“ stehen die einzelnen Streaming-Nodes, die man erstellen und laufen lassen möchte. Diese referenzieren auf ein Docker-Image und werden unter dem Punkt „entrypoint“ als passende Kommandozeile gestartet. Dort werden alle gewünschten Kommandozeilenparameter aufgelistet und mit Werten belegt. Es können unter „services“ beliebig viele Streaming-Nodes erzeugt und parametrisiert werden. Ist dies geschehen, so wird unter „networks“ eine Festlegung eines virtuellen Netzwerks für den Betrieb der im compose-File angelegten Konten gemacht.

Beispiel einer YAML-Datei:

```
1 version: "3.1"
2 services:
3     ais-region-filter:
4         image: dana-vs01:5000/imar/imar-region-filter:latest
5     networks:
```



```
6           - IntelliMar-aishub-aggregation-net
7       entrypoint:
8           - /usr/bin/java
9           - -jar
10          - imar-region-filter-1.2.0-SNAPSHOT.jar
11          - -i
12          - imar-ais-cleaned
13          - -o
14          - imar-ais-turn-filtered-by-region
15          - -a
16          - hard-vessel-turn/region-filter
17          - -b
18          - dana-s06:9092
19          - -s
20          - http://dana-s06:8081
21          - -k
22          - "FALSE"
23          - -g
24          - hard-vessel-turn-region-filter
25          - -r
26          - 60.0/40.0/10.0/20.0
27
28       imar-navstate-filter:
29       image: dana-vs01:5000/imar/imar-navstate-filter:latest
30       networks:
31           - IntelliMar-aishub-aggregation-net
32       entrypoint:
33           - /usr/bin/java
34           - -jar
35           - imar-navstate-filter-1.2.0-SNAPSHOT.jar
36           - -i
37           - imar-ais-turn-filtered-by-region
38           - -o
39           - imar-ais-turn-filtered-by-navstate
40           - -a
41           - hard-vessel-turn/navstate-filter
42           - -b
43           - dana-s06:9092
44           - -s
45           - http://dana-s06:8081
46           - -k
47           - "FALSE"
48           - -g
49           - hard-vessel-turn-navstate-filter
50           - -n
51           - UNDER_WAY_USING_ENGINE
52
53       ...
54
55       networks:
56           IntelliMar-aishub-aggregation-net:
```

In diesem Auszug aus einer YAML-Datei kann man die drei beschriebenen Sektoren sehen, welche durch Einrückungen deutlich hervorstechen. In diesem Fall werden ein Region-Filter und ein NavState-Filter erstellt und mit den dazugehörigen Parametern belegt, zu denen unter anderem die Namen der Apache-Kafka-Topics gehören oder auch die spezifischen Werte wie „UNDER_WAY_USING_ENGINE“ für den NavState-Filter werden belegt. Unter „networks“ wird ein für IntelliMar genutztes virtuelles Netzwerk erstellt bzw. verwendet.

3.2.2.8. Das Brain Manager HMI

Der Brain Manager läuft im Hintergrund und kommuniziert mit der Außenwelt über Kafka. Es ist zwar möglich, aber nicht komfortabel, ihn über die Kafka Kommandozeilentools console-consumer und console-producer zu steuern. Deshalb wurde zu Integrationszwecken eine einfache Benutzeroberfläche erstellt: das Brain Manager HMI (Abbildung 3-24: Brain Manager HMI (Hauptseite)).

Das HMI bietet im oberen Bereich eine Befehlsleiste (in der Grafik rot umrandet). Darunter werden alle verfügbaren Brains (grün umrandet) und ihre Streaming Nodes (gelb umrandet) sowie deren Status (blau umrandet) dargestellt.

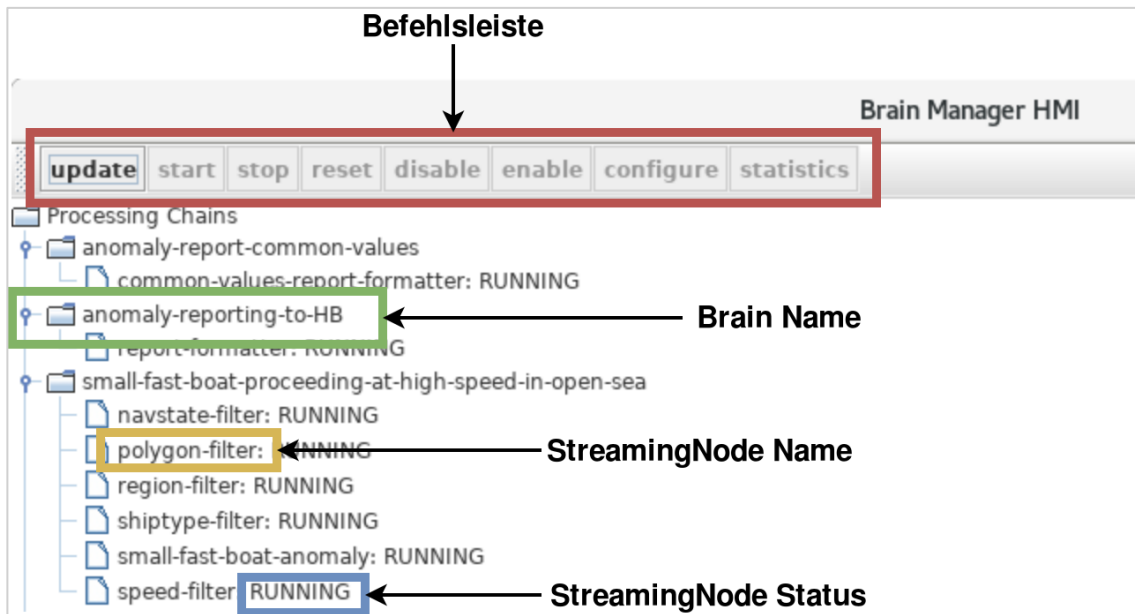


Abbildung 3-24: Brain Manager HMI (Hauptseite)

Je nach Selektion sind verschiedene Aktionen möglich.

Update

Nach einigen Aktionen ändert sich der Status der Streaming Nodes. Der Update Knopf aktualisiert alle dargestellten Inhalte.

Start / stop / reset

Wenn eine einzelner Knoten (Streaming Node oder ein Brain) selektiert ist, wird der entsprechende Befehl ausgeführt.

Disable / enable

Wenn eine einzelne Streaming Node selektiert ist, wird der entsprechende Befehl ausgeführt. Disable bewirkt, dass ein Filter deaktiviert wird, also alle Daten passieren lässt. Dies bedeutet, dass die Eingangsdaten direkt vom Eingangstopic in das Ausgangstopic durchgeleitet werden, was einem Überspringen des Knotens in der Kette – also einem „Disable“ - entspricht

Configure

Wenn eine einzelne Streaming Node selektiert ist, öffnet sich das Konfigurationsfenster (Abbildung 3-25: Brain Manager HMI (Parameter)).

Statistics

Wenn ein einzelner Knoten (Streaming Node oder ein Brain) selektiert ist, werden dessen Statistiken abgefragt und angezeigt.

Konfigurieren der Streaming Nodes

Das Konfigurationsfenster zeigt eine Übersicht über alle bekannten Parameter aller Streaming Node Typen.

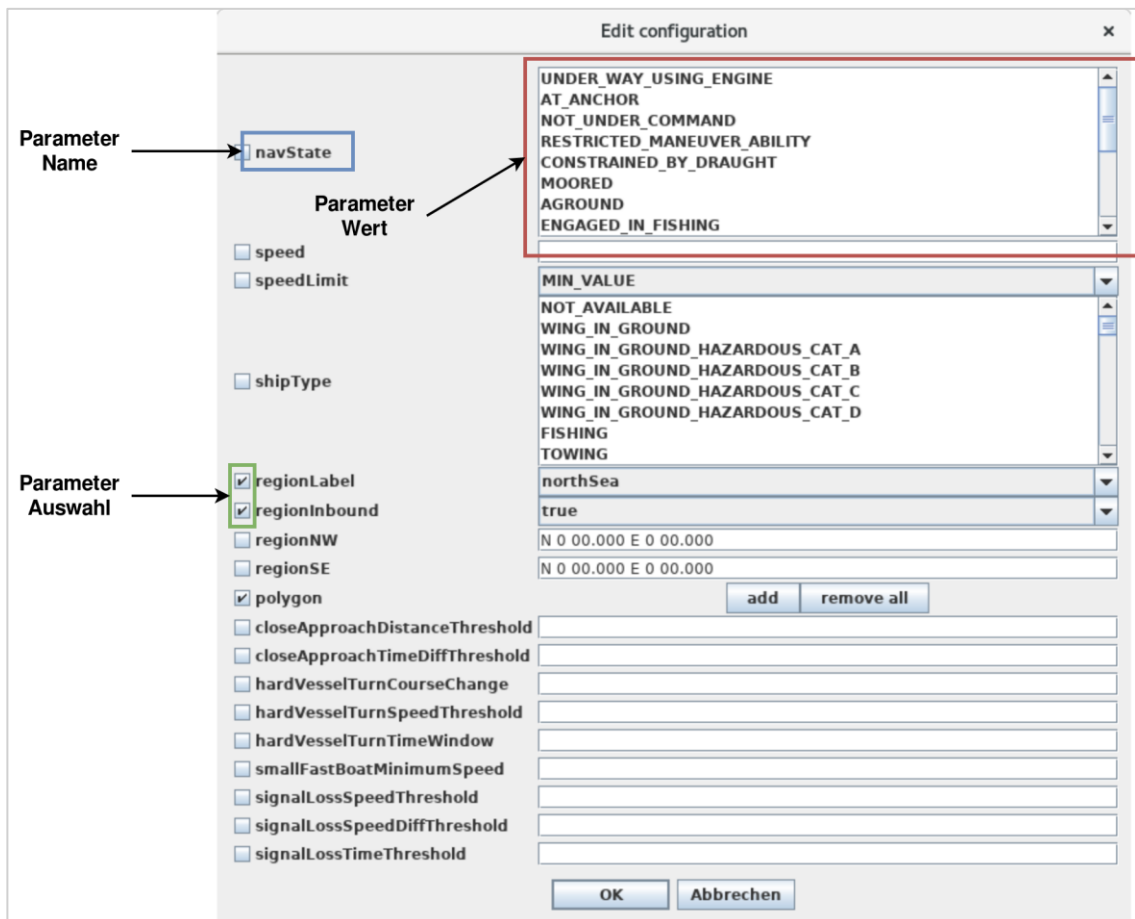


Abbildung 3-25: Brain Manager HMI (Parameter)

- **Parameter Name:** Zeigt den Name des Parameters an.
- **Parameter Auswahl:** Am Anfang sind bei den Parametern Häkchen, die für den ausgewählten Streaming Node verfügbar sind. In oberem Beispiel sind die drei Haken für den Polygon-Filter relevant. Von dieser vordefinierten Auswahl können beliebig viele Parameter abgewählt werden, welcher nicht geändert werden sollen.
- **Parameter Wert:** Bestimmte Parameter haben eine Liste an vordefinierten Werten (Enums), von denen mehrere ausgewählt werden können, z.B. navState. Bei anderen kann nur eine Auswahl genommen werden, z.B. regionLabel. Bei vielen muss ein Wert in das Feld eingetragen werden, z.B. speed. Bleibt man kurze Zeit mit dem Cursor über dem Parameter Wert Feld, so sieht man die Einheit welche der Parameter besitzt.

Wird das Fenster mit OK geschlossen, so wird die Parameteränderung an die Streaming Node weitergeleitet und vom Brain Manager gespeichert.

Eine umfassende Erläuterung der Parameter ist in Kapitel 3.2.6 zu finden.

3.2.2.9. Das Brain Manager CLI Kommandozeilentool

Zusätzlich zum Brain Manager HMI wurde ein Kommandozeilentool entwickelt, mit dessen Hilfe der Brain Manager auch von einem Skript gesteuert werden kann. Über die zusätzlichen Befehle export und import erlaubt dieses Tool es, die gesamte Konfiguration in einer Datei zu speichern um sie später wiederherzustellen.

Eine umfassende Auflistung der Parameter findet sich unter Kapitel 3.2.6.18 Das Brain Manager CLI.

3.2.3. Auswahlprozess der Anomalien für IntelliMar Services

Zu Beginn der Studie im Herbst 2018 wurde ein Auswahlprozess zu untersuchenswerten Anomalien für die Studie IntelliMar durchgeführt. Sowohl die Definition als auch die Durchführung der Auswahl fand unter Einbezug aller Verbundpartner sowie dem assoziierten Verbundpartner Bundespolizei See statt. Airbus DS in Bremen übernahm als leitender Verbundpartner die Koordination und Dokumentation des Prozesses. Daher erfolgt die Darstellung des Prozesses hier nur skizzenhaft.

In mehreren Runden zusammen mit dem assoziierten Verbundpartner Bundespolizei See entstand ein mit 90 verschiedenen Anomalien umfangreicher Katalog. Dieser wurde vom assoziierten Verbundpartner nach Kategorien wie Routenabweichung, Schiffsverhalten, Inkonsistenzen im Meldungsbild, Kommunikation und weiteren geordnet. Jede Anomalie wurde dabei mit einer für die Bundespolizei See relevanten Gewichtung versehen. (Low, Medium, High)

Die Verbundpartner haben diesen Katalog mit weiteren Annotationen wie Realisierungskomplexität, Einteilung für regelbasierte oder KI-Verfahren und Präferenz durch die einzelnen Verbundpartner erweitert, sodass anhand dieser Tabelle in mehreren Diskussionsrunden eine entsprechende Auswahl der zu realisierenden und untersuchenden Anomalien gefunden werden konnte.

Für Airbus DS in Ulm wurde folgende Auswahl festgelegt:

ID	Beschreibung	Projektpartner	Kategorie	Kundenrelevanz
S104	Small fast boat proceeding at high speed in open sea	ULM, DLR DFD	Regel	Mittel
S105	Drastic (> 30°) change of course at open seas	ULM, DLR IKN	Regel	Mittel
H207	Two or more vessels approach unusually close to each other in international waters.	ULM	Regel	Mittel
H208	The new position / travelled distance after an AIS signal loss does not appear consistent	ULM, DLR IKN	Regel	Hoch

	to the previous route (probably a hidden maneuver).			
H209	The same two vessels are sailing a route / maneuver (repeatedly within X months) that indicates a delayed drug delivery (considering the timeframe and drift corridor).	ULM	KI, mit reduzierter Komplexität	Hoch
S604	Double IMO-number in the situation picture	ULM	Regel	Hoch
S605	Double of the international call sign in the situation picture	ULM	Regel	Hoch

Tabelle 3-13: Liste der gewählten Anomalien

Die im Rahmen dieser Studie erstellte und ausgewertete Umfrage zur Bestimmung der User Requirements (siehe Kapitel 3.3.1) unterstreicht die tatsächliche Relevanz der in diesem Auswahlprozess gefundenen und realisierten Anomalien.

3.2.4. Regelbasierte Anomalien

3.2.4.1. Definition

Eine Anomalie ist eine Abweichung vom Normalen, oder auch eine Abnormität genannt. Doch nach welchen Kriterien wird eine Anomalie bewertet bzw. wann ist ein Ereignis genügend abweichend von der Normalität?

In der Mathematik kann man leicht mit bestimmten Tests nachweisen, ob ein Wert in einer Messreihe ein Ausreißer ist. Oft wird dort mit der Standardabweichung gearbeitet, also z.B. wenn ein Messpunkt weiter als zweimal die Standardabweichung vom Durchschnitt aller Punkte entfernt liegt, dann ist es eine Anomalie.

Im Alltag hingegen gibt es oft eine leichtere Bestimmung von Anomalien. Betrachtet man z.B. einen Geisterfahrer auf der Autobahn, so ist dieses Fahrzeug offensichtlich eine Anomalie, weil es anders als alle anderen in die entgegengesetzte Richtung auf der Fahrbahn fährt.

Bei IntelliMar ist die Bestimmung von Anomalien nicht ganz so einfach wie die Erkennung eines Geisterfahrers. Hier wird der Ansatz der regelbasierten Anomalien benutzt. Dies bedeutet, dass eine oder mehrere Regeln bestimmt werden und dann geprüft wird, ob sich die Schiffe an diese Regeln halten. Entspricht also ein Schiffsverhalten nicht diesen Regeln, so wird eine Anomalie bei diesem Schiff entdeckt und entsprechend gemeldet. Ein Beispiel für solch eine Regel kann eine Höchstgeschwindigkeit für Tankschiffe sein. Fährt also ein Tankschiff z.B. schneller als 30 Knoten, so ist dies anomal für solch einen Schiffstyp und wird als Anomalie detektiert. Bei IntelliMar werden nun vieler solcher Regeln aneinander gekettet, um eine möglichst genaue und mit hoher Wahrscheinlichkeit zutreffende Anomalie zu finden. Regelbasierte Detektoren sind oft parametrisiert; so ist es für das hier beschriebene Anomalieerkennungssystem von regelbasierten Anomalien kein großer Aufwand die neuen Parameter umzusetzen. Dies ist ein großer Vorteil von regelbasierten Anomalien, da es schnell geht, z.B. die oben genannte Regel von 30 Knoten auf 40 Knoten zu ändern.

Im Folgenden werden die in IntelliMar verwendeten Anomalieketten genannt und erläutert.

3.2.4.2. Drastic change of course at open seas (Hard-Vessel-Turn)

Beschreibung

Dieser Anomalietyp soll Schiffe entdecken, welche ungewöhnlich enge Kurven auf offener See fahren. Dies ist sehr anormal für Schiffe, da sie nicht plötzlich starke Richtungsänderungen durchführen, wenn sie eine Schiffsroute befahren. Allein am Umstand der Kursänderung kann jedoch nicht einfach entschieden werden, ob die Änderung ein anomales Verhalten darstellt oder nicht. Das Umfahren eines Hindernisses würde nicht als Anomalie gewertet werden können.

Für jede empfangene AIS-Nachricht wird zunächst geprüft, ob bereits eine Nachricht mit demselben Schlüssel (MMSI) empfangen wurde, und es wird ein entsprechender Eintrag in eine HashMap hinzugefügt, falls dies noch nicht der Fall war. Die HashMap verwendet als Schlüssel die MMSI und als zugeordnetem Wert die Instanz einer Klasse zur Speicherung sämtlicher Nachrichten für diese MMSI, die innerhalb eines definierten Zeitfensters liegen. Dieses Zeitfenster berechnet sich über den Zeitstempel der aktuell empfangenen Nachricht und dem konfigurierten Wert des Zeitfensters. Das bedeutet, dass die Liste der innerhalb des Zeitfensters liegenden Werte, bei jeder eingehenden Nachricht zunächst angepasst wird, so dass nur noch Nachrichten mit einem passenden Zeitstempel vorhanden sind.

Im Anschluss wird die aktuelle AIS-Nachricht mit der davor empfangenen Nachricht verglichen, um den Unterschied des Kurses zu errechnen. Um den Gesamtunterschied des Kurses innerhalb des Zeitfensters zu berechnen, werden nun alle relativen Kursunterschiede, der im Zeitfenster erhaltenen Nachrichten, addiert und mit dem konfigurierten Anomaliewert verglichen.

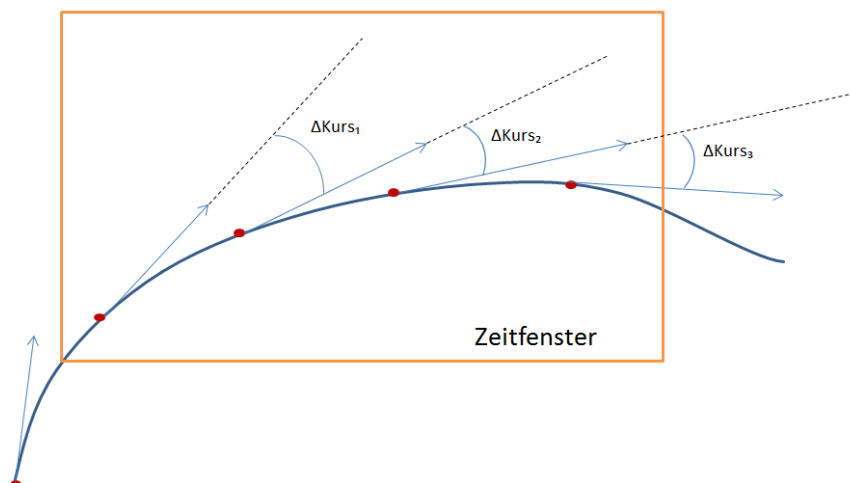


Abbildung 3-26: Turn eines Schiffes innerhalb eines Zeitfensters

Falls die Kursänderung im Zeitfenster groß genug ist, wird ein Anomaliereport erstellt, welcher die entscheidenden Merkmale der Anomalie enthält, wie die Kursänderung innerhalb

des Zeitfensters, den Start- und Endkurs (also den Kurs des ersten und letzten Updates innerhalb des Zeitfensters), die geographische Position, an der die Anomalie detektiert wurde und die Zeit der Entdeckung, sowie den Status der Anomalie. Der Status macht eine Aussage darüber, ob eine Anomalie zum ersten Mal detektiert wurde, oder ob in Folge weiterer Kursänderung in derselben Drehbewegung des Schiffes eine erneute Anomalie detektiert wird. Die folgende Abbildung zeigt die Bewegung mehrerer simulierter Schiffe und die durch den Algorithmus gefundenen Anomalien, wobei das Symbol des roten Sterns den Zeitpunkt beschreibt, bei dem eine Anomalie in einer Drehbewegung zuerst entdeckt wurde und das grüne Dreieck die Anomaliemeldungen mit Status „updated“ anzeigt.

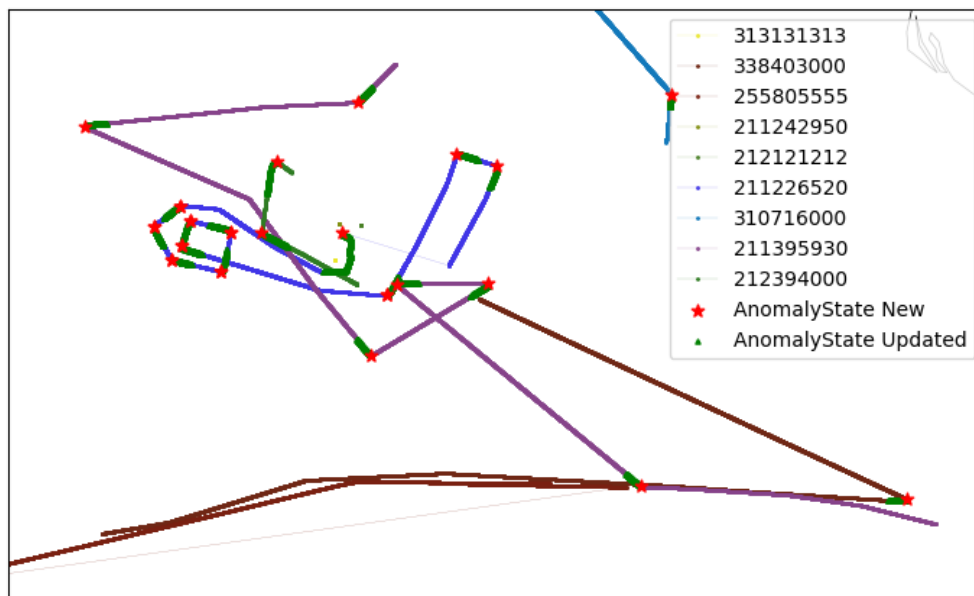


Abbildung 3-27: Veranschaulichung von mehreren gefundenen Turn-Anomalien

Des Weiteren wird ein Signifikanzwert angegeben, der sich anhand der Kursänderung berechnet. Je größer die Kursänderung, desto höher der Signifikanzwert.

Beispiel

Fahrt der ZUSE eines 30° Turns nördlich von Bremerhaven im Rahmen der Validierungskampagne I.

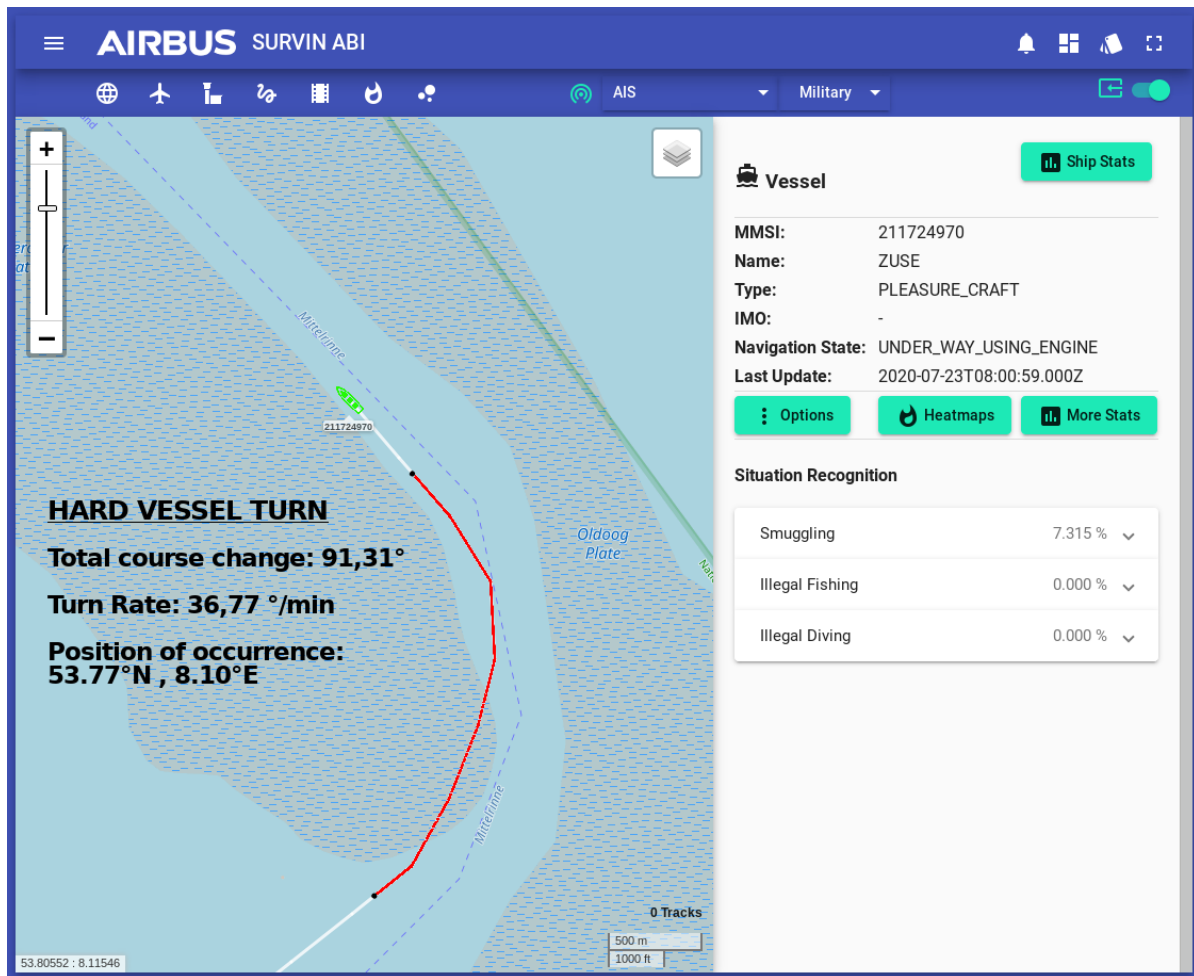


Abbildung 3-28: Ansicht des Airbus Defence and Space Ulm internen HMI: Fahrt der ZUSE eines 30° Turns nördlich von Bremerhaven

In diesem Screenshot des HMIs kann man sehen, wie das Schiff „Zuse“ im Rahmen einer Validierungskampagne eine enge Kurve fährt, um damit eine Anomalie zu provozieren. Die rote Linie zeigt an, wo die Anomalie stattgefunden hat und welchen Weg dabei das Schiff zurückgelegt hat. Der angezeigte Wert „Total Course Change“ gibt an, dass das Schiff während des Turns insgesamt über 90° seinen Kurs geändert hat, und das mit einer Kursänderungsrate von 36,77°/min (Turn Rate). Auch die Position, an der die Anomalie begonnen hat, kann aus der Grafik abgelesen werden (Position of occurrence). Auf der rechten Seite der Abbildung sind weitere Informationen des Schiffes aufgelistet, wie z.B. der Schiffstyp und der Navigationsstatus.

Rahmenbedingungen und Parameter

Bei IntelliMar ist eine drastische Kursänderung mit einer Richtungsänderung von mindestens 30° realisiert. Wobei dieser Richtwert „anomalyCourseChangeDegree“ als Parameter beliebig via Brain-Manager geändert werden kann. Außerdem ist es wichtig ein maximales Zeitintervall anzugeben, denn fährt z.B. ein Schiff eine 40° Kurve, braucht dafür aber zwei Stunden, so ist das keine Anomalie im Gegensatz dazu, wenn das Schiff diese Kurve in einer Minute absolviert. Daraus resultiert ein weiterer Parameter „timeWindowSizeSeconds“, welcher via Brain-Manager beliebig geändert

werden kann. Auch die Geschwindigkeit eines Schiffes ist von Bedeutung, daher gibt es den Parameter „speedThresholdKnotsPerSecond“, dessen Wert überschritten werden muss, um relevant für die Anomalieerkennung zu sein.

Parameter	Beschreibung
hardVesselTurnCourseChange	Eine Kursanomalie wird erkannt, wenn sich der Kurs mindestens um den konfigurierten Wert (Grad) ändert
hardVesselTurnTimeWindow	Zeitfenster in Sekunden für die Anomaliebewertung für jede MMSI
hardVesselTurnSpeedThreshold	Nur Aktualisierungen, bei denen die Geschwindigkeit des Schiffes über dem angegebenen Schwellenwert ist, werden für die Anomalieerkennung berücksichtigt

Tabelle 3-14: Parameter des Hard-Vessel-Turns und deren Bedeutung

Auslegung der Erkennungskette

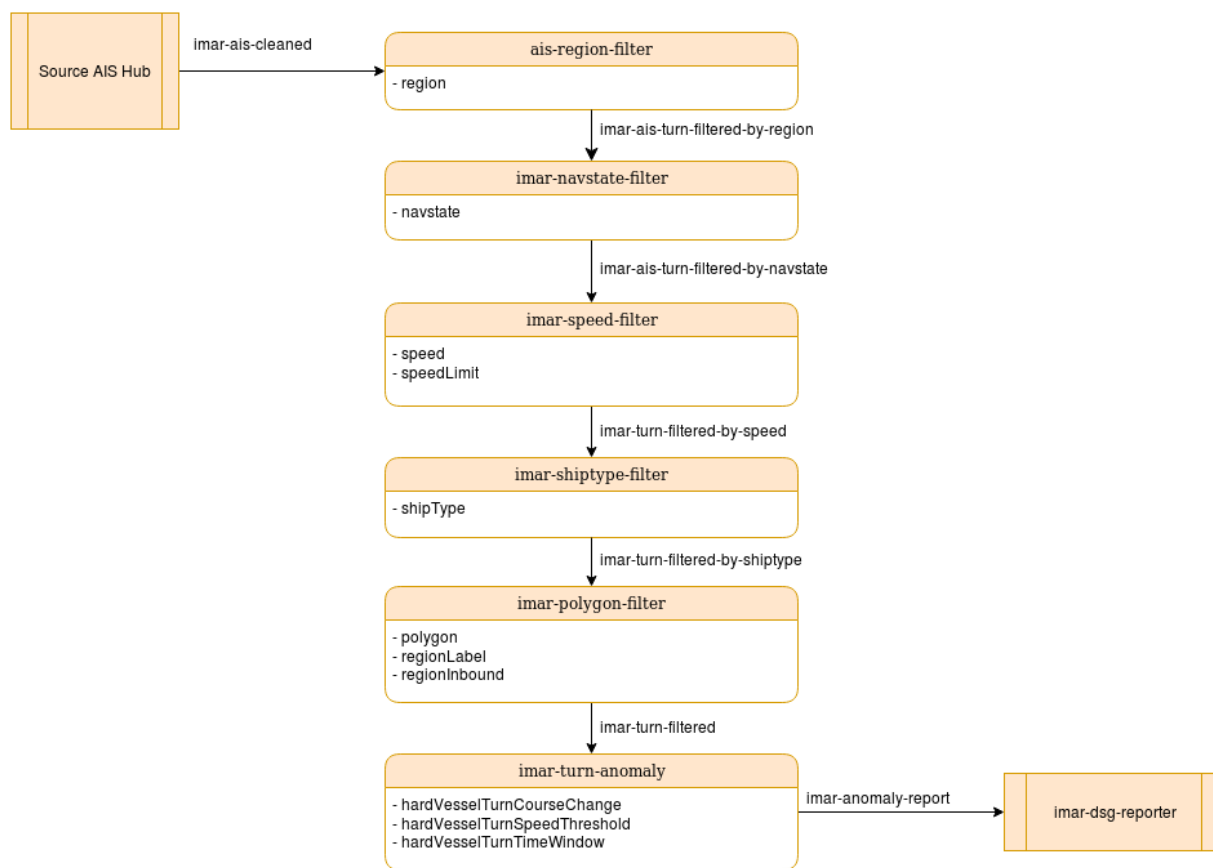


Abbildung 3-29: Übersicht der Anomaliekette „Hard-Vessel-Turn“

Legende für diese und weitere Erkennungsketten:

Source AIS Hub:	Vorprozessierte Eingangsdaten
Pfeile:	Kafkatopics mit entsprechendem Topicnamen
Abgerundete Kästen:	Streaming-Nodes mit möglichen Parametern
imar-dsg-reporter:	Datenlieferung (JSON-Report-Formatter oder DSG)

Tabelle 3-15: Legende für Erkennungsketten**3.2.4.3. The new position after an AIS signal loss does not appear consistent to the previous route (AIS-Signal-Loss)****Beschreibung**

Wie der Titel der Anomalie schon aussagt, werden bei dieser Erkennung Schiffe mit AIS-Signal Verlusten genauer unter die Lupe genommen. Hauptgründe für diese Anomalie sind, dass Schiffe während des Nichtsendens des AIS-Signals angehalten haben könnten (Schmuggel), den Kurs geändert haben (Anschleichen an einen Hafen) oder einfach unentdeckt bleiben wollen. Es wird hier geprüft, ob ein Schiff eine bestimmte Zeitspanne lang das AIS-Signal nicht mehr gesendet hat.

Für jede empfangene AIS-Nachricht wird zunächst geprüft, ob bereits eine Nachricht mit demselben Schlüssel (MMSI) empfangen wurde, und es wird ein entsprechender Eintrag in eine HashMap hinzugefügt, falls dies noch nicht der Fall war. Die HashMap verwendet als Schlüssel die MMSI und als zugeordnetem Wert die Instanz einer Klasse zur Speicherung der zuvor empfangenen Nachricht für diese MMSI. Die aktuelle Nachricht wird nun mit der vorigen Nachricht verglichen und nacheinander werden die folgenden Bedingungen überprüft:

Liegen die beiden Empfangszeitstempel der Nachrichten weiter auseinander als der konfigurierte Grenzwert? Falls nein, so liegt keine Anomalie vor, andernfalls wird die nächste Bedingung überprüft:

Ist der Geschwindigkeitsunterschied der beiden Nachrichten größer als der konfigurierte Grenzwert? Falls ja, so liegt eine Anomalie vor, falls nein, wird eine weitere Bedingung überprüft:

Ist die zurückgelegte Strecke zwischen den beiden empfangenen Schiffspositionen mit der Geschwindigkeit der zuvor erhaltenen Nachricht realistisch? Dazu wird der absolute Abstand zweier WGS84 Positionen errechnet und zwar über den Zwischenschritt der Umwandlung beider Positionen in das ECEF (Earth Centered Earth Fixed) Koordinatensystem und entsprechende Verrechnung der Abstände. Danach wird als Referenzwert ermittelt, wie weit sich das Schiff in der gegebenen Zeit mit der Startgeschwindigkeit bewegt haben müsste (bei Annahme von konstantem Kurs und Geschwindigkeit). Nun werden Erwartungswert und tatsächlich gefahrene Strecke verglichen. Um falschpositive Ergebnisse wegen leichter Kurs- oder Geschwindigkeitskorrekturen zu vermeiden, wird ein gewisser konfigurierbarer Prozentsatz an Abweichung der Strecken erlaubt und nur wenn dieser Wert überschritten wird, wird auch eine Anomalie erzeugt.

In den folgenden Abbildungen sind exemplarisch zwei Fälle zu sehen.

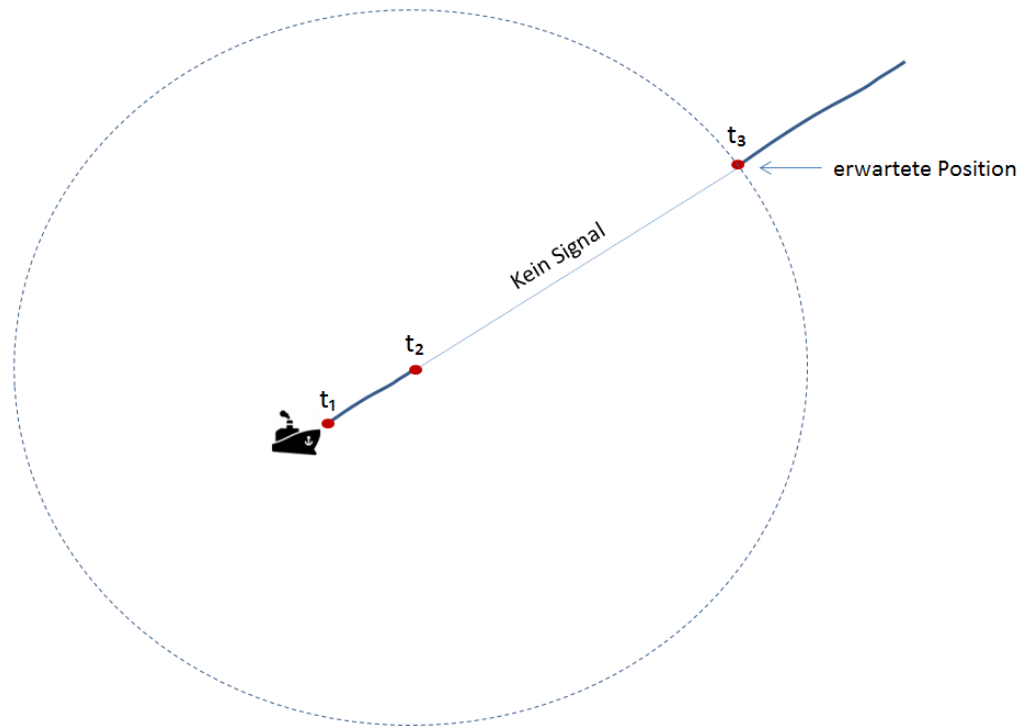


Abbildung 3-30: AIS-Signalverlust eines Schiffes bei gleichbleibendem Kurs

Auf dem oberen Bild ist ein Objekt zu sehen, für welches zwischen den Zeitpunkten t_3 und t_2 keine Nachrichten empfangen wurden, aber seine Position zum Zeitpunkt t_3 dem Erwartungswert bei konstanter Geschwindigkeit (ausgehend von t_2) entspricht.

Das folgende Bild zeigt den Fall einer Anomalie, bei der die Position des Objekts zum Zeitpunkt t_3 deutlich von der erwarteten Position abweicht.

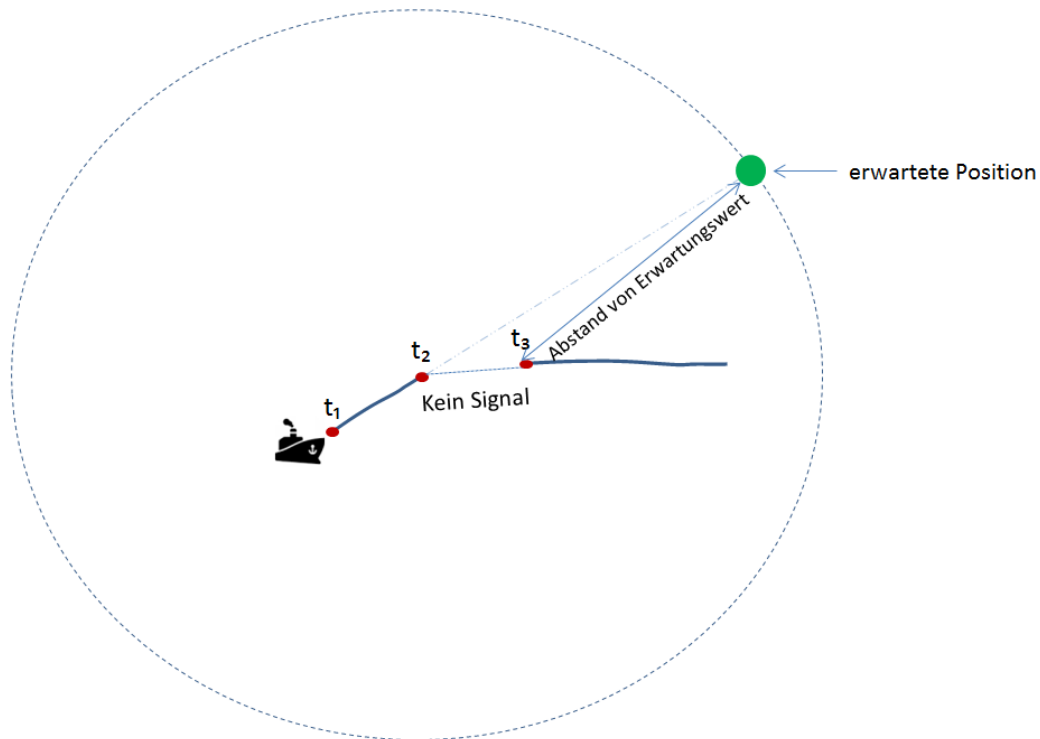


Abbildung 3-31: AIS-Signalverlust eines Schiffes mit Kursveränderung

Der Kreis deutet an, wie weit sich das Objekt mit konstanter Geschwindigkeit ausgehend von t_2 in alle Richtungen bewegt haben könnte (im Zeitraum $t_3 - t_2$).

In dem versendeten Anomaliebericht sind zahlreiche Kenngrößen aufgeführt, um die Anomalie näher zu beschreiben:

- Zeitpunkt und geographische Position der Anomalie
- Unterschied von Kurs und Geschwindigkeit der beiden Nachrichten
- zeitlicher Abstand der Meldungen
- die erwartete Distanz, die das Schiff bei linearer Bewegung hätte zurücklegen müssen (mit gegebener Startgeschwindigkeit)
- die erwartete geographische Endposition, die das Schiff bei linearer Bewegung hätte erreichen müssen
- die geographische Startposition und der Kurs des Schiffes vor der Zeitlücke

Zusätzlich wird auch ein Signifikanzwert für die Anomalie angegeben, welcher sich folgendermaßen berechnet:

Falls eine Anomalie bezüglich der Distanz besteht, wird das Maß errechnet, wie groß der Distanzunterschied zwischen erwarteter und gemessener Position im Vergleich zur gesamten erwarteten Distanz ist. Dabei werden größere Distanzen höher bewertet als niedrige Distanzen. Falls eine Anomalie bezüglich des Geschwindigkeitsunterschieds besteht, wird die Abweichung bezogen auf einen definierten hohen Geschwindigkeitswert normalisiert und geht damit in die Signifikanz ein.

Das heißt, je höher der Distanzunterschied bzw. der Geschwindigkeitsunterschied ist, desto höher auch die Signifikanz.

Beispiel

Fahrt der ZUSE mit Abschaltung des AIS-Signals und anschließender Kursänderung nördlich von Bremerhaven im Rahmen der Validierungskampagne I.

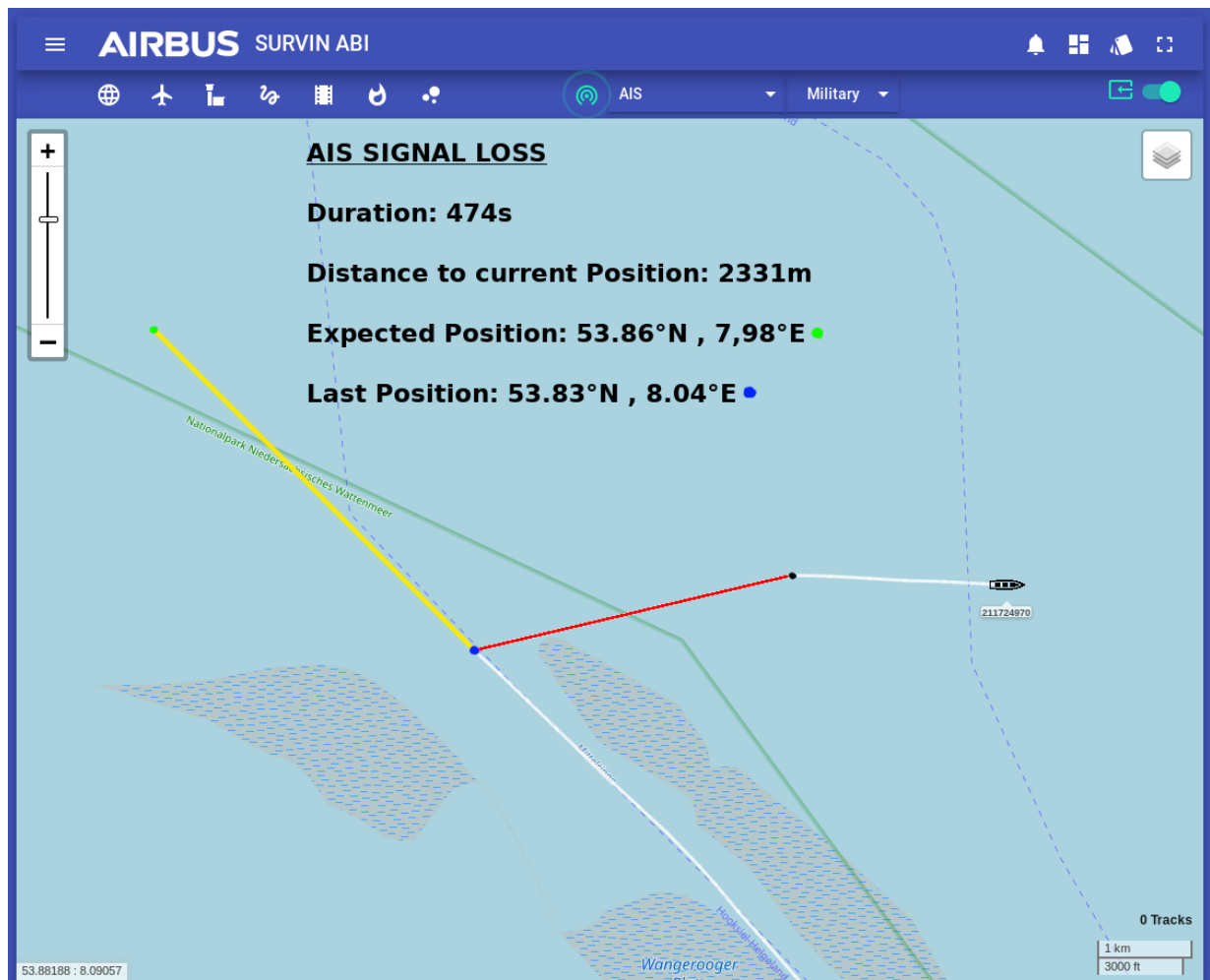


Abbildung 3-32: Ansicht des Airbus Defence and Space Ulm internen HMI.

In dieser Abbildung ist eine „Signal-Loss“-Anomalie zu sehen, bei der das Schiff nach Abschaltung des AIS-Signals (rote Linie) eine starke Richtungsänderung gemacht hat. Die gelbe Linie zeigt die erwartete bzw. berechnete Fahrtstrecke an, welche bis zur erwarteten Position (grüner Punkt) führt. Dort hätte das Boot theoretisch wieder auftauchen sollen, wenn es in den 474 Sekunden (duration) den Kurs und die Geschwindigkeit beibehalten hätte. Doch das Schiff ist 2331 Meter (Distance to current Position) von der Position vor AIS-Verlust (Last Position) entfernt. Diese Abweichungen führten zu einer Anomalieerkennung.

Rahmenbedingungen und Parameter

Bei dieser Anomalie ist vor allem wichtig, dass das Schiff eine Mindestgeschwindigkeit hat, das zwischen Signalverlust und erneutem Signalempfang eine Geschwindigkeitsdifferenz vorhanden ist und dass der Signalverlust eine Mindestdauer hat. Diese drei wichtigen Punkte können als Parameter beliebig bestimmt werden und über den Brain-Manager geändert werden.

Parameter	Beschreibung
signalLossSpeedThreshold	Nur Aktualisierungen, bei denen die Geschwindigkeit des Schiffes über dem angegebenen Schwellenwert ist, werden für die Anomalieerkennung berücksichtigt
signalLossSpeedDiffThreshold	Geschwindigkeitsdifferenzschwelle (Meter/Sekunde), um eine Anomalie zu bestimmen (wenn die Differenz zwischen der vorherigen Geschwindigkeit und der aktuellen Geschwindigkeit nach Signalverlust über dem Schwellenwert liegt)
signalLossTimeThreshold	Signalverlustschwelle in Sekunden (mindestens X Sekunden zwischen Updates gelten als Signalverlust)

Tabelle 3-16: Parameter des AIS-Signal-Loss und deren Bedeutung

Auslegung der Erkennungskette

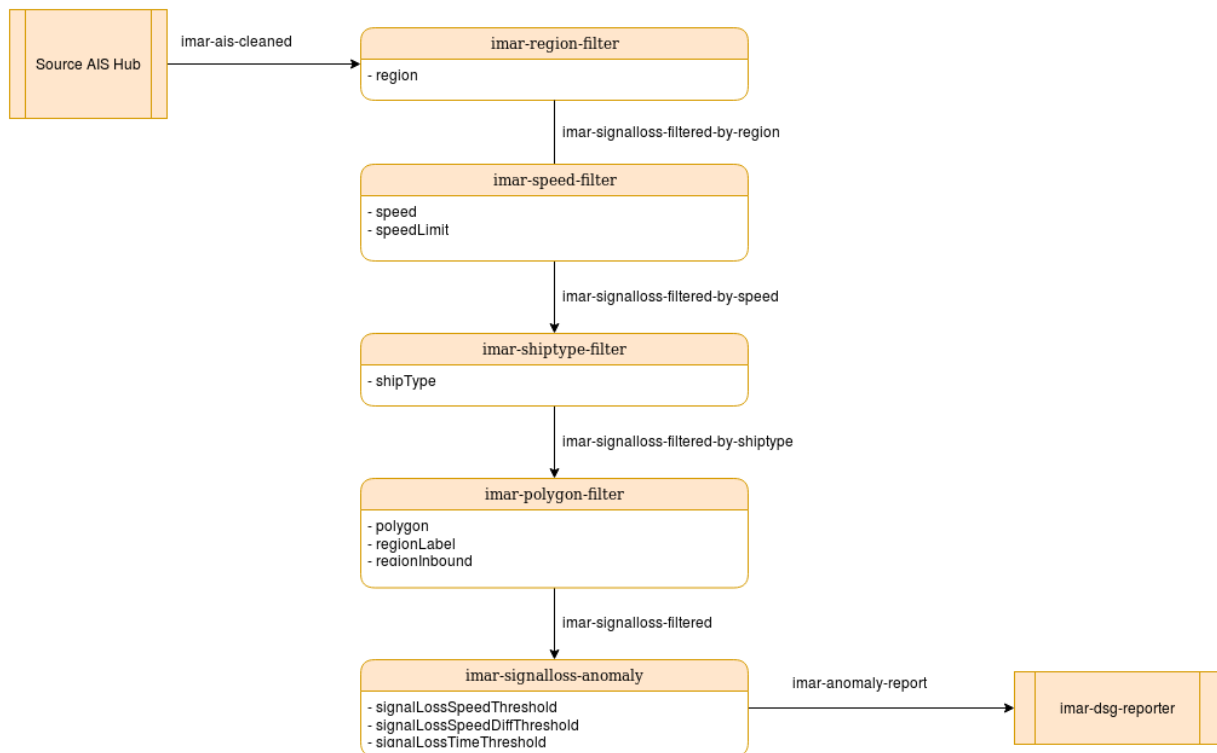


Abbildung 3-33: Übersicht der Anomaliekette „AIS-Signal-Loss“

3.2.4.4. Two or more vessels approach unusually close to each other in international waters. (Vessel-Close-Approach)

Beschreibung

Eine sehr wichtige Anomalie ist das „Rendezvous“, also wenn sich zwei oder mehrere Schiffe untypisch treffen und sehr nahe beieinander vorbei fahren oder anhalten. Das ist vor allem bei Schmuggelszenarien oder anderen Austäuschen von Waren von großer Bedeutung. Es wird in dieser Anomalieerkennung beobachtet, ob sich zwei oder mehrere Schiffe in einem bestimmten Gebiet, in einem bestimmten Zeitraum in einem gewissen Abstand annähern und eine Anomalie dadurch ausgelöst wird.

Für jede empfangene AIS-Nachricht wird zunächst geprüft, ob bereits eine Nachricht mit demselben Schlüssel (MMSI) empfangen wurde, und es wird ein entsprechender Eintrag in eine HashMap hinzugefügt, falls dies noch nicht der Fall war. Die HashMap verwendet als Schlüssel die MMSI und als zugeordnetem Wert die Instanz einer Klasse zur Speicherung der zuvor empfangenen Nachricht für diese MMSI.

Für jede empfangene AIS-Nachricht wird ein Vergleich mit der aktuellen geographischen Position aller durch die MMSI eindeutigen Objekte durchgeführt. Es wird der absolute Abstand zweier WGS84 Positionen errechnet und zwar über den Zwischenschritt der Umwandlung beider Positionen in das ECEF (Earth Centered Earth Fixed) Koordinatensystem und entsprechende Verrechnung der Abstände. Zusätzlich wird der Zeitabstand der Nachrichten ermittelt, denn Objekte mit veraltetem Zeitstempel werden nicht in die Bewertung einbezogen.

Nun wird geprüft, ob sich der Abstand und die Zeitdifferenz zwischen aktuellem update und der Vergleichsposition innerhalb der konfigurierten Grenzen befinden. Nach Durchführung der Verarbeitung für alle Vergleichspaare wird eine Liste der Objekte generiert, die sich innerhalb eines gewissen Zeitintervalls in der Nähe des aktuellen Positionsupdates befunden haben und demnach eine Anomalie darstellen.

Die Signifikanz des Ergebnisses ist ein Maß dafür, wie deutlich die Werte auf eine Anomalie hindeuten. In diesem Fall werden sowohl die Positionsabstände als auch die zeitlichen Abstände auf den maximal zulässigen Wert (der als Grenzwert für die Anomalie konfiguriert wurde) normalisiert, so dass sich jeweils eine Zahl zwischen 0 und 1 ergibt, wobei höhere Werte auf eine höhere Abweichung der verglichenen Ziele untereinander hinweisen. Die Werte werden gewichtet und multipliziert, um die absolute Signifikanz zu errechnen.

Zusätzlich zur Signifikanz werden als Kenngrößen der Anomalie der errechnete Abstand zwischen den Schiffen bzw. die aktuelle Zeit und Position angegeben.

Beispiel

Simuliertes Rendezvous zweier Schiffe, die sich in der Nordsee längs des Schiffes annähern. Das untere Schiff hält an, während das obere Schiff weiter fährt.

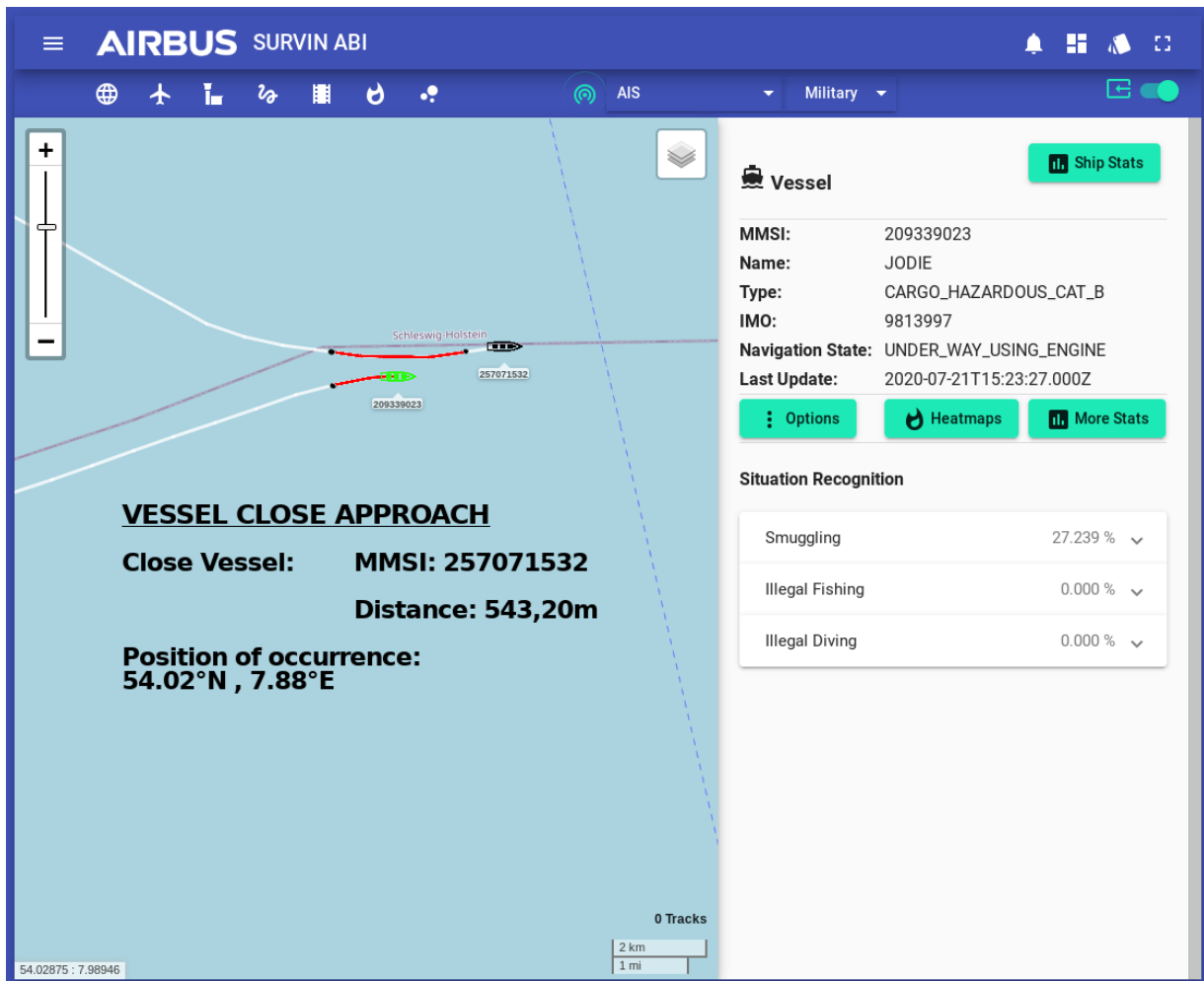


Abbildung 3-34: Ansicht des Airbus Defence and Space Ulm internen HMI.

In diesem Screenshot nähern sich zwei Schiffe an und verursachen eine Anomalie, welche stattfindet während die beiden Schiffe auf den roten Linien fahren. Man sieht auf der rechten Seite die Informationen über das untere grüne Schiff, welches im HMI ausgewählt ist. Außerdem werden die MMSI des nahen oberen Schiffes angezeigt, sowie der Abstand zwischen den beiden Schiffen und die Position des unteren Bootes zu Beginn der Anomalie.

Rahmenbedingungen und Parameter

Für eine Vessel-Close-Approach-Anomalie müssen sich die Schiffe in einer bestimmten Entfernung in einem bestimmten Zeitintervall treffen, sonst kann man dies nicht als Anomalie bewerten. Somit ergeben sich die beiden Parameter „closeApproachDistanceThreshold“ und „closeApproachTimeDiffThreshold“, welche mit dem Brain-Manager gesetzt und geändert werden können.

Parameter	Beschreibung
closeApproachDistanceThreshold	Schwellwert für die Entfernung zweier Schiffe zueinander. Für eine Anomalieerkennung muss die Distanz geringer als dieser Schwellwert sein.

closeApproachTimeDiffThreshold	Zeitfenster in Sekunden in dem die beiden Schiffe sich für eine Anomalie treffen müssen.
--------------------------------	--

Tabelle 3-17: Parameter des Vessel-Close-Approach und deren Bedeutung

Auslegung der Erkennungskette

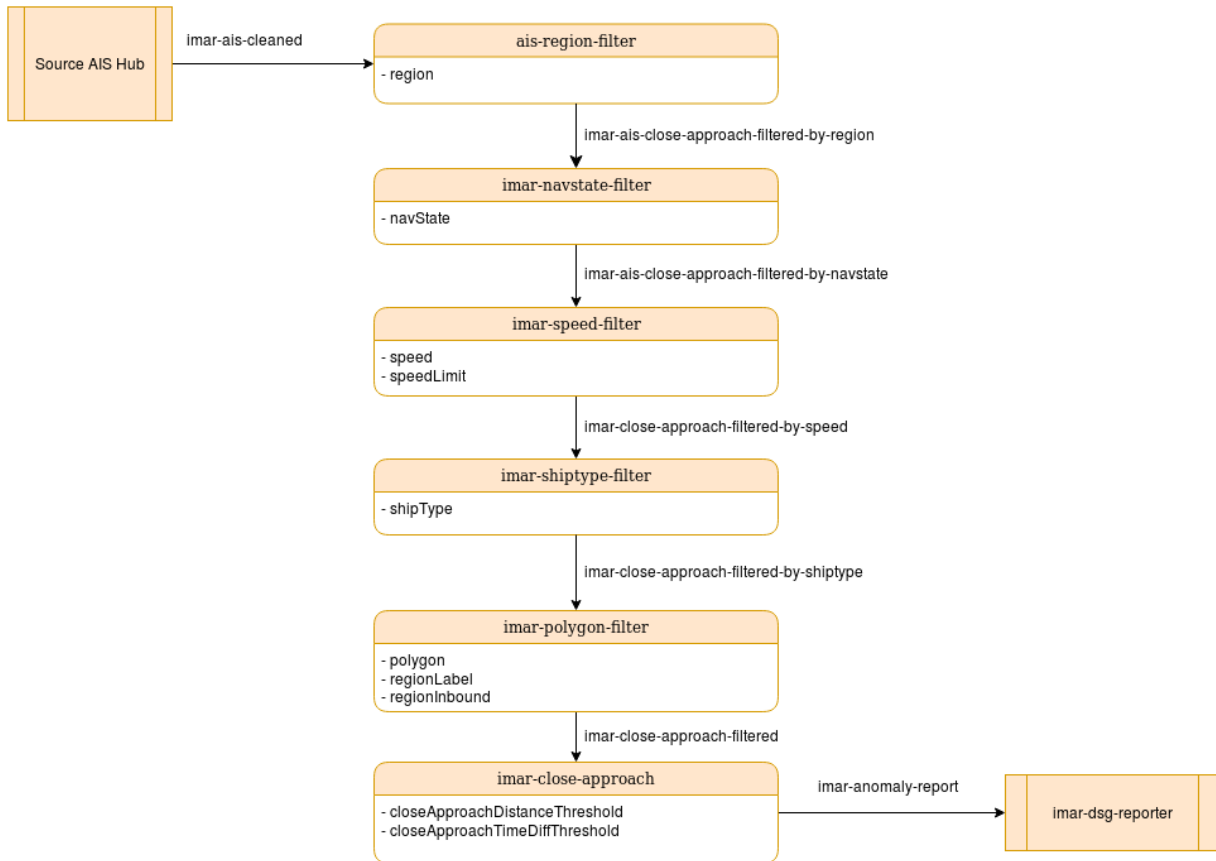


Abbildung 3-35: Übersicht der Anomaliekette „Vessel-Close-Approach“

3.2.4.5. Small fast boat proceeding at high speed in open sea (Small-Fast-Boat)

Beschreibung

Hier werden die Schiffe auf ihre Größe, Geschwindigkeit und Aufenthaltsort überprüft, denn es ist sehr anormal, wenn sich sehr kleine Schiffe auf hoher See befinden und dort sehr schnell fahren. Diese Anomalie kann man neben einer Satellitenbilddauswertung auch mit dem regelbasierten Ansatz lösen.

Für jede empfangene AIS-Nachricht wird geprüft, ob es sich bei dem betreffenden Schiff um ein „small boat“ handelt. Dazu werden die Attribute „vessel length“ und „vessel width“ mit konfigurierbaren Grenzwerten verglichen. Falls die genannten Attribute nicht verfügbar sind, wird die Prüfung mittels Attribut „Message ID“ durchgeführt und nur Objekte der Klasse B werden als „small boat“ anerkannt.

Darüber hinaus wird die gegebene Geschwindigkeit mit einem konfigurierten Grenzwert verglichen. Sind beide Bedingungen erfüllt (Objekt ist nach Definition ein „small boat“ und Geschwindigkeit ist nach Definition „fast“), so wird eine neue Anomalie erzeugt. Falls folgende Nachrichten für dieselbe MMSI immer noch die Kriterien für eine Anomalie erfüllen, so wird der Status der Anomalie auf „update“ gesetzt, um eine Folgeanomalie zu kennzeichnen. Der Signifikanzwert der Anomalie bestimmt sich aus dem Wert der Überschreitung des Geschwindigkeitsgrenzwertes (Je höher die Geschwindigkeit, desto höher der Signifikanzwert).

Beispiel

Schnelle Fahrt eines kleinen Schiffes auf offener See nördlich von Bremerhaven. Die Anomalie wird erkannt sobald das Schiff in das festgelegte Gebiet kommt, hier „Open Sea“. Hierbei handelt es sich um ein simuliertes Szenario für Validierungskampagne III.

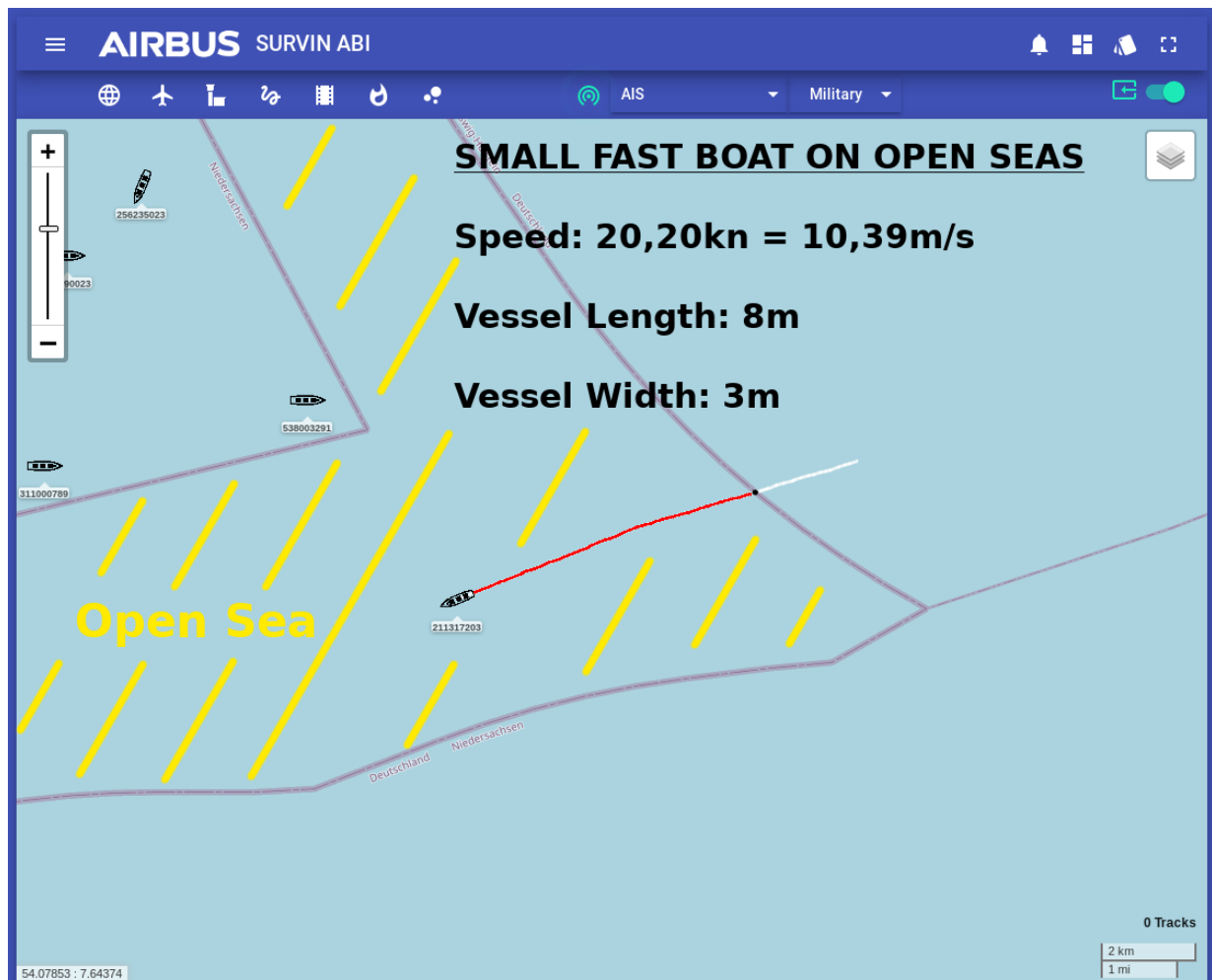


Abbildung 3-36: Ansicht des Airbus Defence and Space Ulm internen HMI.

Gelb markiert ist in diesem Szenario die offene See, welche für diese Anomalie ausschlaggebend ist. Außerdem sieht man eine rote Linie, die zeigt wo das Schiff fährt während die Anomalie entdeckt

wurde. Wichtige Informationen wie die Geschwindigkeit, die Länge und die Breite des Schiffes sind ebenfalls abzulesen.

Rahmenbedingungen und Parameter

Bei der Small-Fast-Boat-Anomalie sind nur die Größe und die Geschwindigkeit des Schiffes von Bedeutung. Das Gebiet in dem die Anomalie erkannt wird, wird im Vorhinein durch einen Region- und einen Polygon-Filter realisiert. Deshalb ergeben sich die drei Parameter „smallFastBoatMinimumSpeed“, „vesselLength“ und „vesselWidth“.

Parameter	Beschreibung
smallFastBoatMinimumSpeed	Schwellwert der Geschwindigkeit, das Schiff muss schneller als dieser Wert sein, um als „fast boat“ zu gelten.
vesselLength	Die maximale Länge des Schiffes, um als „small boat“ zu gelten.
vesselWidth	Die maximale Breite des Schiffes, um als „small boat“ zu gelten.

Tabelle 3-18: Parameter des Small-Fast-Boat und deren Bedeutung

Auslegung der Erkennungskette

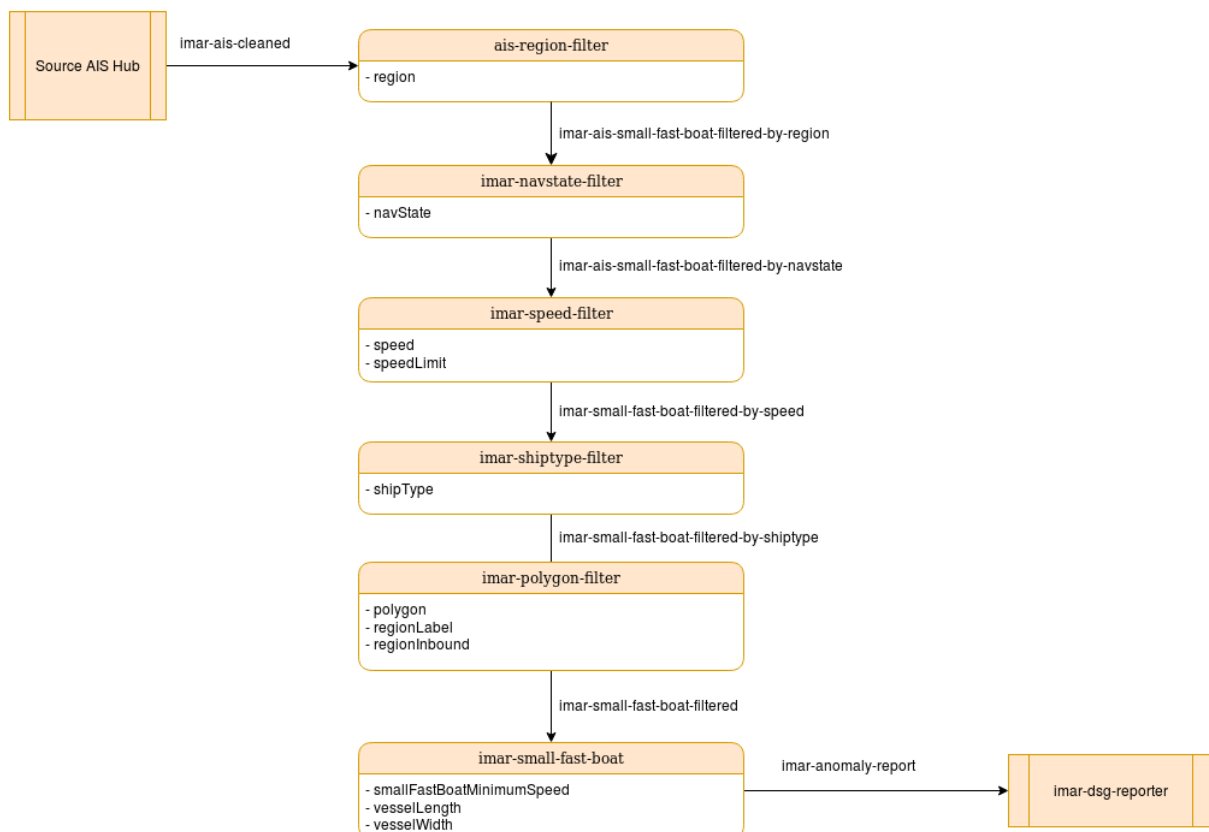


Abbildung 3-37: Übersicht der Anomaliekette „Small-Fast-Boat“

3.2.4.6. Double of the international call sign or IMO-number in the situation picture (Double-Imo-Callsign)

Beschreibung

Bei dieser Anomalie werden die statischen Daten der Schiffe betrachtet, insbesondere das Callsign und die IMO-Nummer. Die IMO-Nummer ist eine eindeutige Nummer für Schiffe, die eingeführt wurde, um Schiffe besser identifizieren zu können und so mehr Betrug verhindern zu können. Das Callsign oder auch Rufzeichen ist ebenfalls eine Kennung zur Identifizierung eines Schiffs unter Maßgabe des Herkunftslandes. Diese beiden Kennungen sind somit auch die Regeln für diese regelbasierte Anomalie. Es wird also geprüft, ob zwei oder mehrere Schiffe eine identische IMO-Nummer und/oder ein identisches Callsign haben. Sind diese Schiffe dann auch noch im selben „Situation Picture“, also gleichzeitig in einer bestimmten Region von Interesse, so wird eine „Double-IMO-Callsign“-Anomalie erkannt. Dies ist sehr nützlich, um betrügerische Schiffskapitäne zu überführen, die entweder keine solcher Kennungen besitzen oder diese manipuliert haben. Das „Situation Picture“ wird hier durch einen Region-Filter plus einem Polygon-Filter repräsentiert.

Für jede empfangene AIS-Nachricht wird zunächst geprüft, ob bereits eine Nachricht mit demselben Schlüssel (MMSI) empfangen wurde, und es wird ein entsprechender Eintrag in eine HashMap hinzugefügt, falls dies noch nicht der Fall war. Die HashMap verwendet als Schlüssel die MMSI und als zugeordnetem Wert die Instanz einer Klasse zur Speicherung der zuvor empfangenen Nachricht für diese MMSI.

Außerdem wird bei jeder empfangenen AIS-Nachricht die HashMap überprüft, ob es schon Schiffe mit einer anderen MMSI gibt, welche aber die identische IMO-Nummer oder das identische Callsign besitzt. Ist dies der Fall so wird das bereits vorhandene Schiff zusätzlich noch in einer weiteren HashMap zwischengespeichert bis die andere HashMap vollständig durchlaufen ist, denn es können beliebig viele Schiffe mit derselben IMO-Nummer oder Callsign unterwegs. Hat man alle Schiffe überprüft und die verdächtigen Boote gesammelt, so wird die Anomalie mit den entsprechenden Werten befüllt und verschickt. Daraufhin wird die HashMap, in der alle Schiffe gespeichert werden, durchlaufen und überprüft, ob noch Daten vorhanden sind, die älter als eine bestimmte Zeit sind und seit dem kein Update mehr erhalten haben. Das bedeutet, dass von Zeit zu Zeit die veralteten Schiffsdaten aus der HashMap gelöscht werden. Die andere HashMap mit den temporär gespeicherten Anomalieschiffen wird zurückgesetzt.

Die Signifikanz des Ergebnisses ist ein Maß dafür, wie deutlich die Werte auf eine Anomalie hindeuten. Bei dieser Anomalie wird die Signifikanz aus der Anzahl an AIS-Nachrichten, bei denen ein doppelter Wert auftritt, sowie der Plausibilität der IMO-Nummer bzw. Callsign berechnet.

Beispiel

Zwei Schiffe mit identischer IMO-Nummer fahren in einer Region nördlich von Bremerhaven. Die Anomalie wird erkannt, sobald sich beide Schiffe in dem „Situation Picture“ (hier blaues Viereck) befinden. Hierbei handelt es sich um eine simulierte Anomalie im Rahmen von Validierungskampagne III.

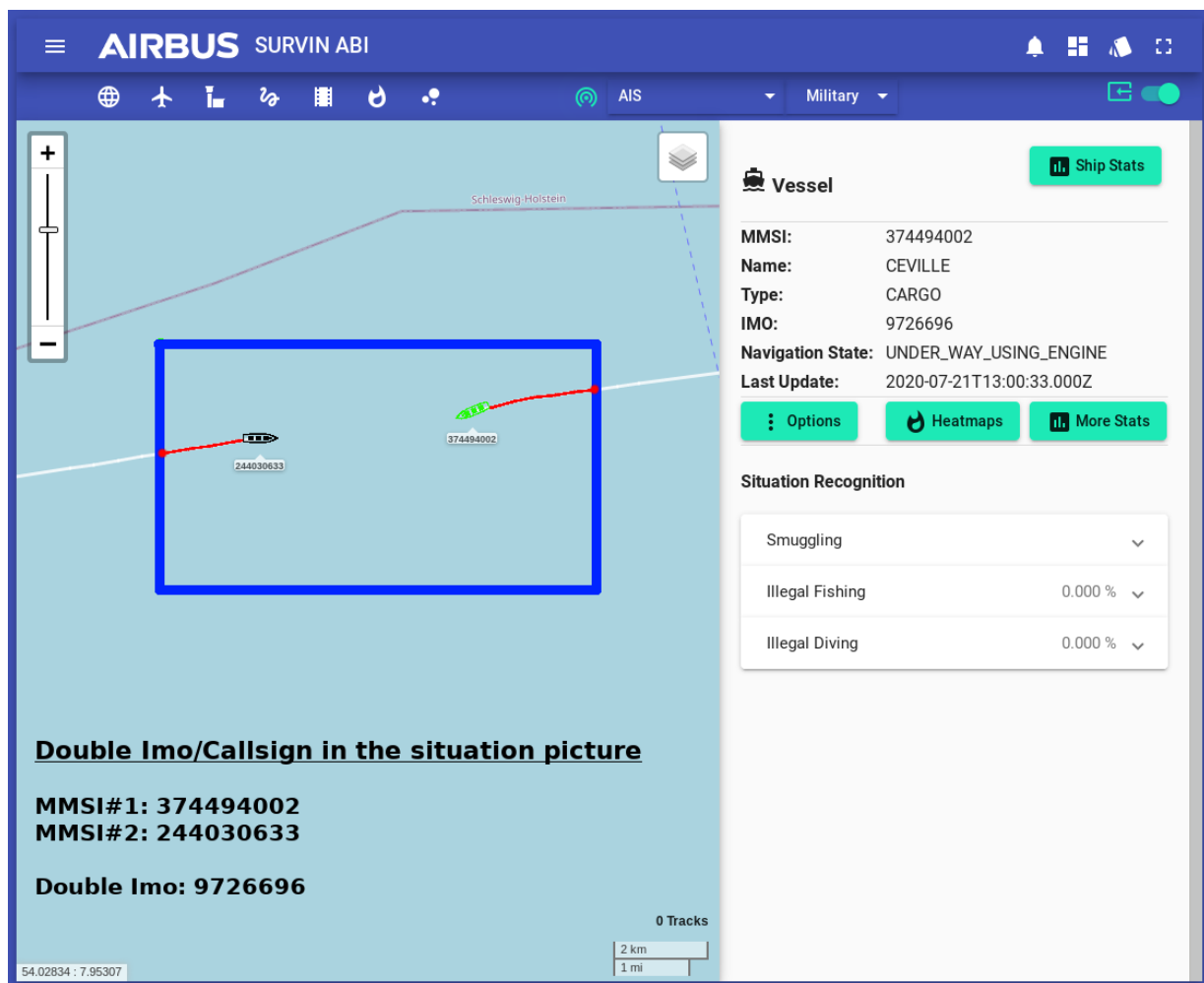


Abbildung 3-38: Ansicht des Airbus Defence and Space Ulm internen HMI.

Das blaue Viereck ist das „Situation Picture“, also die Region in der auf die Anomalie geprüft wird. Außerdem werden die MMSIs der beiden Schiffe angezeigt, sowie der Grund der Anomalie, also hier eine identische IMO-Nummer 9726696. Sobald die beiden Schiffe in das blaue Viereck hinein fahren, wird eine Anomalie ausgelöst und durch die roten Fahrtlinien markiert.

Rahmenbedingungen und Parameter

Bei dieser Anomalie sind keine Rahmenbedingungen bzw. Parameter nötig, da die Regeln für die regelbasierte Detektion eindeutig ist. Es wird immer nach der IMO-Nummer und dem Callsign geprüft und benötigt daher keine Anpassung der Parameter. Das „Situation Picture“ wird durch einen Region- bzw. Polygon-Filter realisiert und wird deshalb schon vor der Anomalieerkennung abgeprüft.

Auslegung der Erkennungskette

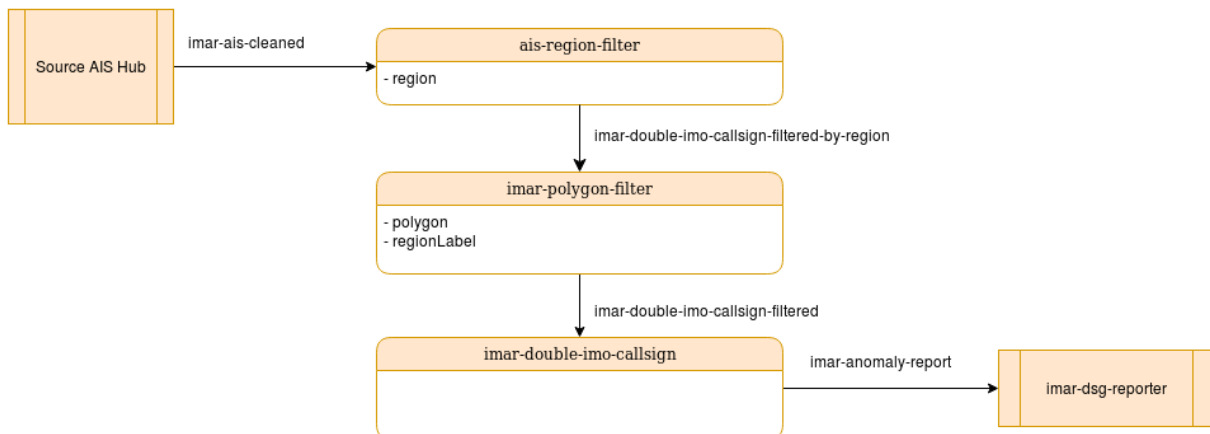


Abbildung 3-39: Übersicht der Anomaliekette „Double-Imo-Callsign“

3.2.5. Benchmarking-Unterstützung

Im Rahmen des Arbeitspaketes 1530 laut Vorhabensbeschreibung [RD-01] wurde im IntelliMar-Verbund Unterstützung zum Themenkomplex Benchmarking geleistet.

Ziel des Benchmarkings ist es, einen Vergleich bezüglich Qualität und Leistung der verschiedenen Anomalie-Erkennungsstrategien herzustellen. Die regelbasierten Anomalie-Detektoren stehen den Machine Learning und KI-basierten Detektoren gegenüber.

Die Begriffe Qualität und Leistung wurden zusammen mit dem Studienpartner DLR bewertet. Zunächst wurde festgestellt, dass ein Vergleich lediglich zwischen Anomalie-Detektoren für die gleiche Anomalie sinnvoll ist. Laut Kapitel 3.2.3 kommen hierfür folgende Anomalien in Frage.

- S104 (Small fast boat proceeding at high speed in open sea)
- S105 (Drastic (> 30°) change of course at open seas)
- H208 (The new position / travelled distance after an AIS signal loss does not appear consistent to the previous route (probably a hidden maneuver).)

Im Vorfeld zu den Benchmark-Bemühungen wurde für Streaming Nodes und somit auch für Erkennungsketten ein Log-System für folgende Parameter eingeführt und umgesetzt:

- Anzahl bearbeiteter Nachrichten bezüglich Dateneingang und Datenausgang
- Durchsatz bearbeiteter Nachrichten bezüglich Dateneingang und Datenausgang

Außerdem steht der Zeitpunkt der Feststellung einer Anomalie zur Verfügung. Dieser Zeitpunkt wird mit jener AIS-Nachricht festgelegt, die einen Detektor auslöst und somit einen neuen Anomalie-Bericht liefert. Damit ist dieser Zeitpunkt als Vergleichsindikator geeignet, da sowohl die Anomalie-Detektoren des DLR als auch die von Airbus DS Ulm nach dem Prinzip der AIS-Datenstrom-Bearbeitung funktionieren.

Den einfachsten Indikator stellt das Faktum des Erkennens einer Anomalie dar. Wenn der Zeitpunkt des Auftretens einer Anomalie bekannt ist, können die Detektoren dahingehend

verglichen werden, ob sie die Anomalie erkannt haben oder nicht- (Hit or Miss). Darüber hinaus kann geprüft werden, ob ein Detektor fehlerhaft Anomalien erkennt, die keine validen Meldefälle darstellen (False Positive).

Während der Abstimmung zu den Key Performance Indikatoren als Basis für das Benchmarking wurde schnell erkannt, dass ein Vergleich der verschiedenen Detektortypen schwierig bzw. nicht sinnvoll ist. Die Herangehensweise zum Erkennen von Anomalien aus dem Datenstrom ist bei regelbasierten und KI-basierten System grundsätzlich zu verschieden: Während regelbasierte Systeme sehr schnell im laufenden Datenstrom Anomalien erkennen können, kommt es bei KI-basierten Systemen gegebenenfalls erst nach einer Einschwingphase oder mehreren rechenintensiven Bewertungsläufen zu einem Ergebnis. Einem Zeit-Benchmark würden die KI-basierten Systeme unterlegen sein, was jedoch keine genauen Rückschlüsse auf die Brauchbarkeit der KI-Systeme erlaubt.

Ein Benchmarking bezüglich der Erkennungsrate (Hit/Miss, False Positive) zwischen regelbasierten und KI-basierten Systemen ist unter Berücksichtigung der Vorbedingungen dieser Studie ebenfalls fragwürdig. Hierfür liegen für die Studie tatsächlich zu wenige Datensätze mit bekannten Anomalien (gelabelte Datensätze) vor. Regelbasierte Systeme würden auf eine gute Erkennungsrate parametrisiert und KI-Systeme würden gegebenenfalls auf die gelabelten Datensätze übertrainiert.

Dennoch ist abschließend davon auszugehen, dass sich die Ansätze ergänzen werden. Es ist sinnvoll, beide Methoden parallel zu betreiben, um durch nachgelagerten Ergebnisvergleich der generierten Anomalien eine noch höhere Erkennungsrate bei geringerer False-Positive-Rate zu erzielen. Diese Erkenntnis wurde durch die durchgeführten Validierungskampagnen im integrierten Betrieb der verschiedenen Systeme untermauert. Während der Validierungskampagnen standen hierfür gelabelte Datensätze bereit. Zur tatsächlichen Güteprüfung standen sowohl schon bekannte Datensätze als auch den Systemen und Testern noch unbekannte Datensätze zur Verfügung.

3.2.6. Benutzerreferenz

Im Folgenden werden alle Streaming-Nodes mit ihren Kommandozeilen-Parametern aufgelistet. Es gibt Parameter, die für alle Streaming-Nodes identisch sind; solche werden vor der Referenz der speziellen Streaming-Node-Parameter vorgestellt

3.2.6.1. Gemeinsame Parameter der Streaming Nodes

Zusammenfassung

Jeder Streaming-Node ist als Java-Executable in JAR-Files (Java Archive) realisiert. Java Executables werden mit dem `java`-Kommando in Verbindung mit der `-jar`-Option gestartet. So entsteht folgendes Befehlsmuster, das für alle Streaming-Nodes identisch ist:

```
<path to java executable>/java -jar <streaming node name> [streaming node options ...]
```

Allgemeine Parameter, welche von jedem Streaming-Node benötigt werden und die jeder Streaming-Node unterstützt, können nur über die Kommandozeile bzw. YAML-File gesetzt werden.

Parameternamen werden sowohl in Langform (einleitender doppelter Bindestrich) und Kurzform (einleitender einfacher Bindestrich) unterstützt. Die jeweilige Benutzung ist wahlfrei und kann in der Kommandozeile beliebig gemischt werden.

Optionen

Option	Bedeutung
-i --input-topic	Name des Topics von dem die Eingangsdaten konsumiert werden. Üblicherweise ist das der Ausgabe-Topic-Name des vorgeschalteten Kettengliedes. Default: „kafka-in-topic“
-o --output-topic	Name des Topics in das die prozessierten Daten gespielt werden. Üblicherweise ist das der Eingabe-Topic-Name des nachgeschalteten Kettengliedes. Default: „kafka-out-topic“
-a --application-id	Interner Name der Anomaliekette und des Streaming-Nodes Default: „kafka-streaming-node“
-b --bootstrap-servers	Kommaseparierte Liste der IP-Adressen der Bootstrap-Servers Default: „localhost:1111“
-s --schema-registry	URL der Schema-Registry Default: „http://localhost:1111“
-k --skip-option	Option, ob der Streaming-Node übersprungen werden soll Default: false
-g -group	Kafka Consumer Group ID Default: „configuration-group“

Tabelle 3-19: Liste der allgemeinen Optionen und Parameter der Streaming-Nodes

Beispiel

```
/usr/bin/java -jar imar-navstate-filter-1.2.0-SNAPSHOT.jar -i imar-ais-turn-filtered-by-region -o imar-ais-turn-filtered-by-navstate -a hard-vessel-turn/navstate-filter -b dana-s06:9092 -schema-registry http://dana-s06:8081 -k "FALSE" -g hard-vessel-turn-navstate-filter -n UNDER_WAY_USING_ENGINE
```

3.2.6.2. imar-ais-aggregator

Zusammenfassung

Streaming-Node, der je MMSI die statischen Daten mit den Positionsdaten zusammenführt.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
imar-ais-aggregator-1.1.0-SNAPSHOT.jar -i imar-ais-hub -o imar-aishub-aggregated -a imar-aggregation -b dana-s06:9092 -s http://dana-s06:8081 -g Aishub-aggregator
```

3.2.6.3. imar-ais-instream

Zusammenfassung

Streaming-Node, der die erhaltenen NMEA-Sätze dekodiert und auf das interne AVRO-Schema mappt.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
ais-in-stream-1.1.0-SNAPSHOT.jar -i imar-ais-timed-and-filtered-nmea-sentences-from-aishub -o imar-ais-hub -a nmea-to-ais-converter-for-aishub-aggregation -b dana-s06:9092 -s http://dana-s06:8081 -g Aishub-instream
```

3.2.6.4. imar-ais-mmsi-partitioner

Zusammenfassung

Streaming-Node zum Empfangen von AIS-Daten, welche als NMEA-Sätze dargestellt sind. Sie werden nach AIS-Transponderklasse vorgefiltert und gegebenenfalls auf mehrere Kafka-Partitionen verteilt

Optionen

Option	Bedeutung und Wertebereich
-d --sessionId	Name der Session, zu der die eingespielten Daten gehören. Default: "AISHUB-LIVE"
-c --class	Gibt an, nach welchen AIS-MessageIDs gefiltert werden soll. Mögliche Werte: „ALL“ (AIS-MessageIDs 1-28), „SUPPORTED“ (1, 2, 3, 5, 18, 19, 24), „A“ (1, 2, 3, 5), „B“ (18, 19, 24), „NONE“ () Default: „SUPPORTED“

Tabelle 3-20: Liste der Optionen für imar-ais-mmsi-partitioner

Beispiel

```
/usr/bin/java -jar ais-mmsi-partitioner-1.0.0-SNAPSHOT.jar -i imar-ais-timed-nmea-sentences-from-ais-hub -o imar-ais-timed-and-filtered-nmea-sentences-from-aishub -a ais-partitioner-for-aishub-aggregation -b dana-s06:9092 -s http://dana-s06:8081 -c SUPPORTED -d AISHUB-LIVE -g Aishub-partitioner
```

3.2.6.5. imar-ais-signal-loss

Zusammenfassung

Streaming-Node zur Anomaliedetektion der Anomalie „AIS-Signal-Loss“.

Optionen

Option	Bedeutung und Wertebereich
BM: signalLossSpeedThreshold	Nur Aktualisierungen, bei denen die Geschwindigkeit des Schiffes über dem angegebenen Schwellenwert ist, werden für die Anomalieerkennung berücksichtigt. Default: 1.0
BM: signalLossSpeedDiffThreshold	Geschwindigkeitsdifferenzschwelle (Meter/Sekunde), um eine Anomalie zu bestimmen (wenn die Differenz zwischen der vorherigen Geschwindigkeit und der aktuellen Geschwindigkeit nach Signalverlust über dem Schwellenwert liegt) Default: 4.0
BM: signalLossTimeThreshold	Signalverlustschwelle in Sekunden (mindestens X Sekunden zwischen Updates gelten als Signalverlust) Default: 600.0

Tabelle 3-21: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-ais-signal-loss

Beispiel

```
imar-signal-loss-1.2.0-SNAPSHOT.jar -i imar-ais-cleaned -o imar-anomaly-report -a signal-loss/signal-loss-anomaly -b dana-s06:9092 -s http://dana-s06: -g signal-loss-ais-anomaly
```

3.2.6.6. imar-ais-tracking

Zusammenfassung

Streaming-Node, der die AIS Daten hauptsächlich bezogen auf die kinematischen Daten säubert und glättet.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
/usr/bin/java -jar imar-ais-tracking-1.3.2-SNAPSHOT.jar -i imar-aishub-aggregated -o imar-ais-cleaned -a imar-tracker/imar-ais-tracker -b dana-s06:9092 -s http://dana-s06:8081 -g Aishub-live-tracker
```

3.2.6.7. imar-common-values-report-formatter

Zusammenfassung

Angepasster JSON-Report-Formatter, der die SI-Einheiten in Einheiten konvertiert, die im maritimen Umfeld üblich sind.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
imar-common-values-report-formatter-1.2.0-SNAPSHOT.jar -i imar-anomaly-report -o imar-anomaly-report-common-values -a common-values/common-values-report-formatter -b dana-s06:9092 -s http://dana-s06:8081 -g common-values-report-formatter
```

3.2.6.8. imar-double-imo-callsign

Zusammenfassung

Streaming-Node zur Anomaliedetektion der Anomalie „Double-Imo-Callsign“.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
imar-duplicated-imo-callsign-1.0-SNAPSHOT.jar -i imar-aishub-aggregated -o imar-anomaly-report -a double-imo-callsign/double-imo-callsign-anomaly -b dana-s06:9092 -s http://dana-s06:8081 -g double-imo-callsign-anomaly
```

3.2.6.9. imar-hard-vessel-turn

Zusammenfassung

Streaming-Node zur Anomaliedetektion der Anomalie „Hard-Vessel-Turn“.

Optionen

Option	Bedeutung und Wertebereich
BM: hardVesselTurnCourseChange	Eine Kursanomalie wird erkannt, wenn sich der Kurs mindestens um den konfigurierten Wert (Grad) ändert.

	Default: 30.0
BM: hardVesselTurnTimeWindow	Zeitfenster in Sekunden für die Anomaliebewertung für jede MMSI Default: 180.0
BM: hardVesselTurnSpeedThreshold	Nur Aktualisierungen, bei denen die Geschwindigkeit des Schiffes über dem angegebenen Schwellenwert ist, werden für die Anomalieerkennung berücksichtigt. Default: 5.0

Tabelle 3-22: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-hard-vessel-turn

Beispiel

```
imar-turn-anomaly-1.4.0-SNAPSHOT.jar -i imar-ais-cleaned-o imar-anomaly-report -a turn/turn-anomaly -b dana-s06:9092 -s http://dana-s06:8081 -g hard-vessel-turn-turn-anomaly
```

3.2.6.10. imar-navstate-filter

Zusammenfassung

Streaming-Node, der die Daten nach dem Navigationsstatus filtert, dabei kann eine Liste an Navigationsstatus als Parameter gesetzt werden.

Optionen

Option	Bedeutung und Wertebereich
-n	Liste an Navigationsstatus, die die Schiffe haben dürfen, um nicht aussortiert zu werden.
BM: navstate	Default: "UNDER_WAY_USING_ENGINE"

Tabelle 3-23: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-navstate-filter

Beispiel

```
imar-navstate-filter-1.2.0-SNAPSHOT.jar -i imar-data-filtered-by-region -o imar- - data-filtered-by-navstate -a navstate/navstate-filter -b dana-s06:9092 -s http://dana-s06: -g navstate-filter -n UNDER_WAY_USING_ENGINE, SAILING, ENGAGED_IN_FISHING
```

Dieser Streaming-Node filtert die durchlaufenden Schiffsdaten und leitet nur die Daten zu Schiffen weiter, welche den Navigationsstatus „UNDER_WAY_USING_ENGINE“, „SAILING“ oder „ENGAGED_IN_FISHING“ haben.

3.2.6.11. imar-polygon-filter

Zusammenfassung

Streaming-Node, der die Daten nach der Region filtert in der das Schiff fährt, dabei kann ein beliebiges n-Eck als Region definiert und dieses Polygon als Parameter gesetzt werden.

Optionen

Option	Bedeutung und Wertebereich
-p --polygon BM: polygon	Angabe von beliebig vielen Koordinaten, aus denen ein Polygon aufgespannt wird. Pro Koordinate muss je ein Längen- und ein Breitengrad angegeben werden. Default: " 60.0/40.0/-5.0/20.0"
BM: regionLabel	Parameter zur Auswahl vordefinierter Regionen. Default: -, da nur im Brain-Manager setzbar.
BM: regionInbound	Boolean-Parameter, der angibt ob das Innere oder das Äußere des Polygons als Akzeptanzbereich genutzt wird. Mögliche Werte sind „True“ (Innere des Polygons) und „FALSE“ (Äußere des Polygons). Default: -, da nur im Brain-Manager setzbar

Tabelle 3-24: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-polygon-filter

Beispiel

```
imar-polygon-filter-1.1.0-SNAPSHOT.jar -i imar-data-filtered-by-shiptype -o imar-data-filtered -a polygon/polygon-filter -b dana-s06:9092 -s http://dana-s06:8081 -g polygon-filter -p54.329338/7.783813,54.338947/7.624512,54.335744/8.151855,54.708755/7.937622,55.100373/8.047485,55.485079/7.717896,56.022948/7.734375,56.607885/7.772827,57.258252/8.33313,57.926517/10.700684,57.15412/11.491699,57.379861/11.469727,58.739706/10.612793,58.089493/8.822021,57.756938/7.03125,58.390197/5.383301,59.366794/4.493408,60.818138/4.21875,60.673179/-0.384521,58.043004/-3.164063,57.692406/-1.538086,56.231139/-2.087402,54.838664/-0.856934,53.265213/0.74707,52.57635/2.087402,51.055207/1.713867,52.254709/4.064941,53.644638/4.899902,53.991625/7.712402
```

Dieser Streaming-Node filtert die durchlaufenden Schiffsdaten und leitet nur die Daten zu Schiffen weiter, welche innerhalb dieses Polygons liegen.

3.2.6.12. imar-region-filter

Zusammenfassung

Streaming-Node, der die Daten nach der Region filtert in der das Schiff fährt. Dabei kann ein Viereck als Region definiert werden.

Optionen

Option	Bedeutung und Wertebereich
-r --region BM: region	Angabe von zwei Koordinaten, aus denen ein Quadrat aufgespannt wird. Pro Koordinate muss je ein Längen- und ein Breitengrad angegeben werden. Default: "60.0/40.0/-5.0/20.0"

Tabelle 3-25: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-region-filter

Beispiel

```
imar-region-filter-1.2.0-SNAPSHOT.jar -i imar-ais-cleaned -o imar-data-filtered-by-region -a region/region-filter -b dana-s06:9092 -s http://dana-s06:8081 -g region-filter -r 60.0/40.0/10.0/20.0
```

Dieser Streaming-Node filtert die durchlaufenden Schiffsdaten und leitet nur die Daten zu Schiffen weiter, die sich südlich vom 60ten Breitengrad, östlich vom 40ten Längengrad, nördlich vom zehnten Breitengrad und westlich vom 20ten Längengrad befinden.

3.2.6.13. imar-report-formatter

Zusammenfassung

Streaming-Node zur Konvertierung des Anomalieberichts vom AVRO-Schema AisAnomalyReport in das JSON-Schema.

Optionen

Keine zusätzlichen Optionen.

Beispiel

```
imar-report-formatter-1.3.0-SNAPSHOT.jar -i imar-anomaly-report -o imar-ais-anomaly-report-for-hb -a anomaly-reporting-to-HB/report-formatter -b dana-s06:9092 -s http://dana-s06:8081
```

3.2.6.14. imar-shiptype-filter

Zusammenfassung

Streaming-Node, der die Daten nach dem Schiffstyp filtert, dabei kann eine Liste an Schiffstypen als Parameter gesetzt werden.

Optionen

Option	Bedeutung und Wertebereich
-t -- ship-type BM: shipType	Liste an Schiffstypen, die die Schiffe haben dürfen, um nicht aussortiert zu werden. Default: „TANKER“

Tabelle 3-26: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-shiptype-filter

Beispiel

```
imar-shiptype-filter-1.2.0-SNAPSHOT.jar -i imar-data-filtered-by-speed -o imar-data-filtered-by-shiptype -a shiptype/shiptype-filter -b dana-s06:9092 -s http://dana-s06:8081 -g shiptype-filter -t TANKER, CARGO, PASSENGER
```

Dieser Streaming-Node filtert die durchlaufenden Schiffsdaten und leitet nur die Daten zu Schiffen weiter, welche vom Schiffstyp „TANKER“, „CARGO“ oder „PASSENGER“ sind.

3.2.6.15. imar-small-fast-boat

Zusammenfassung

Streaming-Node zur Anomaliedetektion der Anomalie „Small-Fast-Boat“.

Optionen

Option	Bedeutung und Wertebereich
BM: smallFastBoatMinimumSpeed	Schwellwert der Geschwindigkeit in Meter pro Sekunde, das Schiff muss schneller als dieser Wert sein, um als „fast boat“ zu gelten. Default: 10.0
BM: vesselLength	Die maximale Länge des Schiffes in Meter, um als „small boat“ zu gelten. Default: 20.0
BM: vesselWidth	Die maximale Breite des Schiffes in Meter, um als „small boat“ zu gelten. Default: 10.0

Tabelle 3-27: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-small-fast-boat

Beispiel

```
imar-small-fast-boat-1.0.0-SNAPSHOT.jar -i imar-small-fast-boat-filtered -o imar-anomaly-report -a small-fast-boat-proceeding-at-high-speed-in-open-sea/small-fast-boat-anomaly -b dana-s06:9092 -s http://dana-s06:8081 -g small-fast-boatproceeding-at-high-speed-in-open-sea-anomaly
```

3.2.6.16. imar-speed-filter

Zusammenfassung

Streaming-Node, der die Daten nach Geschwindigkeit filtert, dabei kann man eine Unter- bzw. Obergrenze setzen.

Optionen

Option	Bedeutung und Wertebereich
-l --speedLimit BM: speedLimit	Gibt an, ob es eine Geschwindigkeitsober- oder Geschwindigkeitsuntergrenze sein soll. Mögliche Werte: „MAX“ oder „MIN“. Default: „MIN“
-c --speed BM: speed	Wert der Geschwindigkeitsgrenze in kn (Knoten) an. Default: 2.5

Tabelle 3-28: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-speed-filter

Beispiel

```
imar-speed-filter-1.2.0-SNAPSHOT.jar -i imar-data-filtered-by-navstate -o imar-data-filtered-by-speed -a speed/speed-filter -b dana-s06:9092 -s http://dana-s06:8081 -g speed-filter -c 2.5 -l MAX
```

Dieser Streaming-Node filtert die durchlaufenden Schiffsdaten und leitet nur die Daten zu Schiffen weiter, welche langsamer als 2,5 Knoten fahren.

3.2.6.17. imar-vessel-close-approach

Zusammenfassung

Streaming-Node zur Anomaliedetektion der Anomalie „Vessel-Close-Approach“.

Optionen

Option	Bedeutung und Wertebereich
BM: closeApproachDistanceThreshold	Schwellwert für die Entfernung in Meter zweier Schiffe zueinander. Für eine Anomalieerkennung muss die Distanz geringer als dieser Schwellwert sein. Default: 500.0
BM: closeApproachTimeDiffThreshold	Zeitfenster in Sekunden in dem die beiden Schiffe sich für eine Anomalie treffen müssen. Default: 30.0

Tabelle 3-29: Liste speziellen Optionen und Parameter, die vom Brain Manager gesetzt werden können (gekennzeichnet mit „BM:“) für imar-vessel-close-approach

Beispiel

```
imar-close-approach-1.1.0-SNAPSHOT.jar -i imar-ais-cleaned -o imar-anomaly-report -a close-approach/close-approach-anomaly -b dana-s06:9092 -s http://dana-s06:8081 -g close-approach-anomaly
```


3.2.6.18. Das Brain Manager CLI

Zusammenfassung

Zusätzlich zum Brain Manager HMI wurde ein Kommandozeilentool entwickelt, mit dessen Hilfe der Brain Manager auch von einem Skript gesteuert werden kann. Über die zusätzlichen Befehle export und import erlaubt dieses Tool es, die gesamte Konfiguration in einer Datei zu speichern, um sie später wiederherzustellen. Das BrainManagerCLI (Command Line Interface) dient daher als skriptfähiger Ersatz für das BrainManager HMI und kann zur automatisierten Konfiguration von Erkennungsketten und einzelnen Streaming Nodes eingesetzt werden.

Optionen

Option	Argument	Bedeutung und Wertebereich des Arguments
-a --address	X	Objekt für das der Befehl in <code>-c/--command</code> bestimmt ist. Streaming Node: <code><chain>.<node></code> Kette: <code><chain>.*</code>
-b --bootstrap-servers	X	Kommaseparierte Liste der zu kontaktierenden Kafka Bootstrap-Server
-c --command	X	Das auszuführende Kommando. (siehe Tabelle 3-2: Brain Manager Benutzerkommandos Fehler! Verweisquelle konnte nicht gefunden werden.)
--export		Status und Konfiguration in Datei speichern. Der Dateiname muss als zusätzlicher Parameter path angegeben werden. Bsp: <code>path:/my/nice/file.txt</code>
-i --input-topic	X	Ergebnistopic des Brainmanagers Default: <code>imar-user-response</code>
-k --skip-option		Aus Kompatibilitätsgründen vorhanden, jedoch nutzlos.
-o --output-topic	X	Befehlstopic des Brainmanager Default: <code>imar-user-command</code>
--prefix	X	Ausgabepräfix zur Komponentenmarkierung beim LIST-Kommando
-s --schema-registry-url	X	URL der zu verwendenden Kafka-Schema-Registry. Bsp: <code>http://localhost:8081/</code>
-v --verbose		Erweiterte Statusausgabe beim LIST-Kommando
<key>[:<comma separated list of value>]	X	Kommandoargumente: nach der Liste der Optionen kann eine leerzeichengetrennte Liste von Kommandoargumenten angehängt werden. Ein Argument ist ein mit einem Doppelpunkt getrenntes Schlüssel-Wert-Paar
<ohne Option>		Ausgabe der Hilfeseite.

Tabelle 3-30: Liste aller Optionen für BrainManagerCLI

Beispiel

BrainManagerCLI ...

Gibt die Hilfeseite aus.

```
Brainmanager -import path:/tmp/nodeconfig
```

Liest eine in /tmp/nodeconfig gesicherte Konfiguration wieder ein

```
Brainmanager -a chain1.* -c LIST
```

Zeigt alle Elemente der Streaming-Node-Kette chain1

```
BrainManagerCLI -a chain1.node2 -c SET _CONFIG navState :AT_ANCHOR
```

Setzt in Streaming Node node2 der Kette chain1 den parameter navstate auf AT_ANCHOR

3.2.7. Entwicklung zu ML-basierten Detektoren

3.2.7.1. Einführung

Im Zuge der Studie IntelliMar wurde die Masterarbeit mit dem Titel „Maritime Situationserkennung mittels dynamischer Bayes'scher Netze“ erstellt (siehe Kapitel 3.9). Der folgende Teil des Berichts basiert auf dieser Masterarbeit. Dabei wird auf die Anwendung von dynamischen Bayes'schen Netzen für die Situationserkennung im maritimen Umfeld eingegangen.

Ein dynamisches Bayes'sches Netz (DBN) ist ein probabilistisches graphisches Modell, welches eine Methodik zum Schließen unter Unsicherheit über die Zeit darstellt. Mit diesem Modell kann eine Aussage über die Existenz der modellierten Situation getroffen werden, indem Echtzeitdaten analysiert und Beobachtungen in die Berechnungen des DBNs mit einfließen. Ein DBN kann auf zwei Arten definiert werden. So kann ein DBN mittels Expertenwissen bestimmt oder mit maschinellen Lernverfahren generiert werden.

In der Masterarbeit wird mit einer prototypischen Realisierung eine Schmuggelerkennung mittels DBNs und der Verwendung von AIS-Daten implementiert. Zum einen werden in der Arbeit die DBNs für die Erkennung von maritimen Situationen betrachtet und zum anderen ein echtzeitfähiges Konzept vorgestellt, mit dem die probabilistische Situationserkennung mittels eines Open-Source Projektes prototypisch umgesetzt wird. Mit Hilfe der verteilten Streaming-Plattform Apache Kafka können die Daten anhand kleiner Prozesse analysiert und Informationen ausgetauscht werden.

Zur Evaluation der Ergebnisse für die korrekte Situationserkennung werden Vergleichswerte durch das logische Schließen genutzt und das implementierte DBN auf Robustheit und Sprungverhalten untersucht. Die Ergebnisse zeigen, dass eine beispielhafte Situation mittels DBNs und AIS-Daten korrekt erkannt wird.

Die DBNs eignen sich auch für sicherheitskritische Anwendungen, da diese aufgrund der Graphen- und Wahrscheinlichkeitstheorie erklärbar sind. Der Einsatz von DBNs im maritimen Raum wurde bereits untersucht und z.B. in (Fischer, 2016) und in (Yuanqiao Wen, 2019) als nützliche Methodik evaluiert.

3.2.7.2. Dynamische Bayes'sche Netze

In diesem Kapitel wird die Theorie zu dem dynamischen Bayes'schen Netz erläutert. Das DBN ist Teil der probabilistischen graphischen Modelle und eine Erweiterung des

Bayes'schen Netz (BN), weswegen dieses zuerst vorgestellt wird. Im Anschluss wird das DBN betrachtet, welches das BN um die Zeitkomponente ergänzt.

Bayes'sches Netz

Das Bayes'sche Netz ist eine Methode zum statischen Schlussfolgern unter Unsicherheit basierend auf der bedingten Wahrscheinlichkeitsrechnung und eines direkten azyklischen Graphen (DAG) (vgl. (Rammelt, 2019)). Nachfolgend soll auf die Repräsentation eingegangen werden und die Struktur und die Parametrisierung eines BNs erläutert werden.

Ein BN repräsentiert eine gemeinsame Wahrscheinlichkeitsverteilung von einer Anzahl an diskreten oder kontinuierlichen Zufallsvariablen als ein DAG. Bei diskreten Zufallsvariablen entspricht jeder Knoten im DAG einer Variable und einer zugehörigen bedingten Wahrscheinlichkeitstabelle (CPT), die die Wahrscheinlichkeit der einzelnen Zustände einer Variablen gegeben durch die Elternknoten beinhaltet. Dabei werden die bedingten Unabhängigkeitsannahmen ausgenutzt,

die durch die Struktur des Netzes impliziert werden (vgl. (Sucar, 2015)). Die gemeinsame Wahrscheinlichkeitsverteilung eines BNs wird allgemein definiert als (vgl. (Sucar, 2015)):

$$P(X^1, \dots, X^n) = \prod_{i=1}^n P(X^i | Pa(X^i)).$$

Mit $Pa(X^i)$ werden die Elternknoten der Zufallsvariable X^i bezeichnet. Wenn X^i keine Elternknoten besitzt, dann ist $Pa(X^i) = 0$ und $P(X^i | Pa(X^i)) = P(X^i)$, welches die A-priori-Wahrscheinlichkeit von X^i bezeichnet. $P(X^i | Pa(X^i))$ ist für jedes i die bedingte Wahrscheinlichkeitsverteilung (CPD) eines Knoten bzw. einer Variable (vgl. (Fischer, 2016)). Bei diskreten Variablen können diese mit den CPTs dargestellt werden. BNs können auch durch kontinuierliche Variablen oder durch eine gemischte Form aus diskreten und kontinuierlichen Variablen repräsentiert werden. In dieser Arbeit werden die Ergebnisse der BNs bzw. DBNs nur mit diskreten und binären Variablen betrachtet, weil damit die zwei Zustände „Situation existiert“ und „Situation existiert nicht“ abgebildet werden können. Die Abbildung 3-40 zeigt ein Beispiel eines BNs mit 5 Knoten bzw. Variablen und deren zugehörigen CPDs. Die gemeinsame Wahrscheinlichkeitsverteilung ist gegeben durch das Produkt der CPDs (vgl. (Fischer, 2016)):

$$P(A, B, C, D, E) = P(A)P(E)P(B|A)P(C|B)P(D|B, E).$$

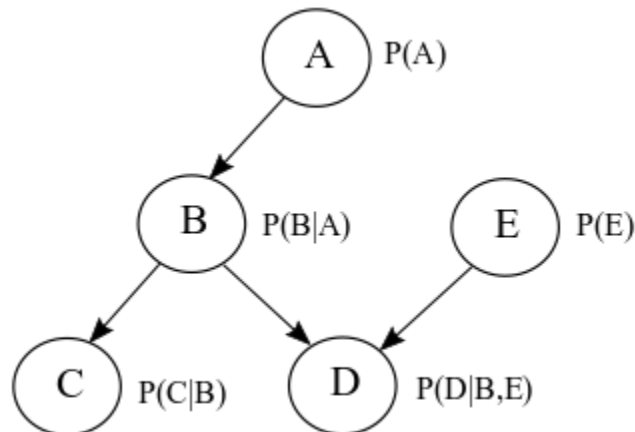


Abbildung 3-40: Beispiel eines BNs mit den Variablen A,B,C,D und E und den zugehörigen CPDs.

Anhand der Struktur des BNs können sowohl bedingte Abhängigkeiten zwischen Eltern- und Kindknoten als auch bedingte Unabhängigkeitsannahmen verifiziert werden. So ist in der Abbildung 3-40 der Knoten *C* bedingt unabhängig von *A*, *D* und *E* gegeben durch *B*. Variablen, deren Zustand durch Beobachtung bekannt ist, werden als „gegeben“ bezeichnet. In mathematischer Form wird dies folgendermaßen geschrieben (vgl. (Sucar, 2015)):

$$P(C|A, B, D, E) = P(C|B)$$

Die durch die Struktur eines BNs implizierten bedingten Unabhängigkeitsannahmen sollten den bedingten Unabhängigkeitsbeziehungen der gemeinsamen Wahrscheinlichkeitsverteilung entsprechen und umgekehrt. Das Verifizieren dieser bedingten Unabhängigkeiten wird auch als *DSeperation* bezeichnet. In einem BN gibt es 3 grundlegende Abhängigkeitsstrukturen für drei Variablen und zwei Pfeilen (vgl. (Sucar, 2015)):

- seriell: $X \rightarrow Y \rightarrow Z$.
- divergent: $X \leftarrow Y \rightarrow Z$.
- konvergent: $X \rightarrow Y \leftarrow Z$.

Bei der seriellen und der divergierenden Verbindung sind *X* und *Z* jeweils bedingt unabhängig gegeben durch *Y*. Wenn *Y* bei der konvergierenden Verbindung jedoch bekannt ist, werden *X* und *Z* bedingt abhängig. Dieser Effekt wird im englischen als „explaining away“ bezeichnet, weil durch den bekannten Zustand von *Y* und das Beobachten des Zustandes einer der beiden anderen Variablen auf den Zustand der dritten Variable geschlossen werden kann (vgl. (Sucar, 2015)). Ist *Y* gegeben und ist beispielsweise der Zustand von *X* bekannt, so kann der Zustand von *Z* vernachlässigt werden, da durch die Ursache *X* der Effekt *Y* erklärt werden kann. In der Abbildung 3-40 können alle drei Abhängigkeitsstrukturen verifiziert werden. *A*, *B* und *C* haben eine serielle Verbindung. *C*, *B* und *D* haben eine divergierende Verbindung und *B*, *D* und *E* haben eine konvergierende Verbindung.

Ist die Struktur eines BNs festgelegt, können anschließend die Parameter definiert werden. Die Parameter eines BNs sind die bedingten Wahrscheinlichkeiten eines jeden Knotens gegeben durch die Elternknoten des Graphen. Das Verhalten eines BNs bzw. DBNs wird wesentlich von den Parametern der einzelnen Variablen bestimmt. Im Falle von diskreten Variablen können Wurzelknoten durch ihre A-priori-Wahrscheinlichkeiten definiert werden,

die verallgemeinert ebenfalls in einer CPT gesetzt werden können. Alle anderen Knoten können durch die CPTs der Variablen gegeben der Elternknoten definiert werden. Die Abbildung 3-41 zeigt das vorherige Beispiel eines BNs mit denselben Variablen A – E und den assoziierten bedingten Wahrscheinlichkeitsverteilungen anhand der CPTs. Die Wahrscheinlichkeitswerte wurden zufällig gewählt und sollen nur die CPTs der einzelnen Variablen veranschaulichen. In der Abbildung 3-41 ist zu erkennen, dass die Wurzelknoten A und E die A-priori-Wahrscheinlichkeit in Form einer eindimensionalen CPT dargestellt ist. Alle anderen Knoten haben mehrdimensionale CPTs. Je mehr Elternknoten ein Knoten besitzt, desto komplexer wird die bedingte Wahrscheinlichkeitstabelle.

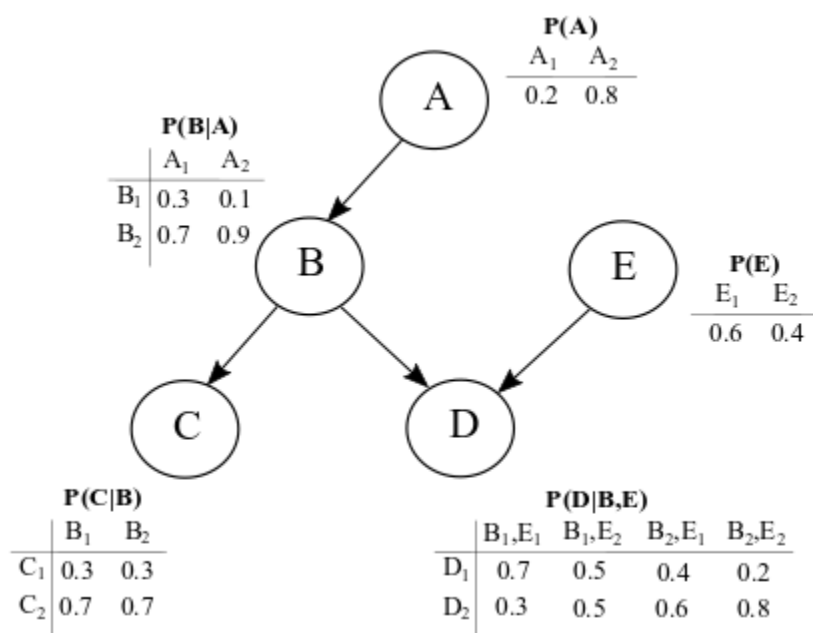


Abbildung 3-41: Beispiel eines BNs mit den Variablen A,B,C,D und E und den assoziierten CPTs.

Die Anzahl der zu bestimmenden Parameter pro Variable X^i und CPT lässt sich bei binären Variablen mit (vgl. (Koller & Friedman, 2009; Fischer, 2016)):

$$|CPT(X^i)| = 2 * 2^k = 2^{k+1}.$$

berechnen, wobei k für die Anzahl der Elternknoten von X^i steht. Um die Gesamtzahl der zu bestimmenden Parameter zu ermitteln, müssen die Ergebnisse von $|CPT(X^i)|$ jedes einzelnen Knoten X^i aufsummiert werden. Die Gesamtzahl der Parameter aller CPTs kann bei binären Variablen halbiert werden, da die Summe einer Spalte von einer CPT immer 1.0 ergeben muss. In Abbildung 3-41 ergibt z.B. der Wert von B_1 und B_2 pro Spalte in der CPT von $P(B|A)$ immer 1.0 und somit die Wahrscheinlichkeit von 100%.

Dynamisches Bayes'sches Netz

Das dynamische Bayes'sche Netz erweitert das Bayes'sche Netz dahingehend, dass Schlussfolgerungen über ganze Zeitreihen durchgeführt werden können. Das BN dient dabei grundsätzlich als Schablone für jeden Zeitschritt. Zeitlich gerichtete Kanten zwischen Variablen zweier benachbarter Schablonen ergänzen dies zu einem DBN.

Das DBN kann als ein Paar $(B_0, 2TBN)$ definiert werden, wobei B_0 die A-priori-Verteilung von $P(X_0^{1:n})$ zum Zeitpunkt $t = 0$ über die Menge $X_{1:n} = \{X_1, \dots, X_n\}$ von Zufallsvariablen ist (vgl. (Koller & Friedman, 2009), (Fischer, 2016)). Das $2TBN$ definiert die bedingte Wahrscheinlichkeitsverteilung über einen Zeitschritt hinweg wie folgt (Koller & Friedman, 2009; Fischer, 2016):

$$P(X_t^{1:n} | X_{t-1}^{1:n}) = \prod_{i=1}^n P(X_t^i | Pa(X_t^i)).$$

Zu jedem Zeitschritt können die Knoten eines DBN einer sogenannten Zeitscheibe zugeordnet werden. X_t^i repräsentiert dabei einen Knoten der Zeitscheibe t und $Pa(X_t^i)$ die Elternknoten von X_t^i , wobei die einzelnen Elternknoten sich in der Zeitscheibe t oder in der Zeitscheibe $t - 1$ befinden können (vgl. (Fischer, 2016)). Die gemeinsame Wahrscheinlichkeitsverteilung von $X_{0:T}^{1:n}$ mit einem $T > 0$ definiert ein DBN über T Zeitschritte und kann folgendermaßen berechnet werden (vgl., (Koller & Friedman, 2009; Fischer, 2016)):

$$P(X_{0:T}^{1:n}) = \prod_{t=0}^T \prod_{i=1}^n P(X_t^i | Pa(X_t^i)).$$

Die Abbildung 3-42 zeigt ein Beispiel eines DBN mit einem Paar $(B_0, 2TBN)$. Das DBN ist mit 3 Knoten X_t^1 , X_t^2 und X_t^3 modelliert und zeigt eine sequenzielle Abhängigkeitsstruktur. Die Apriori-Verteilung B_0 ist zum Zeitpunkt $t = 0$ definiert. Das $2TBN$ zeigt die zeitlichen bedingten Abhängigkeiten für dieses DBN.

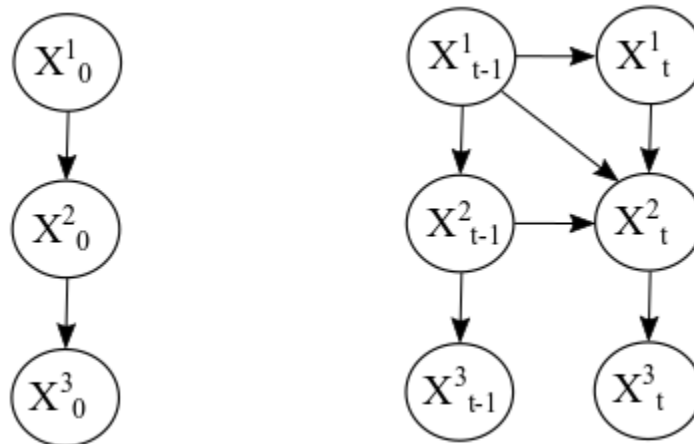


Abbildung 3-42: Beispiel eines DBNs mit der A-priori-Verteilung B_0 links und dem $2TBN$ rechts (basierend auf (Fischer, 2016; Sucar, 2015)).

In der Abbildung 3-42 ist nun das DBN mit dem Paar $(B_0, 2TBN)$ aus Abbildung 3-43 über 4 Zeitschritte dargestellt. B_0 dient dabei als Startverteilung und das $2TBN$ kann für das Duplizieren des DBNs über mehrere Zeitschritte heran genommen werden.

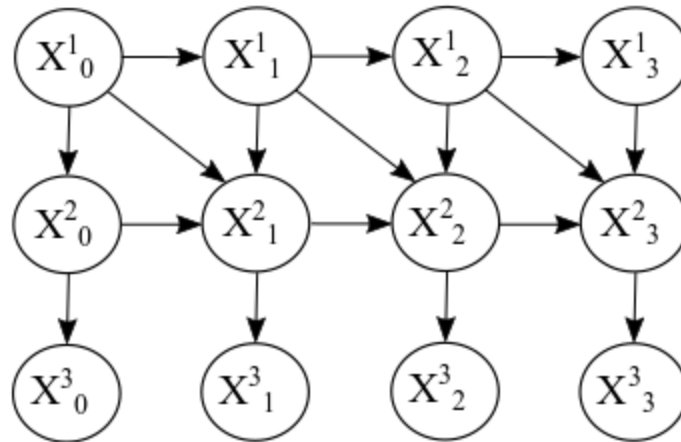


Abbildung 3-43: Beispiel eines DBNs mit 3 Knoten über 4 Zeitschritte (vgl. (Sucar, 2015)).

Für die Modellierung und Parametrisierung eines DBNs werden normalerweise folgende vereinfachenden Grundannahmen gemacht (vgl. (Sucar, 2015; Rammelt, 2019)):

1. Temporale Kanten zeigen immer in Richtung zukünftiger Zeitscheiben.
2. Das Problem ist stationär, d.h. die Struktur und die Parameter des DBNs ändern sich nicht über die Zeit, ausgenommen ist davon die Prior-Zeitscheibe $t = 0$.
3. Die Markov-Bedingung 1. Ordnung wird erfüllt, d.h. Kanten existieren nur innerhalb einer Zeitscheibe t und zur folgenden Zeitscheibe $t + 1$. Dadurch ist die Zeitscheibe $t - 1$ unabhängig von der Zeitscheibe $t + 1$ gegeben durch alle Knoten in Zeitscheibe t .

Zur Definition eines DBNs sind damit nur die Struktur einschließlich der zeitlichen bedingten Abhängigkeiten und die Parameter bzw. die CPTs aus der Prior-Zeitscheibe $t = 0$ und der folgenden Zeitscheibe $t = 1$ notwendig, die alle folgenden Zeitscheiben repräsentiert (vgl. (Rammelt, 2019)).

Methoden zur Bestimmung der DBN-Parameter

Während die Struktur der BNs und DBNs meist intuitiv und verständlich modelliert werden kann, gibt es bei der Bestimmung der Parameter prinzipiell mehrere Möglichkeiten. Zum einen können die Parameter anhand von Daten erlernt werden. Zum anderen können mit Expertenwissen die Parameter manuell definiert werden oder beispielsweise Gewichtungsverfahren verwendet werden.

Lernen in dynamischen Bayes'schen Netzen

Sucar zeigt mit welchen Methoden die Parameter mittels Daten erlernt werden können (vgl. (Sucar, 2015)). Ein DBN kann mittels maschinellem Lernen anhand eines Datensatzes generiert werden. Dabei können entweder die Parameter des DBNs oder die Struktur des DBNs erlernt werden. Ist die Struktur beispielsweise durch Expertenwissen bekannt, so können beim Parameter lernen die bedingten Wahrscheinlichkeitstabellen (CPTs) von den Daten ermittelt werden. Die CPT für jede Variable kann aus den Daten auf der Grundlage der Häufigkeit jedes Wertes oder der Kombination von Werten geschätzt werden, um einen Parameter zu erhalten, der am plausibelsten erscheint. Beim Lernen der Parameter gibt es zwei typische Fälle bei der Annahme, dass die Struktur bekannt ist (vgl. (Sucar, 2015)). Im

ersten Fall enthält der zur Verfügung stehende Datensatz genug Daten, um die Werte der Variable zu berechnen. Hierfür wendet man die sogenannte *Maximum-Likelihood*-Schätzung an. Im zweiten Fall sind manche Variablen nicht vollständig beobachtbar, da ein unzureichender Datensatz zur Verfügung steht. In diesem Fall wird der sogenannte *Expectation-Maximization*-Algorithmus angewendet.

Das Lernen der Parameter wird zur maritimen Situationserkennung in dieser Arbeit nicht verwendet, weil die Datengrundlage für illegale Vorfälle nicht ausreichend genug ist und sich die Parameterbestimmung durch Expertenwissen hierfür besser geeignet ist.

Nutzung von Expertenwissen

Das Nutzen von Expertenwissen und das manuelle Setzen der Parameter kann z.B. durch Expertenbefragung erreicht werden, indem Fragen für jede Variable und deren Zustände beantwortet werden müssen. Anschließend können die ermittelten Parameter in die CPTs eingetragen werden. Des Weiteren kann auch Vorwissen genutzt werden, das beispielsweise durch frequentistische Beobachtungen gewonnen werden kann. Sind die Wahrscheinlichkeiten der Zustände einer Variable bekannt, dann können diese direkt gesetzt werden. Wenn aber überhaupt kein Vorwissen über eine Variable vorhanden ist, so wird das Prinzip der maximalen Entropie angewendet. Dies führt zu einer Gleichverteilung für die Zustände einer Variable (vgl. (Fischer, 2016)). Im Falle von binären diskreten Variablen gilt (vgl. (Fischer, Reisch, & Beyerer, 2014)):

$$\text{prior}(X) = P(X) = 0,5.$$

Fischer geht ebenfalls auf die Nutzung von Vorwissen ein, beschreibt Expertenbefragungen und zeigt weitere Methoden für die Bestimmung der Parameter auf (vgl. (Fischer, 2016)). So wird z.B. ein Gewichtungungsverfahren vorgestellt, auf das Noisy-Or-Modell eingegangen und ausführlich das generalisierte lineare Modell erläutert.

Die vorgestellten Methoden können sowohl für die Bestimmung der Parameter von BNs als auch von DBNs verwendet werden.

Inferenz bei einem DBN

Um Schlussfolgerungen mit einem DBN über die Zeit zu machen, gibt es verschiedene Möglichkeiten Inferenz-Berechnungen durchzuführen. Mit verschiedenen Inferenz-Algorithmus kann die Wahrscheinlichkeit einer Situation anhand der beobachteten Evidenzen z.B. zum aktuellen Zeitpunkt, zu einem vergangenen Zeitpunkt oder zu einem zukünftigen Zeitpunkt berechnet werden. Außerdem kann die wahrscheinlichste Erklärung zu einer Folge von Zuständen berechnet werden, welche am besten die Evidenzen generiert.

Für die Situationserkennung wird nach der Wahrscheinlichkeit der Situation von Interesse S gesucht. Die Wahrscheinlichkeit der Situation S gegeben durch die Menge der beobachteten Evidenzen $\mathbf{e} = \{e_1, e_2, \dots\}$ kann mit der bedingten Wahrscheinlichkeit $P(S|\mathbf{e})$ berechnet werden. Der Kern der Bayes'schen Inferenz ist der Satz von Bayes, der die Berechnung der bedingten Wahrscheinlichkeit beschreibt und zur Ermittlung der Wahrscheinlichkeit der Situation S folgendermaßen umgeformt werden kann (vgl. (Fischer, 2016)):

$$P(S|\mathbf{e}) = \frac{P(\mathbf{e}|S)P(S)}{P(\mathbf{e})}.$$

Für die Inferenzberechnung der DBNs über eine Zeitspanne gibt es 4 hauptsächlich genutzte Möglichkeiten:

- Filterung
- Glättung
- Prädiktion
- Wahrscheinlichste Erklärung

Das häufigste Inferenzproblem kann mit der Filterung berechnet werden, welches die Wahrscheinlichkeit zum aktuellen Zeitpunkt mit allen vorherigen und aktuellen Beobachtungen berechnet. Die Glättung wird zur Wahrscheinlichkeitsberechnung eines vergangenen Zeitpunktes verwendet und die Prädiktion zur Wahrscheinlichkeitsberechnung eines zukünftigen Zeitpunktes. Bei der wahrscheinlichsten Erklärung wird die wahrscheinlichste Folge von Zuständen der gesuchten Situation berechnet, welche durch alle beobachteten Evidenzen erzeugt wurde.

Die vorgestellten Inferenz-Möglichkeiten können mit verschiedenen Algorithmen berechnet werden. Diese Inferenzalgorithmen teilen sich in exakte und approximative Algorithmen auf. In (Murphy, 2002) wurden einige Algorithmen für die Inferenzberechnung zusammengetragen und vorgestellt. Auch Fischer gibt in (Fischer, 2016) eine Übersicht über die wichtigsten Inferenzalgorithmen. Hier wird eine kurze Übersicht über gängige Inferenzalgorithmen gegeben und auf den Inferenzalgorithmus eingegangen, der in dieser Arbeit verwendet wurde.

Die Inferenzalgorithmen der DBNs basieren auf den Inferenzalgorithmen der BNs. Die wichtigsten Algorithmen für die Inferenzberechnung bei den BNs sind die Variablenelimination, der Junction-Tree-Algorithmus und das Message Passing, welche zu den exakten Inferenztypen zählen. Bekannte approximative Inferenzalgorithmen für BNs sind beispielsweise der Loopy-Belief-Propagation-Algorithmus oder stochastische Sampling Methoden, die auf dem Importance Sampling oder dem Monte Carlo Markov Chain basieren (vgl. (Murphy, 2002)).

Exakte Inferenzalgorithmen können für jedes zeitdiskrete DBN angewendet werden. Bei komplexeren diskreten DBNs benötigen sie jedoch einen zu großen Rechenaufwand, wodurch approximative Inferenzalgorithmen besser geeignet sind (vgl. (Murphy, 2002)). Die approximierten Inferenzalgorithmen können in deterministische und stochastische Algorithmen aufgeteilt werden. Approximierte stochastische Inferenzalgorithmen können für fast jedes Modell verwendet werden, liefern exakte Berechnungen in einem Bereich von sehr vielen Abtastwerten (samples) und sind leichter als approximative deterministische Algorithmen zu implementieren. Diese Sampling Methoden haben aber den Nachteil, dass sie im Vergleich zu den approximierten deterministischen Algorithmen mehr Zeit bei der Inferenzberechnung benötigen und deswegen langsamer sind (vgl. (Murphy, 2002)).

In der Masterarbeit wurde der approximative deterministische *faktorierte Frontier-Algorithmus* mit dem Importance Sampling und mit Hilfe des Java Projekts *Amidst-Toolbox* verwendet, welches ein Open Source Projekt für skalierbares probabilistisches maschinelles Lernen ist und womit DBNs modelliert und Inferenzberechnungen mit verschiedenen Algorithmen ausgeführt werden können. Dieser Algorithmus lieferte von den vorhandenen Algorithmen die besten Ergebnisse.

Motivation für die Verwendung der DBNs zur Situationserkennung

Im Folgenden soll die Motivation für die Verwendung der DBNs für die Situationserkennung im maritimen Umfeld erläutert werden. DBNs sind eine probabilistische Methodik zum

Schlussfolgern unter Unsicherheit mit Einbeziehung der Zeit. Das Schließen unter Unsicherheit hat gegenüber logischen Verfahren, wie der Aussagen und Prädikatenlogik, den Vorteil, dass bei nicht vollständiger Informationsbasis bereits Aussagen über die Existenz von Situationen getroffen werden können. Durch den Umgang mit Unsicherheit und der Zeitkomponente können DBNs außerdem verrauschte Sensordaten kompensieren (vgl. (Fischer, 2016)). Bei nicht ausreichender Datengrundlage können DBNs mit Hilfe von Expertenwissen modelliert und parametrisiert werden. BNs und insbesondere DBNs liefern in der Anomaliedetektion und in der spezifischen Situationserkennung sehr gute Ergebnisse (vgl. (Fischer, 2016)). Gerade bei Situationen, die eher selten auftreten, ist die Verwendung eines DBNs gegenüber maschinellen Lernverfahren vorteilhaft, da hierzu wenig bis keine Daten vorhanden sind. Anhand von Experten- und Vorwissen kann beispielsweise mit frequentistischen Beobachtungen jedoch eine Situationserkennung wissensbasiert umgesetzt werden. Mit den DBNs und deren Bayes'schen Wahrscheinlichkeitsrechnung wird einerseits eine konsistente mathematische Grundlage und andererseits eine intuitiv verständliche Beschreibungsmöglichkeit zur Verfügung gestellt (vgl. (Rammelt, 2019)). Mit den DBNs können komplexere Probleme und abstrakte Situationen abgebildet werden. Einzelne, kleinere bzw. elementare Situationen können mittels eines DBNs zu einer komplexeren und abstrakteren Situation verknüpft werden. Im maritimen Umfeld kann beispielsweise die abstrakte Situation „Schmuggelboot“ abgebildet werden, die aus mehreren elementaren Situationen besteht, wie z.B. „Schiff hat verdächtigen Schiffstyp“, „Schiff nähert sich anderem Schiff“ und „Schiff befindet sich in verdächtigem Gebiet“. Mittels kleiner Analyse- Prozesse von Live-Daten für die elementaren Situationen können so Beobachtungen in das DBN in naher Echtzeit einfließen und somit auf eine komplexere und abstrakte Situation geschlussfolgert werden.

Ein weiterer Vorteil bei der Verwendung von DBNs ist die Erklärbarkeit einer implementierten automatischen Situationserkennung. DBNs sind aufgrund der Graphen- und Wahrscheinlichkeitstheorie von Natur aus erklärbar und haben so einen Vorteil gegenüber maschinellen Lernverfahren. Beim Deep Learning sind zum Beispiel die Techniken trotz sehr guter Ergebnisse bei einer ausreichend großen Datenmenge nicht komplett nachvollziehbar (vgl. (IOSB, 2020)). Für kritische Anwendungen werden aber meist zertifizierte Techniken benötigt, wodurch sich maschinelle Lernverfahren noch nicht richtig etablieren konnten (vgl. (IOSB, 2020)).

3.2.7.3. Umsetzung der Situationserkennung

Bei der Umsetzung der automatischen Situationserkennung wurde zuerst die Situationsmodellierung betrachtet und in Folge dessen ein Situationsnetzwerk zur Erkennung eines Schmuggelbootes modelliert. Dieses einfach gehaltene Situationsnetzwerk zeigt die Struktur eines BNs, welches im Anschluss mit einer zeitlichen Kante zu einem DBN erweitert wurde. Des Weiteren wurden auch Überlegungen zu einer komplexeren Schmuggelsituation gemacht. Nach der Situationsmodellierung wird erläutert, wie die Situationserkennung prototypisch realisiert wurde. Bei der prototypischen Umsetzung wurde ein Konzept erarbeitet, wie ein DBN-Framework mit der verteilten Streaming Plattform Apache Kafka integriert werden kann.

Situationsmodellierung

Bei der Situationsmodellierung geht es darum, eine Situation mit ihren Merkmalen darzustellen. Hier wird die Situation von Interesse in Form eines Bayes'schen Netzes dargestellt. Dies dient als Grundlage zur Erstellung eines DBNs. Eine gesuchte Situation

besteht in der Regel aus mehreren elementaren Situationen, die diese beschreiben. Elementare Situationen implizieren keine Existenz von weiteren Situationen und können zur Existenz von abstrakteren Situationen führen. Situationen, deren Existenz von der Existenz anderer Situationen abhängt, werden als abstrakte Situationen bezeichnet (vgl. (Fischer, 2016)). Eine modellierte Situation von Interesse kann unterschiedliche Abstraktionsgrade aufweisen. So gibt es Situationen, bei denen die Beschreibung komplexer ausfällt und die somit einen höheren Abstraktionsgrad besitzen.

Bei der Modellierung gibt es mehrere Möglichkeiten, die Abhängigkeiten der Situationen darzustellen. In der Masterarbeit wird für die Erkennung eines Schmuggelbootes ein Situationsnetzwerk aus hinreichenden und notwendigen Bedingungen verwendet. Dieses Situationsnetzwerk ist in der Abbildung 3-44 dargestellt. Das Situationsnetzwerk besteht aus der gesuchten Situation von Interesse S und drei elementaren Situationen. Mit einer hinreichenden Bedingung, kann die Existenz einer Situation durch die Existenz einer anderen Situation erklärt werden. Mit einer notwendigen Bedingung erklärt die Existenz einer Situation S die Existenz von einer anderen Situation (vgl. (Fischer, 2016)). Die zwei Situationen „Position war innerhalb verdächtigen Gebiet“ und „Position war nahe anderem Boot“ sind als hinreichende Situationen modelliert, da diese bei Existenz auf ein Schmuggelboot hindeuten. Die Situation „kein Tanker, Fracht- oder Passagierschiff“ ist als notwendige Situation modelliert, weil ein Schmuggelboot meist einen verdächtigen Typen aufweist. Die Situationen sind zur Vereinfachung mit den Nummerierungen S , E^1 , E^2 und E^3 bezeichnet.

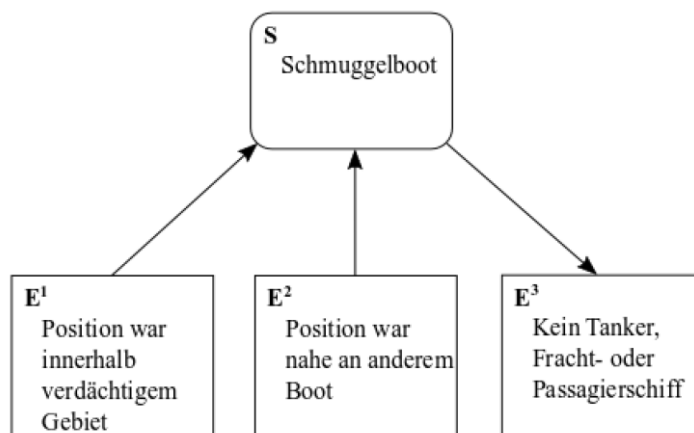


Abbildung 3-44: Eine einfach modellierte Situation eines Schmuggelbootes.

In der Masterarbeit wurde außerdem ein komplexeres Situationsnetzwerk modelliert, welches in der Abbildung 3-45 zu sehen ist. Hier wurde eine verzögerte Drogenübergabe dargestellt, die sowohl aus elementaren als auch abstrakten Situationen besteht. Die Abhängigkeiten wurden bei diesem Situationsnetzwerk in Form eines naiven Bayes'schen Klassifikators mit einer zusätzlichen Abstraktionsebene modelliert. Der naive Bayes'sche Klassifikator repräsentiert eine spezielle Form eines Bayes'schen Netzes.

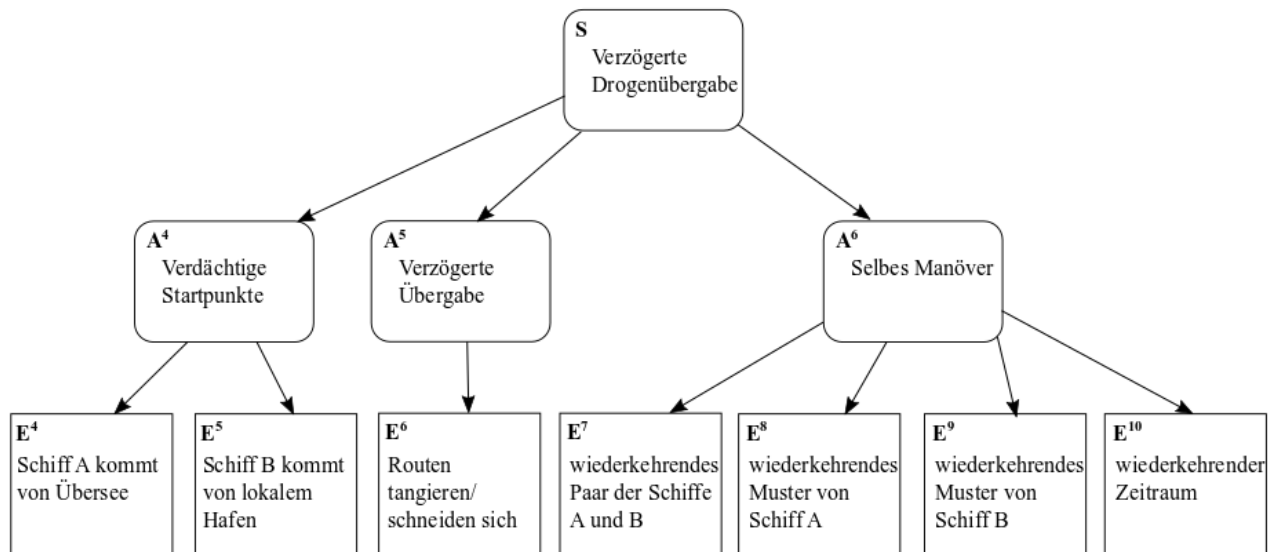


Abbildung 3-45: Das Situationsnetzwerk für die verzögerte Drogenübergabe.

Die abstrakte Situation A^4 besteht aus den zwei elementaren Situationen E^4 und E^5 , welche auf verdächtige Startpunkte der Schiffe A und B hindeuten. Die abstrakte Situation A^5 enthält eine elementare Situation E^6 , mit der eine verzögerte Übergabe erkannt werden kann, wenn sich die Routen der beiden Schiffe schneiden oder annähernd tangieren. Eine Route beschreibt die Wegpunkte eines Schiffes. Die dritte abstrakte Situation A^6 besteht aus den vier elementaren Situationen $E^7 - E^{10}$. Diese Situationen dienen zur Erkennung eines wiederholten Manövers der Schiffe A und B, das sich immer in einem gewissen Zeitraum wiederholt. Die Schiffe A und B zeigen bei diesem Manöver wiederkehrende Bewegungsmuster.

Erstellung des DBNs

Ist die Situation in Form eines Bayes'schen Netzes modelliert, so kann nun ein DBN erstellt werden. Dafür müssen zuerst die temporalen Verbindungen definiert werden. Generell macht es Sinn für alle abstrakten Situationen einschließlich der Situation von Interesse temporale Verbindungen zu definieren. Da bei den elementaren Situationen pro Zeitschritt immer die aktuellen Evidenzen durch Beobachtungen in die Inferenz-Berechnung mit einfließen, sollten die vergangenen Evidenzen nur indirekt berücksichtigt werden.

Die Abbildung 3-46 zeigt das $2TBN$ des Situationsnetzwerks aus Abbildung 3-44, welches zur Schmuggelerkennung verwendet wird. In Zeitscheibe t ist das BN der Situation von Interesse zu sehen, welches mit einer temporalen Kante mit der abstrakten Situation S aus Zeitscheibe $t-1$ verbunden ist. Die zeitliche Kante ist mit der gepunkteten Linie dargestellt. Des Weiteren sind die vereinfachenden Grundannahmen zur Definition einer DBN-Struktur und die Tatsache, dass in einem DAG keine Zyklen erlaubt sind, bei der Erstellung des DBN zu berücksichtigen.

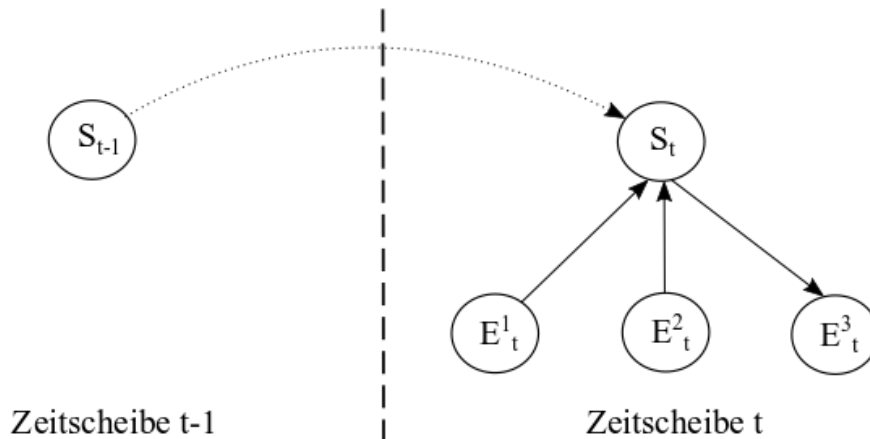


Abbildung 3-46: Das 2TBN des Situationsnetzwerks aus Abbildung 3-44.

Die Parameter des DBNs wurden manuell bestimmt und das Verhalten des DBN für die Situationserkennung anhand einer Evaluation von Sprungverhalten, Robustheit und korrekter Situationserkennung optimiert. Für die Parametrisierung dieses DBNs wurden folgende Annahmen getroffen. Schmuggelaktionen kommen vergleichsweise zum normalen Schiffsverkehr eher selten vor, weswegen dafür eine sehr geringe Wahrscheinlichkeit gewählt wird. Ein Rendezvous außerhalb von Häfen oder Flüssen ist auch eher unwahrscheinlich, wodurch ebenfalls eine niedrige Existenzwahrscheinlichkeit gewählt werden kann. Die Wahrscheinlichkeit, dass ein Schiff ein verdächtiges Gebiet befährt, ist jedoch vergleichsweise etwas höher. Dies kann für jedes Gebiet, aber auch für jeden Schiffstyp individuell betrachtet werden. Ein Schmuggelboot ist in der Regel kein Tanker, Fracht- oder Passagierschiff, aber dies kann nicht vollkommen ausgeschlossen werden, womit die Existenzwahrscheinlichkeit hier annähernd gleich verteilt ist. Des Weiteren deuten ein Schiff in einem verdächtigem Gebiet, aber vor allem ein Rendezvous auf See sehr stark auf ein Schmuggelboot hin, während der Schiffstyp eines Schmuggelboots die Existenz der Situation von Interesse weniger stark beeinflusst. Anhand dieser Annahmen können die Parameter für das DBN im Bereich von 0.0, ..., 1.0 für $P(S_i)$ bestimmt werden. Die Abbildung 3-47 zeigt für welche Situationen wie viele Parameter gesetzt werden müssen. Wie bereits erwähnt, müssen dafür nur die Parameter für die Knoten, also die Situationen, in der Prior-Zeitscheibe $t = 0$ und in der folgenden Zeitscheibe $t = 1$ definiert werden. Die Anzahl der Parameter kann mit der vorgestellten Formel zur CPT-Berechnung bestimmt werden. Für das DBN mussten insgesamt 18 Parameter definiert werden.

Situation S^i	$ CPT(S^i) $	Parameter zu bestimmen
S_0	$2^{2+1} = 8$	$2^2 = 4$
$E_0^1, E_0^2,$	$2^{0+1} = 2$	$2^0 = 1$
E_0^3	$2^{1+1} = 4$	$2^1 = 2$
S_1	$2^{3+1} = 16$	$2^3 = 8$
$E_1^1, E_1^2,$	$2^{0+1} = 2$	$2^0 = 1$
E_1^3	$2^{1+1} = 4$	$2^1 = 2$
Summe	36	18

Abbildung 3-47: Anzahl der CPT-Einträge und der zu bestimmenden Parameter pro Situation

Bei der Situationsmodellierung ist generell darauf zu achten, dass ein Situationsnetzwerk einerseits eine überschaubare Anzahl an Situationen und die daraus resultierende Anzahl an Parameter besitzt, andererseits müssen die Situationen in einem Situationsnetzwerk die Situation von Interesse ausreichend beschreiben, womit eine klare Aussage über die Existenz einer Situation getroffen werden kann.

Prototypische Realisierung

In der Masterarbeit wurde ein echtzeitfähiges Konzept zur probabilistischen Situationserkennung erarbeitet und umgesetzt. Die folgende Abbildung zeigt die Architektur der prototypischen Realisierung. Mit *AMIDST-Toolbox* wurde ein DBN-Framework ausgewählt, welches auf der Programmiersprache Java basiert und Open Source ist. Mit Hilfe der Streaming-Plattform Apache Kafka können Daten analysiert und Informationen ausgetauscht werden. Diese beiden Technologien wurden zur Realisierung des Prototyps verwendet. In der Architektur gibt es grundsätzlich zwei Arten von Anwendungen: Das *DBN-Node*, welches für die Inferenz-Berechnung von aktiven Schiffen zur Situationserkennung verwendet wird und der *analysis process*, welcher AIS-Daten analysiert und Evidenzen an das DBN-Node versenden kann. Die dunkelblauen Rechtecke symbolisieren Kafka-Komponenten und das abgerundete hellblaue Rechteck den DBN-Node.

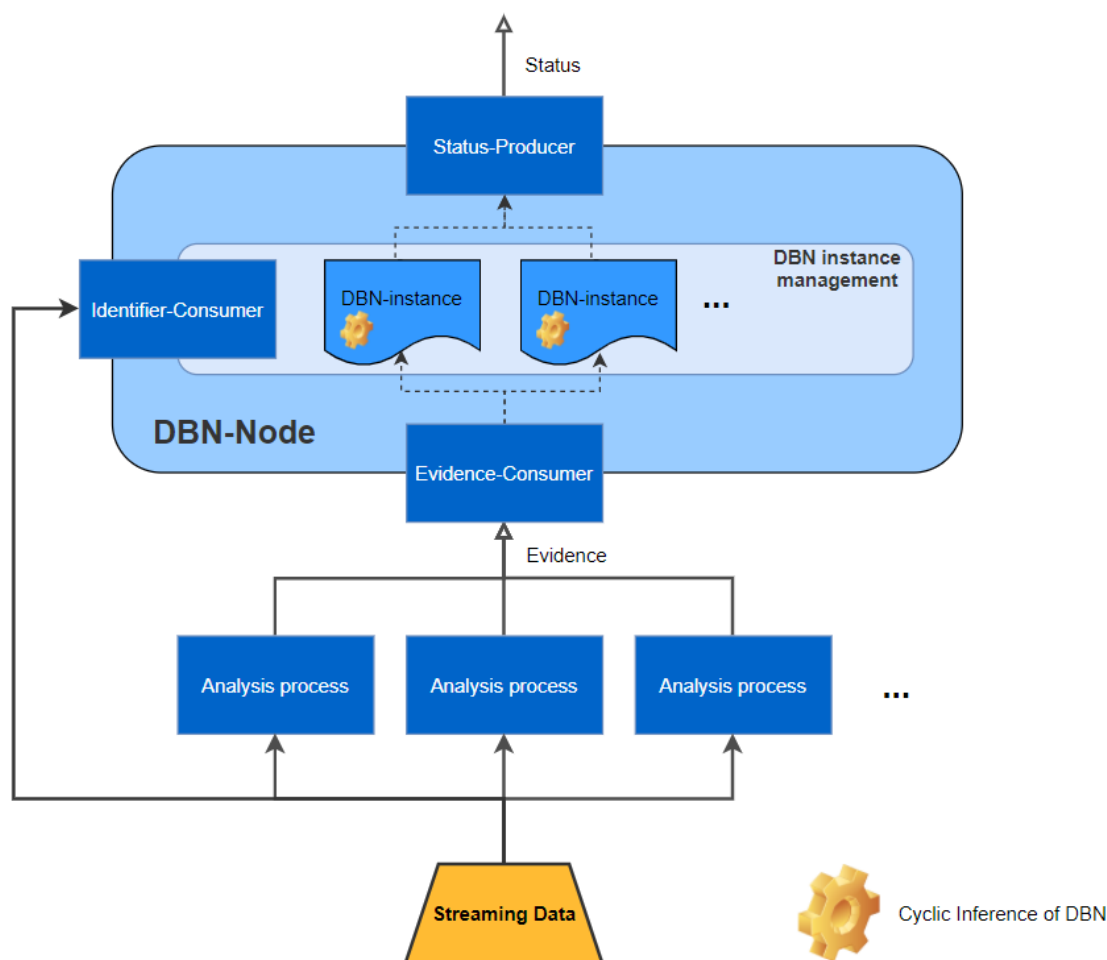


Abbildung 3-48: Architektur der prototypischen Realisierung

Der DBN-Node besitzt zwei Kafka-Consumer mit dem zum einen die MMSI-Nummern der zu Grunde liegenden Daten empfangen (Identifier-Consumer) und zum anderen die beobachteten Evidenzen der Analyse-Prozesse in die DBN-Berechnung miteinfließen können (Evidence-Consumer). Mit Hilfe des Identifier-Consumers werden die DBN-Instanzen verwaltet, womit nur die aktiven Schiffe für die Schmuggelaktivität in Betracht gezogen werden. Mit dem Status-Producer werden die Wahrscheinlichkeiten zu jedem Schiff publiziert. Die Berechnungen der Wahrscheinlichkeiten finden dabei zyklisch statt. Je nach Situation von Interesse kann es mehrere Analyse-Prozesse geben. Jede elementare Situation wird von einem Analyse-Prozess untersucht. Die Analyse-Prozesse beim Prototyp wurden alle mit Kafka-Streams realisiert, die Daten einlesen, prozessieren und daraus folgend Informationen wieder publizieren können. Beim Prototyp wurden für die 3 elementaren Situationen drei Analyse-Prozesse verwendet.

3.2.7.4. Ergebnisse

Zur Evaluation der Ergebnisse wurde der Ansatz aus (Fischer, 2016) gewählt, indem zum einen die Robustheit und das Sprungverhalten untersucht und zum anderen die Ergebnisse eines DBNs mit dem logischen Schließen verglichen wurden. Im Nachfolgenden werden zuerst die Ergebnisse von Robustheit und Sprungverhalten des erstellten DBNs zur Erkennung eines Schmuggelbootes vorgestellt. Mittels eines simulierten Schmuggelszenarios werden anschließend die Ergebnisse mit dem logischen Schließen verglichen. Das simulierte Szenario basiert auf AIS-Daten. Des Weiteren werden die Ergebnisse vorgestellt, die anhand von realen AIS-Daten erstellt wurden.

Evaluation von Robustheit und Sprungverhalten

Zur Evaluation von Robustheit und Sprungverhalten werden jeweils simulierte Evidenzen für die drei elementaren Situationen E^1 , E^2 und E^3 erzeugt.

Für die Untersuchung der Robustheit wird für jeden Zeitschritt und jede elementare Situation eine positive oder negative Evidenz per Zufall und mit einer bestimmten Wahrscheinlichkeit erzeugt. Im Folgenden sei q die Wahrscheinlichkeit, mit der eine negative Evidenz erzeugt wird und es gilt für jede elementare Situation E (vgl. (Fischer, 2016)):

$$P(E) = \begin{cases} 1, & \text{mit Wahrscheinlichkeit } 1 - q, \\ 0, & \text{mit Wahrscheinlichkeit } q. \end{cases}$$

Die Abbildung 3-49 zeigt die Ergebnisse bezüglich der Robustheit mit drei verschiedenen Werten von q . $P(\text{Schmuggelboot})$ bleibt bei einem geringen Rauschwert von $q = 0,1$ im Bereich von $0,8 \dots 1,0$, womit Situationen ab einem Schwellwert von $0,8$ erkannt werden können. Bei einer Rauschzahl von $q = 0,4$, wenn die Daten mit einer Wahrscheinlichkeit von 40% verrauscht sind, kann die Situation von Interesse bei einem Schwellwert von $0,8$ noch oft entdeckt werden. Der Wert von $P(\text{Schmuggelboot})$ liegt aber öfters auch darunter, wodurch hier keine Situation erkannt wird. Bei dem starken Rauschwert von $q = 0,9$ werden bei einem Schwellwert von $0,8$ keine Situationen mehr erkannt, jedoch sind noch vereinzelt starke Anstiege zu erkennen. Dies liegt vor allem daran, dass die elementare Situation E_2 („Position war nahe an anderem Boot“) einen großen Einfluss auf die Existenz der Situation hat und das DBN aufgrund einer temporalen Kante eine geringe Trägheit bei der Erkennung aufweist.

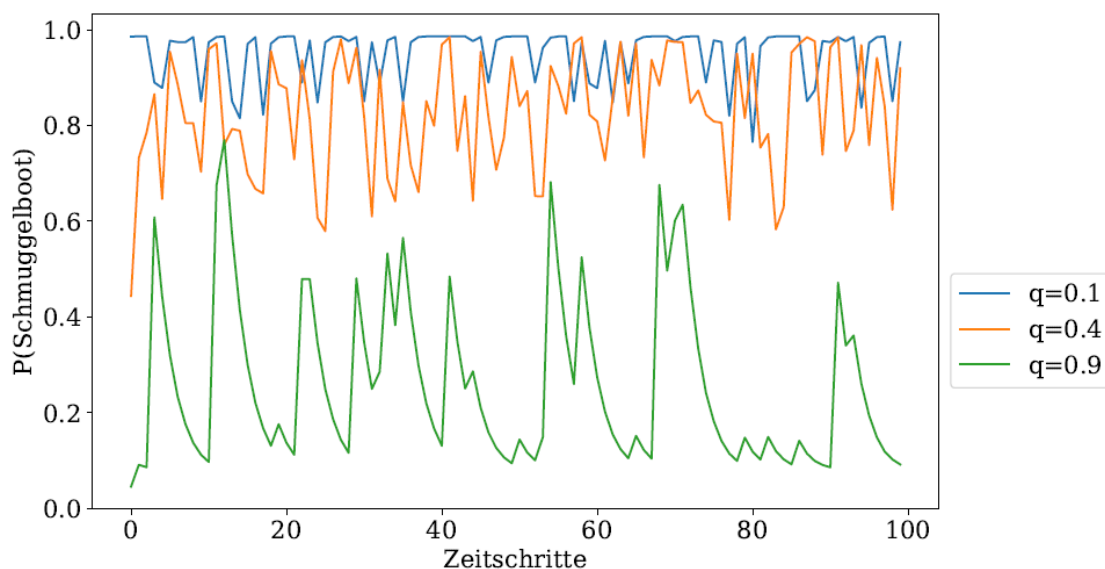


Abbildung 3-49: Robustheit mit drei verschiedenen Rauschwerten q .

Zur Evaluation des Sprungverhaltens wurden auch simulierte Evidenzen verwendet, die für eine bestimmte Verweildauer ausschließlich positiv oder negativ waren. Für alle Elementarsituationen E^1 , E^2 und E^3 wurde eine Folge von positiven und negativen Evidenzen generiert. In der Abbildung 3-50 ist die Existenzwahrscheinlichkeit von $P(\text{Schmuggelboot})$ und die Folge von positiven und negativen Evidenzen dargestellt. Alle 5 Zeitschritte findet ein Wechsel zwischen positiven und negativen Evidenzen statt. In den ersten fünf Zeitschritten ist die Wahrscheinlichkeit schon sehr hoch und verringert sich aufgrund der folgenden fünf negativen Evidenzen wieder. Ab dem Zeitschritt 10 sind wieder nur positive Evidenzen gesetzt und $P(\text{Schmuggelboot})$ steigt innerhalb eines Zeitschritts stark an. Dieses Verhalten wiederholt sich für die nächsten Folgen von Evidenzen. In dieser Abbildung ist also ein schneller Anstieg der Wahrscheinlichkeit bei positiven Evidenzen zu erkennen und ein etwas langsamerer Abfall bei negativen Evidenzen.

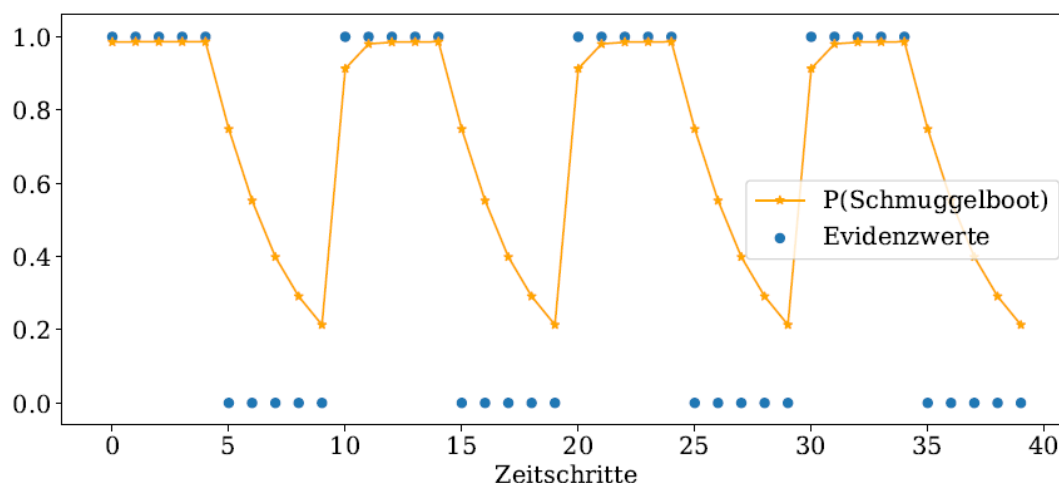


Abbildung 3-50: Sprungverhalten mit 5 aufeinander folgende gleiche Evidenzen.

In diesem Fall spielt die Trägheit des DBNs eine geringere Rolle, da bei den elementaren Situationen E^1 („Position war innerhalb verdächtigem Gebiet“) und E^2 („Position war nahe an anderem Boot“) nach einem vergangenen oder gegenwärtigen Zustand gefragt wird und der Zustand beim Eintreten der Situation dauerhaft gespeichert wird. Durch die Verwendung des DBNs zur Erstellung der nachfolgenden Ergebnisse mit simulierten und realen AIS-Daten kann zum einen die Existenz von elementaren Situationen bei positiver Evidenz in wenigen Zeitschritten erkannt werden und zum anderen wird mit dem einfach gehaltenen DBN eine geringe Rechenkapazität benötigt.

Evaluation mit simulierten AIS-Daten

Im Folgenden werden Ergebnisse mit dem einfachen DBN anhand eines simulierten Szenarios erstellt und mit dem logischen Schließen sowie mit einem äquivalenten BN verglichen. Für die Erstellung der folgenden Ergebnisse wurde die prototypische Realisierung mit dem einfach implementierten DBN und mit den drei Analyse-Prozessen für die elementaren Situationen $E^1 - E^3$ verwendet. Der Analyse-Prozess von E^1 erkennt Schiffe, die sich in einem verdächtigen Gebiet befinden. Das verdächtige Gebiet wird mit einem Polygon definiert. Für die elementare Situation E^2 wird die Distanz eines Schiffes zu den anderen aktiven Schiffen berechnet und bei zu kurzer Distanz in einer gewissen Zeitspanne als Rendezvous gewertet. Der Analyse-Prozess von E^3 überprüft den Schiffstyp, ob dieser kein Tanker, Fracht- oder Passagierschiff ist.

Die Werte des logischen Schließens sollen als Vergleichswerte zur Veranschaulichung der Ergebnisse des DBNs dienen. Das Ziel ist es nicht, mit den Ergebniswerten des DBNs möglichst nahe an die Ergebniswerte des logischen Schließens zu kommen (Fischer, 2016). Das Ergebnis des logischen Schließens berechnet sich anhand der Existenz der elementaren Situationen und liefert dabei den Wert Eins, wenn alle elementaren Situationen existieren und Null, wenn mindestens eine elementare Situation nicht existiert. Eine elementare Situation existiert, wenn die Evidenz der Situation wahr ist und somit Eins ist. Die Überprüfung ob eine Evidenz wahr ist, wird mittels der Analyse-Prozesse festgestellt. Für die modellierte Situation eines Schmuggelbootes S berechnet sich das Ergebnis des logischen Schließens aus den elementaren Situationen $E^1 - E^3$ wie folgt:

$$E^1 = 1 \wedge E^2 = 1 \wedge E^3 = 1 \Rightarrow S = 1.$$

Die Ergebnisse wurden anhand eines beispielhaften simulierten Szenarios mit AIS-Daten erstellt. In diesem Szenario wird eine Schiffsbegegnung simuliert. Insgesamt sind fünf Schiffe unterwegs, die folgende MMSI-Nummern besitzen: 1234, 4012, 4210, 4500 und 6300. Für das Szenario wurde ein verdächtiges Gebiet per Polygon festgelegt, dass von den Schiffen mit den MMSI-Nummern 1234, 4012, 4210 und 4500 befahren wird. Die Abbildung 3-51 soll das Schmuggelszenario verdeutlichen. Die Schmuggelaktion findet in dem rot gekennzeichneten Kreis statt und die Schiffe mit der MMSI-Nummer 4012 und 4210 waren an der Schmuggelaktion beteiligt. Für die Situation von Interesse sind die Positionen der Schiffe und der Schiffstyp interessant. Die Positionen der Schiffe und somit die gefahrenen Routen können der Abbildung 3-51 entnommen werden.

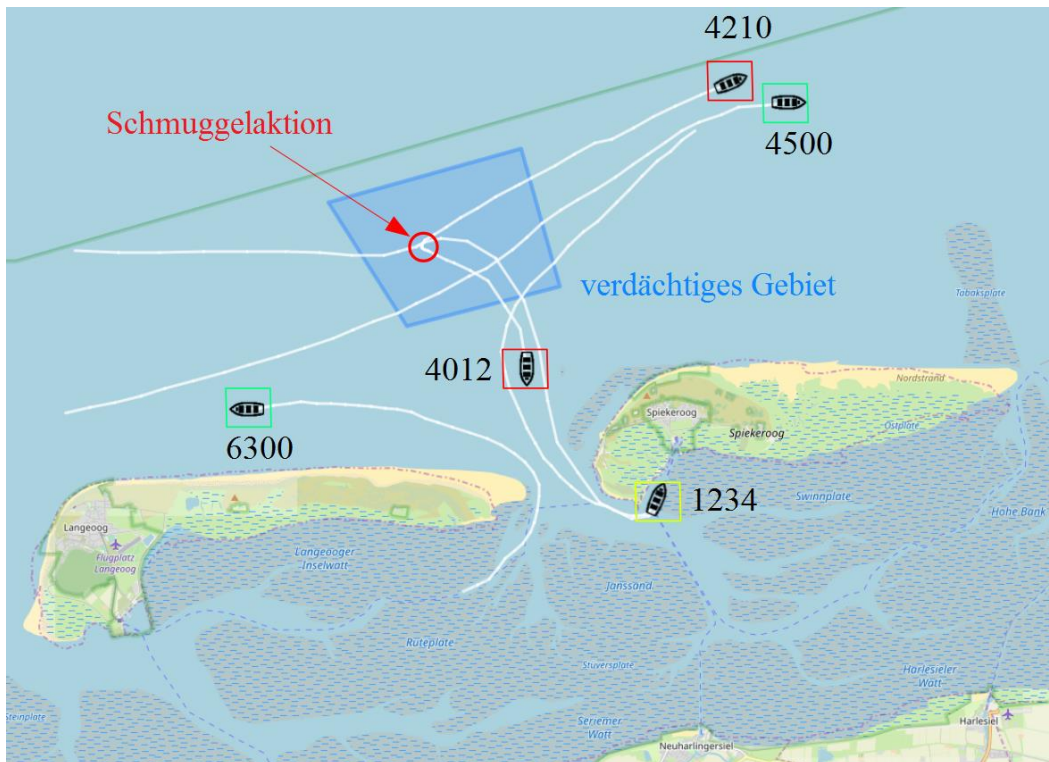


Abbildung 3-51: Das maritime Lagebild zeigt den Schiffsverkehr des simulierten Szenarios.

In der Abbildung 3-52 sind die Schiffstypen der einzelnen Schiffe zu erkennen. Die Existenzwahrscheinlichkeit der Schiffe für ein Schmuggelboot wird in dieser Abbildung mit farbigen Rechtecken markiert. Rot deutet dabei auf eine hohe Existenzwahrscheinlichkeit hin, gelb auf eine mittlere und grün auf eine niedrige Existenzwahrscheinlichkeit.

Schiff mit MMSI	Schiffstyp
1234	Fischerboot
4012	Vergnügungsboot
4210	Frachtschiff
4500	Tanker
6300	Passagierschiff

Abbildung 3-52: Die Schiffstypen der einzelnen Schiffe im beispielhaften simulierten Szenario.

In der Abbildung 3-53 sind die Ergebnisse des Schiffs mit der MMSI-Nummer 4012 dargestellt, welches an der Schmuggelaktion beteiligt war. In dieser und den folgenden Abbildungen werden die Ergebnisse des logischen Schließens und die Ergebnisse des DBNs zur Situationserkennung eines Schmuggelbootes veranschaulicht. Wie zu erkennen ist, sind ab dem Zeitschritt 11 alle Evidenzen wahr, womit die Situation laut dem logischen Schließen eingetreten ist. Die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ nimmt bis zum Zeitschritt 6 langsam zu, da gleich zu Beginn ein Schiffstyp erkannt wurde, der kein Tanker, Fracht- oder Passagierschiff ist. Zum Zeitschritt 7 wurde erkannt, dass das Schiff innerhalb des verdächtigen Gebiets unterwegs ist und zum Zeitpunkt 11 war das Schiff nahe an einem anderen Schiff, wodurch die Existenzwahrscheinlichkeit auf den maximalen Wert von 0,9863 angestiegen ist. Die Wahrscheinlichkeit $P(\text{Schmuggelboot})$ steigt mit der Zeit und mit dem

Eintreten der Evidenzen auf das Maximum an. Wenn eine der elementaren Situationen eingetreten ist und die Evidenz wahr ist, dann bleibt diese dauerhaft wahr, da das Schiff sich in einem verdächtigen Gebiet aufgehalten hatte und verdächtig nahe an einem anderen Schiff war. Der Schiffstyp wird in der Regel nicht verändert.

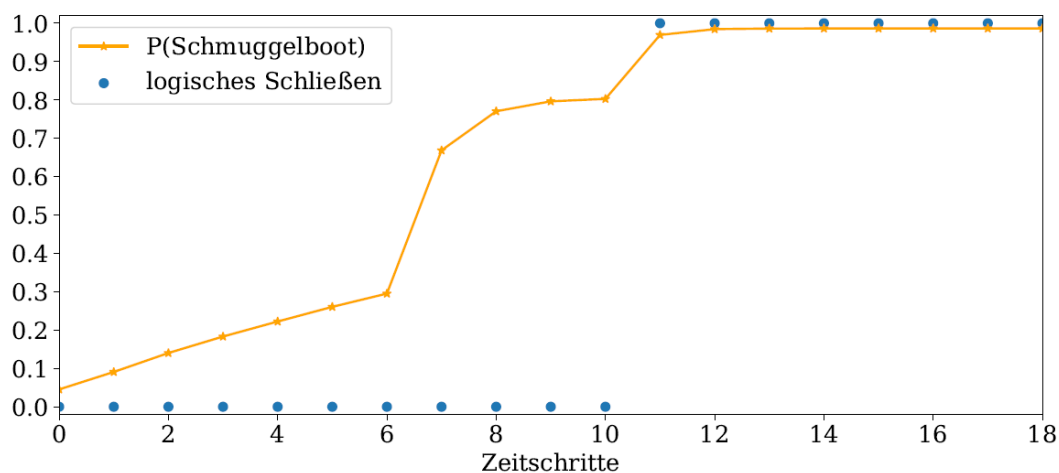


Abbildung 3-53: Ergebnisse des Schiffes mit der MMSI-Nummer 4012.

Die Abbildung 3-54 zeigt die Ergebnisse des Schiffes mit der MMSI-Nummer 4210, das genauso in der Schmuggelsituation involviert war. Da dieses Schiff vom Typ her ein Frachtschiff ist, sind nicht alle drei Evidenzen wahr und somit bleibt beim logischen Schließen der Wert auf Null. Bis zum Zeitpunkt 6 pendelt sich die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ auf einen niedrigen Wert ein, bevor erkannt wird, dass das Schiff in einem verdächtigen Gebiet unterwegs ist. Die Existenzwahrscheinlichkeit steigt. Zum Zeitpunkt 11 war dieses Schiff dem anderen Schiff mit der MMSI-Nummer 4012 sehr nahe, wodurch ein sogenanntes Rendezvous erkannt wurde. Die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ steigt auf ein Maximum von 0,9743 an. Zu erkennen ist, dass ein ähnlich hohes, jedoch geringeres Maximum als in Abbildung 3-53 erreicht wird, obwohl das Schiff ein Frachtschiff ist und somit keinen verdächtigen Schiffstyp besitzt. Dies liegt daran, dass der Einfluss des Rendezvous sehr hoch gesetzt und der Einfluss des verdächtigen Schiffes (kein Tanker, Fracht- oder Passagierschiff) schwächer definiert wurde.

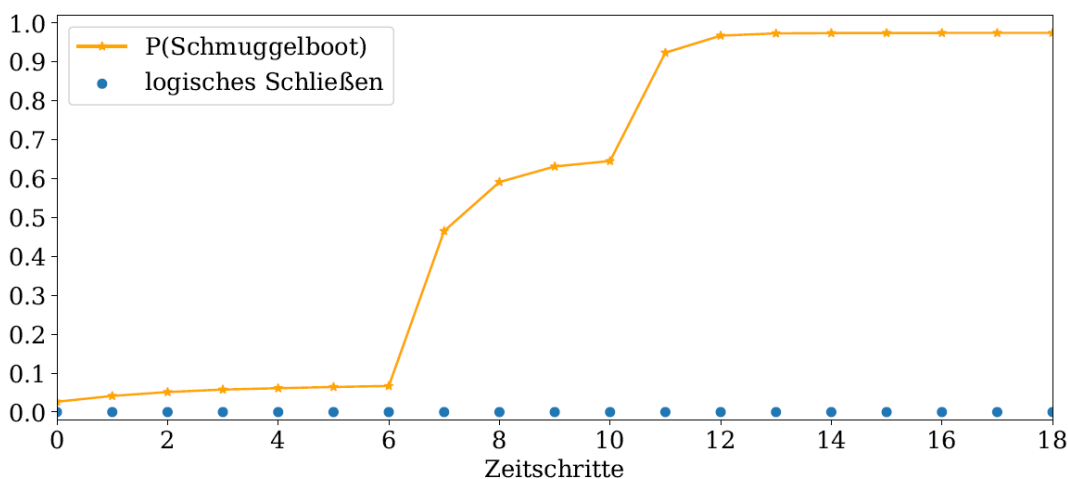


Abbildung 3-54: Ergebnisse des Schiffes mit der MMSI-Nummer 4210.

Die Ergebnisse des Schiffes mit der MMSI-Nummer 1234 sind in Abbildung 3-55 dargestellt. Die Abbildung zeigt, dass das logische Schließen die Situation nicht erkannt hat, da hier nicht alle Evidenzen wahr sind. Die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ steigt gleich zu Beginn aufgrund eines verdächtigen Schiffstyps. Zum Zeitschritt 9 wird erkannt, dass das Schiff in einem verdächtigen Gebiet unterwegs ist, wodurch die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ auf ein Maximum von 0,8047 steigt. Die Evidenz von einem Rendezvous ist hier nicht wahr. Da diese Evidenz am stärksten für die Situation eines Schmuggelbootes spricht, wird nur eine Wahrscheinlichkeit von 80,47% erreicht. Die erreichte Wahrscheinlichkeit ist verdächtig, weist aber nicht ausreichend auf ein Schmuggelboot hin.

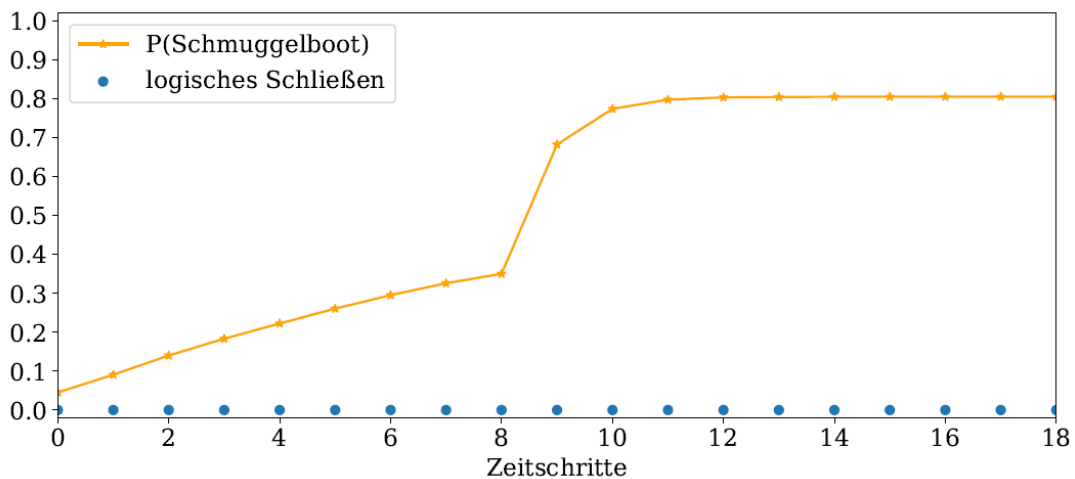


Abbildung 3-55: Ergebnisse des Schiffes mit der MMSI-Nummer 1234.

In der Abbildung 3-56 sind die Ergebnisse des Schiffes mit der MMSI-Nummer 4500 zu sehen. Hier wurde laut dem logischen Schließen kein Schmuggelboot erkannt. Auch die Wahrscheinlichkeit von maximal 0,6513 ist im Vergleich zu den vorherigen Ergebnissen nicht sehr hoch, weswegen das Ergebnis eher für eine geringe Existenzwahrscheinlichkeit für ein Schmuggelboot spricht. Bei diesem Schiff existiert nur die elementare Situation, bei der erkannt wurde, dass das Schiff in einem verdächtigen Gebiet war.

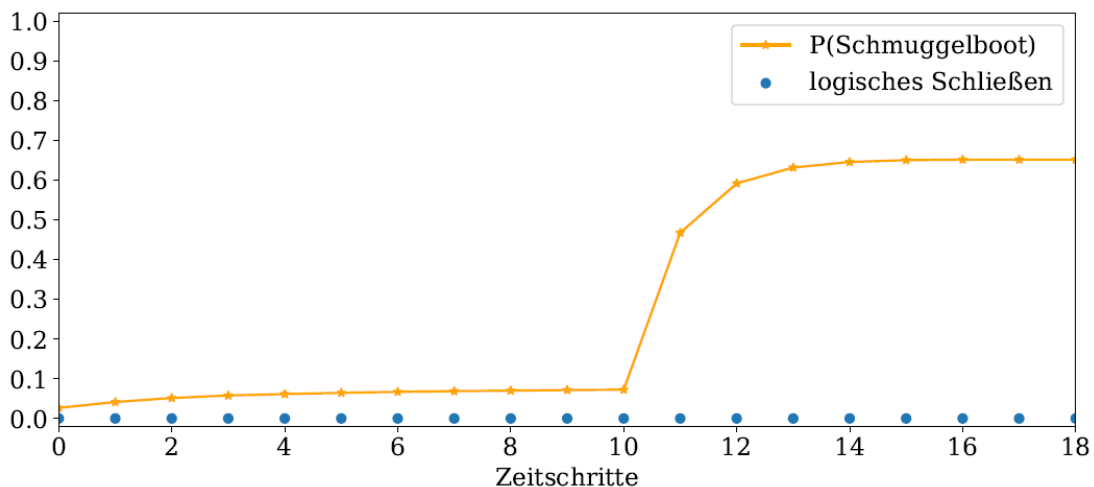


Abbildung 3-56: Ergebnisse des Schiffes mit der MMSI-Nummer 4500.

Die Abbildung 3-57 zeigt abschließend die Ergebnisse des Schiffes mit der MMSI-Nummer 6300. Das Schiff zeigt keine Auffälligkeiten für ein Schmuggelboot, da alle Evidenzen der elementaren Situationen nicht wahr sind. Die Wahrscheinlichkeit von $P(\text{Schmuggelboot})$ erreicht maximal den Wert von 0,0731.

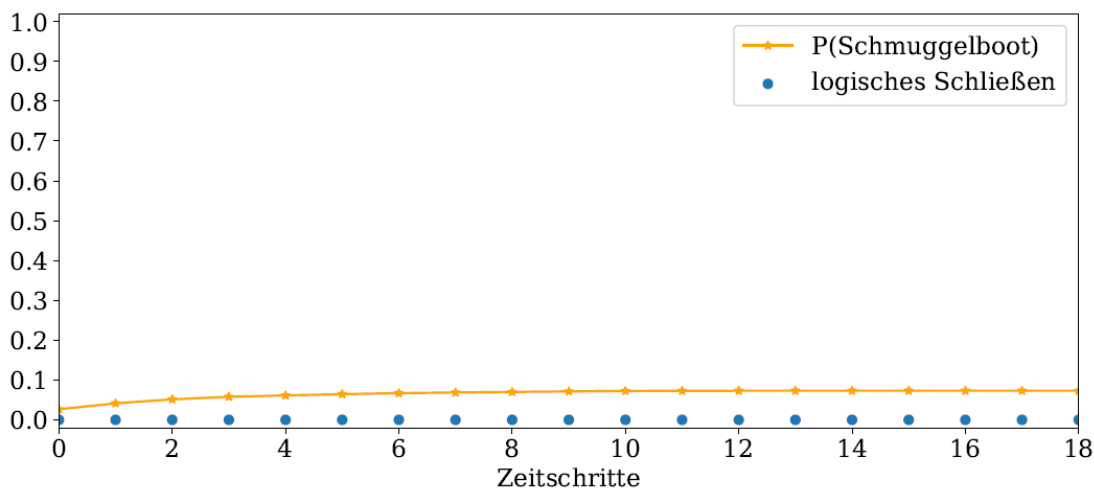


Abbildung 3-57: Ergebnisse des Schiffes mit der MMSI-Nummer 6300.

Anhand der vorgestellten Ergebnisse und mit dem nicht betrachteten Fall, dass bei den elementaren Situationen für ein Rendezvous (E^1) und für einen verdächtigen Schiffstyp (E^3) eine maximale Existenzwahrscheinlichkeit von 0,8768 erreicht wird, kann ein passender Schwellwert bestimmt werden. Wahrscheinlichkeitswerte, die größer oder gleich des Schwellwerts sind, weisen eindeutig auf die Existenz der Situation hin. Für dieses DBN kann der Schwellwert auf 0,81 gelegt werden. Damit werden alle Situationen erkannt, die mit einem Rendezvous zusammen hängen. Der Schwellwert kann auch mit der Berechnung des harmonischen Mittel aus Genauigkeit und Trefferquote bestimmt werden, wie Fischer in (Fischer, 2016) zeigt. Vor allem bei komplexeren DBNs bietet sich dieses Vorgehen an.

Evaluation mit realen AIS-Daten

Die prototypische Realisierung wurde auch mit realen AIS-Daten getestet, die live ins System eingespielt wurden. Die AIS-Daten wurden mit einem Polygon in der deutschen Bucht gefiltert, das Häfen, Flüsse und enge Flussmündungen ausschließt. Bei einem zweistündigen Durchgang wurden AIS-Daten von über 100 Schiffen analysiert und bewertet. In dieser Zeit wurde kein Rendezvous erkannt und auch keine hohe Existenzwahrscheinlichkeit für ein Schmuggelboot erreicht. Das Erkennen eines Schmuggelboots mit einer reinen Analyse von AIS-Daten konnte hier in einem kurzen Zeitraum von zwei Stunden nicht nachgewiesen werden. Für eine genauere Evaluierung müsste eine Analyse von mehreren Tagen oder Wochen durchgeführt werden. Zum einen kann aus den erreichten Ergebnissen mit realen AIS-Daten geschlossen werden, dass unter Annahme eines normalen Schiffsverkehrs ohne Schmuggelaktion, die offene See und die Konfiguration zu Erkennung eines Rendezvous richtig definiert wurden. Dadurch sind keine fehlerhaften Erkennungen eingetreten. Zum anderen kann gesagt werden, dass hierfür eine reine Analyse von AIS-Daten nicht ausreichend sein kann und zusätzliche maritime Datenquellen für die Erkennung einer Schmuggelaktion eingesetzt werden sollten. Ein Grund dafür ist, dass Schiffe ihre AIS-Sensorik auch ausschalten können. Durch Detektion mit anderen Sensoren würden diese Schiffe aber wieder verdächtig werden.

Zusammenfassung der Ergebnisse

Anhand der Evaluation von Robustheit und Sprungverhalten des DBNs konnte das Verhalten bezüglich fehlerhaften Beobachtungen aufgezeigt und eine gewisse Trägheit dargelegt werden. Die Ergebnisse dieser Untersuchung zeigen auch, dass mit der prototypischen Realisierung die vordefinierte Situation eines Schmuggelboots anhand eines beispielhaften Szenarios erkannt wurde. Mit der Festlegung eines passenden Schwellwerts kann eine sichere Aussage über die Existenz der Situation gemacht werden. Die Existenzwahrscheinlichkeit kann beispielsweise mit einer unterschiedlich farbigen Markierung im maritimen Lagebild visualisiert werden, um ein besseres maritimes Situationsbewusstsein zu erreichen. Interessant wäre es, wenn reale Daten zur Verfügung stehen würden, bei denen eine Situation von Interesse tatsächlich eingetreten ist. Damit könnte das implementierte DBN zur Erkennung eines Schmuggelboots mit echten Daten verifiziert werden. Die Situation von Interesse könnte beispielsweise mit realen Schiffen nachgestellt werden. Anhand der Ergebnisse lässt sich feststellen, dass mit einem DBN und mit Verwendung von simulierten AIS-Daten eine beispielhafte maritime Situation korrekt erkannt werden konnte.

3.2.7.5. Fazit

In der maritimen Situationserkennung stellen sich DBNs sowie BNs als nützliche Methodik dar. Diese Modelle können mit reinem Expertenwissen erstellt werden und Situationen erkennen, die nicht direkt beobachtbar sind. Vor allem dort, wo keine ausreichende Datengrundlage besteht oder eine erklärbare Methodik für kritische Anwendungen benötigt wird, ist der Einsatz von DBNs sinnvoll. Mit der prototypischen Realisierung wurde eine probabilistische Situationserkennung zur Erkennung von Schmuggelbooten auf Basis von AIS-Daten implementiert. Dafür wurde ein DBN verwendet, mit dem Schlussfolgerungen unter Einbeziehung von Unsicherheit über die Zeit gemacht werden können. In der Arbeit wurde auf die Repräsentation von DBNs eingegangen, Methoden zur Bestimmung der Parameter aufgezeigt und Ansätze bei der Modellierung vorgestellt. Bei der Situationsmodellierung muss darauf geachtet werden, dass eine passende Anzahl an Situationen bzw. Knoten im Situationsnetzwerk modelliert wird, um die Existenz einer Situation von Interesse ausreichend zu beschreiben und die Anzahl der zu bestimmenden Parametern möglichst gering zu halten. Durch die Umsetzung des Konzepts zur prototypischen Realisierung konnten die Funktionalitäten eines DBNs in ein Echtzeitsystem mittels Apache Kafka integriert werden. Mit dem verwendeten Tooling können zum einen DBNs implementiert und zum anderen die Situationserkennung anhand der Evaluierungsergebnisse entsprechend getunt werden. Für eine korrekte Situationserkennung mittels der DBNs spielen auch die Analyse-Prozesse und deren Erkennungsqualität eine wesentliche Rolle. Sind diese gut auf die zu erkennende Situation abgestimmt, so kann das DBN korrekte Ergebnisse zur Existenz einer Situation ermitteln.

Zusammenfassend kann gesagt werden, dass die Zielsetzung der Masterarbeit mit einer prototypischen Realisierung erreicht werden konnte und somit eine korrekte Erkennung einer maritimen Situation mittels DBNs und mit Verwendung von simulierten AIS-Daten aufgezeigt werden konnte. In der Masterarbeit wurden zum einen die DBNs zur Erkennung von maritimen Situationen ausführlich betrachtet und zum anderen ein Konzept für die Situationserkennung mittels DBNs vorgestellt, welches mit aktueller Technologie realisiert wurde und sich für Echtzeitsysteme eignet.

3.3. AP 2100 - Service Concepts

3.3.1. AP 2110 - User Requirements (International)

Als nationales Forschungsvorhaben standen bei IntelliMar mit der Bundespolizei See ein deutscher Bedarfsträger im Vordergrund, der im Vorfeld der technischen Entwicklungsarbeiten sehr detailliert Anomalien aus der Praxis beschreiben hat, die auf illegale Handlungen schließen lassen. Auf diese Weise könnten die Algorithmen an akute Bedarfe erprobt werden. Aus Sicht der kommerziellen Dienste-Entwicklung und der Vermarktung im globalen Markt war es darüber hinaus von starkem Interesse, eine internationale Stakeholder-Community zu befragen und zu prüfen, inwieweit die nationalen Anforderungen sich auch im globalen Umfeld widerspiegeln.

Die Befragung der Endnutzer hatte folgendes Ziel:

- Erfassung des Informationsbedarfs
- Erfassung der relevanten typischen Nutzerszenarien
- Ableitung der Systemperformance
- Gap-Analyse (Wo liegen Informationslücken?)
- Erweitertes Projektziel: Relevanz der IntelliMar Ergebnisse für internationale Nutzer

3.3.1.1. Methodik der Bedarfserfassung

Die Bedarfserfassung richtete sich an maritime Organisationen mit sicherheitsrelevanten Aufgaben im internationalen Umfeld. Der Fragenkatalog umfasst mit allgemeinen Informationen, Informationsanforderungen und Dienste-Anforderungen drei Felder, die detaillierten Fragen zu den jeweiligen Themen enthalten. Der Gesamtumfang der Fragen und Antwortmöglichkeiten wurde begrenzt, so dass die Beantwortung nicht länger als 45 Minuten in Anspruch nehmen sollte.

1. General Information

- *Name and Type of your Organization*
- *What is the type of your Organization?*
- *What is your position in your Organization?*

2. Information Needs

- *In which field does your organization operate?*
- *Which automated behavior monitoring services provided by the European Maritime Safety Agency (EMSA) do you use?*
- *Please indicate which anomalies are relevant for your operations?*
- *Which vessel parameters are of interest to you?*
- *Which Information that would be required for your operational task does not exist today?*

3. Service Requirements

- *Which is the preferred monitoring scenario?*
- *What is the expected age of information in the maritime picture?*
- *Which functionalities do you expect from a maritime surveillance system?*

- *Efficiency of Work: How would you describe the distribution of time between A) Mental work B) Operating the system and C) System?*
- *In order to fulfil your task more efficiently, what would be the ideal innovation?*
- *Do you have any general comments?*

Erleichterung der Bearbeitung des Fragebogens: Die Beantwortung der Fragen wurde soweit wie möglich durch das interaktive Anklicken von möglichen Antworten erleichtert. Darüber hinaus wurde bei Multiple-Choice Fragen auch die Möglichkeit gegeben, wo sinnvoll, Freitext einzutragen, sollte keine der vorgegebenen Antworten in Frage kommen.

Ermittlung der minimal notwendigen Anforderungen: Für die weitere Verwertung der Antworten im Rahmen der Systementwicklung ist es von zentraler Bedeutung, die minimal und maximal Anforderungen zu kennen, um das System einerseits ausreichend groß zu dimensionieren aber auch andererseits im Kostenrahmen zu halten. Zu diesem Zweck wurden Fragen, die eine Einschätzung zur Wichtigkeit adressierten, mit entsprechenden Antwortmöglichkeiten versehen, die eine realistische Bedeutung des jeweiligen Parameters zulassen.

*Which vessel parameters are of interest to you?

	Mandatory	Nice to have	No interest
Geolocation Y (X Coordinate)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Length	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Width	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Heading	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Speed over ground (SOG)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Destination	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estimated Time of Arrival ETA	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Historical routes	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Ship Flag	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Abbildung 3-58: Erfassung der Wichtigkeit von Schiffparametern (Beispiel)

Maximierung der Rücklaufquote: Um eine möglichst hohe Rücklaufquote zu forcieren, wurde der Online Fragebogen durch die Bundespolizei See im Rahmen regelmäßiger Koordinationstreffen an die Mitgliederorganisationen der ECGFF und NACGF kommuniziert und versandt. (siehe Kapitel 3.3.1.2) – Link zu der Umfrage: <https://survey.intelligence-airbusds.com/index.php/17941?newtest=Y&lang=en>)

3.3.1.2. Übersicht der Nutzergruppen

Für die Nutzerbefragung wurden Mitglieder des European Coast Guard Functions Forum (ECGFF) sowie das North Atlantic Coast Guard Forum (NACGF) einbezogen.

Das European Coast Guard Functions Forum ist ein unverbindliches, freiwilliges, unabhängiges und nicht politisches Forum. Zu den Mitgliedern zählen Küstenwachbehörden von 25 EU-Mitgliedstaaten und assoziierten Schengen-Ländern sowie die Europäische Kommission und ihre Institutionen und Agenturen mit entsprechenden Kompetenzen in Küstenwachfunktionen. ECGFF wurde 2009 ins Leben gerufen und wird von der EU co-finanziert.

Das North Atlantic Coast Guard Forum (NACGF) erweitert die Internationale Zusammenarbeit in Nordamerika und fördert die Zusammenarbeit und den Informationsaustausch zwischen Behörden mit Küstenwacheinsätzen.

Beide Organisationen umfassen 43 Länder und sind in der folgenden Graphik grün hervorgehoben.

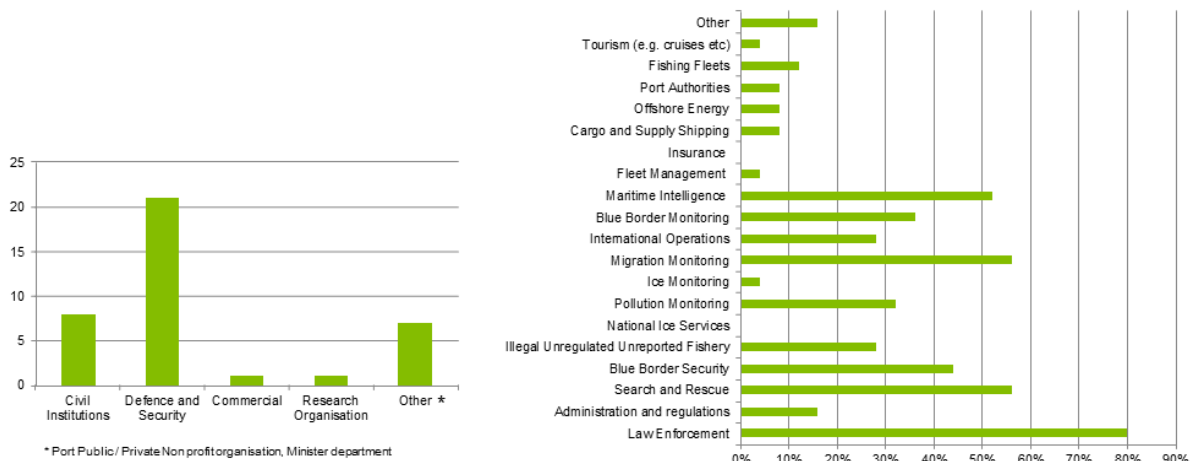


Abbildung 3-59: Mitglieder des ECGFF und NACGF

Feedback: Aus den 22 Ländern (incl. Frontex) sind 40 individuelle Antworten eingegangen. Das entspricht mehr als 500 beantwortete Fragen.

3.3.1.3. Ergebnis der Bedarfserfassung

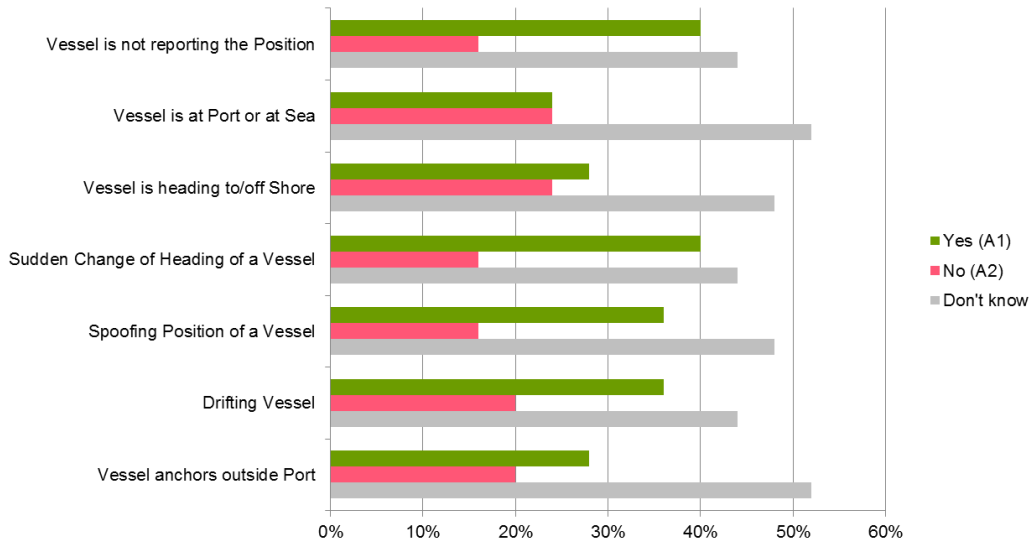
Frage: **Allgemeine Informationen & Nutzerprofil**



Die Mehrzahl der befragten Organisationen ist dem Bereich Verteidigung und Sicherheit zugeordnet, gefolgt von kommerziellen maritimen Unternehmen, die ein weites Aufgabenspektrum abdecken. Strafverfolgung, Such- und Rettungseinsätze und Migrationsüberwachung an der „blauen Grenze“ zählen zu den häufigsten Anforderungen.

2. Information Needs

Frage: *Which automated behavior monitoring services provided by the European Maritime Safety Agency (EMSA) do you use?*



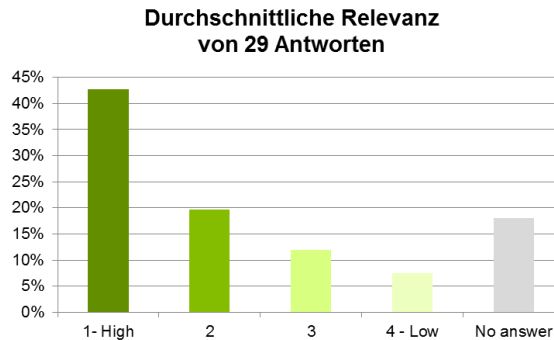
Aktuell bietet die European Maritime Safety Agency (EMSA) operationelle und automatische Anomalie-Erkennungsdienste an, die von Europäischen Mitgliedsstaaten kostenlos genutzt werden können. Aus Projektsicht war es daher wichtig zu verstehen, welche Anomalien bekannt und auch genutzt werden, um im Rahmen dieses und folgenden Forschungsprojekten den Fokus auf neue bzw. schwer zu erfassende Anomalien auszurichten. Das Ergebnis zeigte, dass die sieben Anomalien bei ca. der Hälfte (50 %) der Befragten nicht bekannt waren. Ca. 20 % der Nutzer sind durchschnittlich an den angebotenen Anomalien nicht interessiert, und zwischen 30 und 40 % der Nutzer nehmen den Anomaliedienst der EMSA in Anspruch.

Frage: *Please indicate which anomalies are relevant for your operations?*

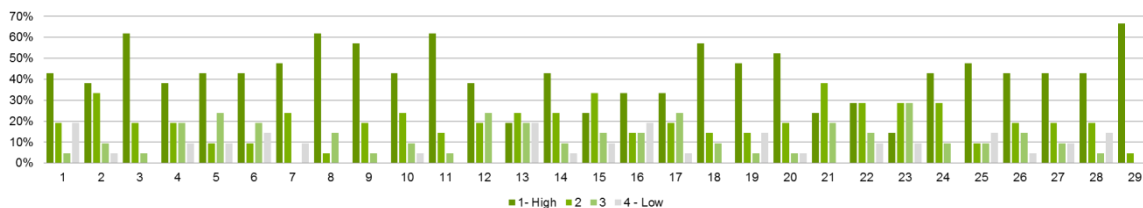
- Vessel proceeding with course and speed in a manner which is clearly uneconomic.
- Small fast boat proceeding at high speed in open sea
- Vessel loitering without apparent or obvious legitimate navigational reason (distress, technical difficulties, etc.) in open sea or in close vicinity of any territorial waters or coastal area.
- Vessel maneuvering to avoid contact.
- Sailing in sensitive areas or in sensitive seaways with maritime infrastructure. E.g.: Sailing along underwater cables/pipelines etc.
- Vessel not taking action to avoid collision (IRPCS).
- Sudden splitting into more than one contact.
- Two or more vessels approach unusually close to each other in international waters
- The new position or travelled distance after an AIS signal loss does not appear consistent to the previous route (probably a hidden maneuver).
- The same two vessels are sailing a route / maneuver (repeatedly within X months) that indicates a delayed drug delivery (considering the timeframe and drift corridor).
- Vessel is switching off AIS signal
- Vessel are actively fishing in a no-fishing zone
- Vessel enters a particular zone the first time
- Listed Vessels of international interest for further investigation Examples: IUU List, Paris MOU/Sirenac, EMSA banned ships, etc.
- Drastic (> 30°) change of course at open sea
- Alteration to common routes Example: Ferry far of the usual path
- Appearance of vessels which are untypical for the area (e.g. open boat at the high seas, large vessel in narrow/shallow waters...)
- Detection and early warnings for vessels posing a potential risk, arriving at specific areas or following unusual routes.
- tracking for vessels involved in illegal activities, smuggling prohibited substances, or breaking embargoes
- detection of vessels approaching shores to transfer goods or persons illegally
- Small fast boat proceeding at high speed in open sea
- Heading for the coast / the beach outside ports / jetties / moorings
- Launching dinghies outside SAR- / MOB-Situations Exceptions: At a shipyard or near ports / jetties / moorings
- Vessel changing ship's name, port of registry, IMO number or funnel markings at sea
- Ship's name, IRCS, shipping company not registered in world shipping lists or databases
- Double IMO-number, MMSI, Call sign, in the situation picture
- The same ship (identical IMO) has changed its name several times within one year (typical behavior for drug boats).
- A vessel has reported a route (e.g. passed ports in the Security Report) that appears wrong or not plausible / possible for this type of ship (with typical capabilities).
- Detection of dark vessels by earth observation. Ship is not sending an AIS Signal

Die Bundespolizei See hat eine umfassende Liste von Anomalien aus der Praxis formuliert und dem Projektteam zur Verfügung gestellt. Im Fragebogen konnten die Befragten die

jeweilige Wichtigkeit auf einer Skala von 1 (hohe Wichtigkeit) bis 4 (geringe Wichtigkeit) einordnen, so dass mit der Auswertung dieser Frage im Kern beantwortet werden kann, ob die Anomalieentwicklung in IntelliMar den Bedarf der internationalen Nutzer abdeckt.



Die Bildung eines Durchschnittswertes verdeutlicht eine hohe Relevanz (1 – 43 % und 2 – 20 %) aller Anomalien im internationalen Umfeld.

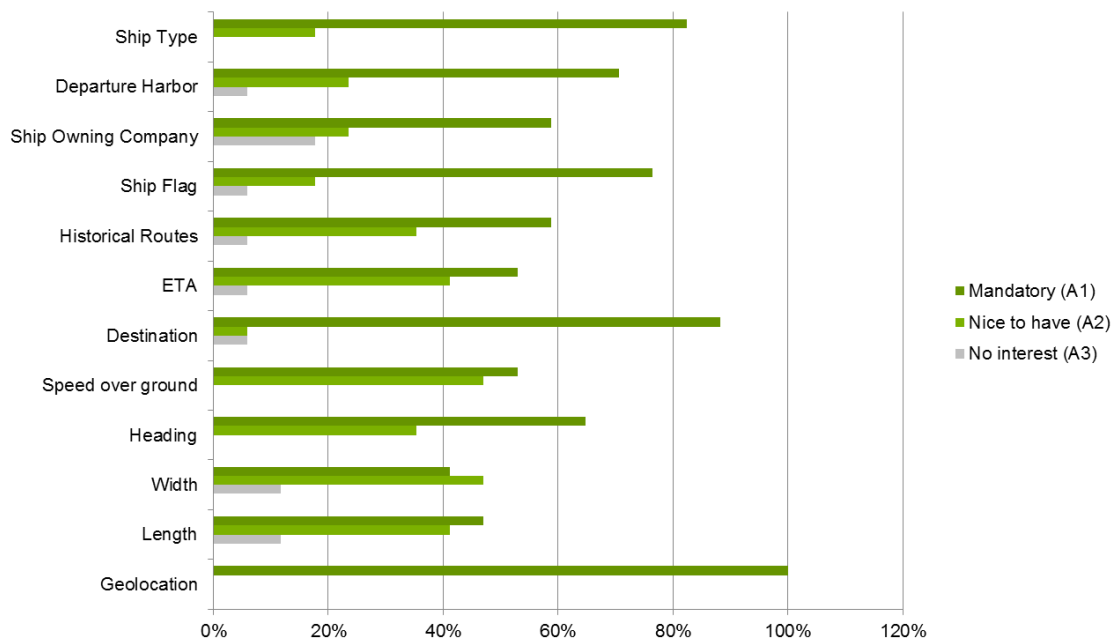


In der individuellen Betrachtung der 5 Anomalien mit höchster Relevanz konnte folgende Ergebnis festgestellt werden: Die Anomalien Nr. 29, 11 und 8 wurden in IntelliMar als Grundlage für die automatische Anomaliedetektion verwendet. Dadurch kann hiermit festgestellt werden, dass die Entwicklungsarbeiten auch für internationale Sicherheitsbehörden von großem Interesse sind.

Top 5 Anomalien mit hoher Relevanz

- Nr 29** Detection of dark vessels by earth observation Ship is not sending and AIS Signal
- Nr 11** Vessel is switching off AIS signal
- Nr 8** Two or more vessels approach unusually close to each other in international waters
- Nr 3** Vessel loitering without apparent or obvious legitimate navigational reason (distress, technical difficulties, etc.) in open sea or in close vicinity of any territorial waters or coastal area.
- Nr 18** Detection and early warnings for vessels posing a potential risk, arriving at specific areas or following unusual routes.

Frage: Which vessel parameters are of interest to you?



Bei der Frage zu den Schiffsparemtern, die aus Sicht der Nutzer von Interesse sind, wurden drei Auswahlmöglichkeiten zur Verfügung gestellt. Mit „Mandatory“ (verpflichtend), Nice-to-have (wünschenswert) und No Interest (kein Interesse) konnte somit die Priorität der Parameter ermittelt werden. Aus Sicht fast aller Befragten sind auf diese Weise die genaue Ortsangabe (Geolocation), der Zielort (Destination) und der Schiffstyp (Ship Type) absolute wichtige Parameter eines Schiffes, die in ein Lagebild gehören. Mit abfallender Häufigkeit antworten mehr als die Hälfte der Nutzer, dass der Herkunftshafen, die Schiffsflagge sowie Heading (Fahrtrichtung des Schiffes) wichtig sind.

Frage: Which Information that would be required for your operational task does not exist today?

Diese Frage konnte mit Freitext beantwortet werden. Die Originalantworten sind im Folgenden aufgeführt:

- *Information from sensors other than AIS and LRIT*
- *Total number of persons onboard*
- *System integration between national systems and e.g. EMSA system.*
- *Data about vessel technical specification (length, width, type)*
- *Connection of law enforcement registries (national, European) to crew / pax list and this way to vessel (IMO number)*
- *Crew- and Passenger list & nationalities*
- *Cargo on board, Cargo Manifest*
- *Historical routes, Track History*
- *Comprehensive sensors coverage of coastline out to at minimum EEZ, integrated into*

a command/display/communications system common to all maritime security partners.

3.3.2. AP 2120 - Service Requirements (International)

Für die Entwicklung von Service Konzepten im Bereich Maritime Sicherheit bzw. Überwachung ist es zielführend, zunächst die typischen generischen Grundbedürfnisse der Anwender zu beschreiben. Nach diesen Aspekten richtet sich die grundsätzliche Konzeption der Dienste.

Zeitlichkeit:

Dienste innerhalb eines maritimen Beobachtungsszenarios bringen meist starke Anforderungen an die zeitliche Reaktivität mit sich. Dabei ist sowohl die Lieferzeit des zu erstellenden Produktes wichtig, als auch die zeitliche Wiederholrate eines maritimen Lagebildes. Viele Nutzer aus dem Bereich küstennaher, bspw. mitteleuropäischer Einsatzgebiete sind sogar Echtzeit-Anwendungen gewohnt wie Lagebilder auf Basis von terrestrischen Überwachungssystemen. Satelliten-basierte bildgebende Verfahren allerdings liefern, wenn auch anhand von Momentaufnahmen, eine unverfälschte und nicht wie z.B. AIS manipulierbare Übersicht – auch in Bereichen fernab landgestützter Sensorik.

Qualität:

Die Nutzbarkeit der bereitgestellten Dienste hängt davon ab, ob die spezifischen Anwender-Bedarfe gedeckt werden können. Üblicherweise ist bei einer maritimen Überwachungslage der Detailgrad der gelieferten Information ebenso wichtig wie eine für den Nutzer akzeptable Flächenabdeckung. Die im Rahmen dieses Projektes erhobenen Statistiken zur Anforderungsanalyse durch typische Nutzergruppen ergab ein hohes Maß an Interesse, möglichst viele Details v.a. Objektparameter zu extrahieren und in das Lagebild zu integrieren. Zeitlichkeit, wie im Punkt zuvor erwähnt, kann zudem auch ein Qualitätskriterium sein. In besonderer Art und Weise ist jedoch in den meisten Fällen die wichtigste Qualitätskomponente die Verlässlichkeit der für den Nutzer bereitgestellten Information. Dabei ist es ein Gradmesser mit welcher Sicherheit beispielsweise bestimmte Objekte detektiert, vermessen, klassifiziert oder einem AIS Signal zugeordnet werden können.

Ergonomie und Nutzerspezifizierung:

Die in der Ausrichtung, Auflösung und Abdeckung flexiblen Einsatzarten von Erdbeobachtungssatelliten, insbesondere Radarsensoren, bedeutet auch, dass Betreiberseitig sowohl ein genaues Verständnis der Kundenbedarfe, als auch das Expertenwissen um die Konfiguration der Satelliten Programmierung bestehen muss. Klassischerweise wird dies über Kundendienst-Schnittstellen gelöst. Dies jedoch bringt oftmals gewisse Verzögerungen mit sich. Nutzern ist also geholfen, wenn simple direkte Möglichkeiten bestehen, Satellitenaufnahmen so zu planen, dass die erzeugten Bilddaten in optimaler Art und Weise in einen Service eingegliedert werden.

3.3.2.1. Optimierung der Zeitlichkeit

In Kapitel 3.3.3.3 wird besonders betrachtet, wie auf der Satellitenbild-Auswerter-Seite die Servicebereitstellung beschleunigt werden kann. Zeitersparnisse im gesamten Service

Zyklus jedoch sind gleichermaßen hilfreich für eine optimierte Dienstleistung. Also sind innovative Automatisierungen im Bestellprozess oder bei der Satellitenauswahl und Programmierung ebenfalls wertvolle Stellschrauben für einen gesteigerten Wert eines Services für den Nutzer.

Ein im Rahmen dieses Projektes weiterentwickeltes Konzept fokussiert sich auf eine solche Automatisierung, die es erlaubt, übliche Szenarien auf Nutzerseite in festgelegte Sub-Prozesse zu übersetzen, die basierend auf zuvor abgefragten Nutzerszenarien automatisch zum Einsatz kommen könnten. Einer der Schlüssel zur Optimierung der Handhabung des Bestellprozesses ist eine deutliche Simplifizierung der komplexen Routinen, wie sie innerhalb eines Systems wie dem Service Segment von TerraSAR-X üblich sind. Diese werden nicht nur für maritime Anwendungen genutzt und beinhalten daher Parameter und Einstellungen, welche in der Regel nur für terrestrische Zwecke benötigt werden. Das erdachte Konzept sieht daher vor maritime Nutzerszenarien in verschiedene Parameter-Konfigurationen zu übersetzen und diese mit Bezeichnungen zu versehen, die bereits den Anwendungshintergrund widerspiegeln. Diese Überlegung ist Sensor-unabhängig. Beispielsweise könnte ein Konzept „Non-commercial Fishing“ zum Einsatz kommen. Eine Auswahl dieser Option durch den Kunden würde dann automatisch erwirken, dass die Wahl des Sensor-Modus zwingend auf eine höher aufgelöste Konfiguration fällt, da üblicherweise nicht kommerzielle Fischerei in kleinen Fahrzeugen geschieht. Die Sensor Auflösung ist jedoch nur eine Stellschraube, ein Szenario zu charakterisieren. Tabelle 3-31 zeigt am Beispiel von TerraSAR-X typische Konfigurationsmöglichkeiten und stellt vereinfacht deren Effekt auf die maritime Nutzbarkeit der so erhobenen Daten dar.

Sensor Parameter	Wahlmöglichkeit	Effekt
Auflösung	Hohe Auflösung 3 m	TerraSAR-X StripMap (SE spatially enhanced) für Klassifikation
	Hohe bis mittlere Auflösung 7 m	TerraSAR-X StripMap (RE radiometrically enhanced) für Klassifikation
	Mittlere Auflösung 18.5 m	TerraSAR-X ScanSAR – für Objekt Erkennung
	Mittlere bis grobe Auflösung 40 m	TerraSAR-X Wide ScanSAR – für Objekt Detektion
Prozessier-Level	RE	Radiometrisch optimiert - über Wasser grundsätzlich empfehlenswert
	SE	Auflösung optimiert – nur bei sehr ruhiger See ein leichter Vorteil
Polarisation	VV	Wellenmuster gut sichtbar
	HH	Objekte und Wellenmuster sichtbar
	HV	Gute Unterscheidbarkeit Objekt/Wasser,

		keine Wellenmuster erkennbar
Aufnahmewinkel	14° - 20°	Steile Winkel: Heckwellen sichtbar, Wellengang sichtbar, Bestimmung der Bewegungsrichtung, leicht reduzierte effektive Auflösung
	20° - 45°	Mittlere Winkel: Gute Grundeinstellung. Gute generelle Objekterkennung, Öl-Verschmutzung erkennbar
	45° - 60°	Flache Winkel: Leicht erhöhte theoretische Auflösung, reduzierte Radiometrie, keine Ölfilm Detektion, erschwerte Bewegungserkennung
Produkt Typ	Multi Look Ground Range Detected MGD	Schnell verfügbar
	Geocoded Ellipsoid Corrected GEC	Ortsgenau
	Enhanced Ellipsoid Corrected EEC	Ortsgenauest
Schwadbreite	Mittel	30 km (TerraSAR-X StripMap)
	groß	100 km (TerraSAR-X ScanSAR)
	sehr groß	bis 270 km (TerraSAR-X Wide ScanSAR)

Tabelle 3-31: Sensor Parameter und maritime Empfehlungen

Bestehende Software-Lösungen wie SaVoir (Taitus Software) bieten Schnittstellen zur nutzerseitigen Planung von Aufnahmen verschiedenster kommerzieller und nichtkommerzieller Satelliten an, jedoch fehlt eben jene anwendungsspezifische Filtermöglichkeit für die Aufnahmekonfiguration. Ziel des erarbeiteten Konzeptes ist es, dass der Nutzer sich bestenfalls gar nicht mit den technischen Stellschrauben beschäftigen muss, sondern einen Request aus einer Liste vordefinierter Anwendungsprofile auswählt und dabei das optimierte Produkt erhält. Dies beschleunigt den Bestellvorgang immens und trägt dazu bei, End-to-End Prozesse zu optimieren und Fehlbestellungen zu vermeiden. Eine Kategorisierung der Anwendungsmöglichkeiten erlaubt es auch, verschiedene Sensoren zu kombinieren und diese in entsprechend passender Konfiguration in Gruppen einzuteilen. Bspw. kann ein Sentinel 1 A/B in Interferometric WideSwath Konfiguration ein sehr ähnliches Profil bedienen wie bspw. TerraSAR-X ScanSAR. Eine Bestellplattform könnte so Sensoragnostisch genutzt werden.

Die Zeitersparnis bei der Produktauswahl ist von großem Wert, da in Notfallsituationen nicht selten das letztmögliche Zeitfenster zur Programmierung des Satelliten verstreicht, wenn dessen Aufnahmekonfiguration nicht geklärt werden kann. Ein prototypisch entwickelter vollautomatischer Wizard (vergl. Abbildung 3-60) kombiniert, basierend auf Nutzereingaben,

die in Tabelle 3-31 gelisteten Parameter intelligent und gibt eine optimierte Sensorauswahl – und Konfiguration aus.

Parameter	Value	Next Step	Instruction
MINIMUM VESSEL SIZE (What is the smallest vessel size to be detected)	15-40m	NEXT STEP	FOLLOW INSTRUCTIONS BELOW
ORBIT DIRECTION (Ascending or Descending)	ascending	COMPLEXITY INDICATOR	Operational
LOOK DIRECTION (Left or Right)	left	MODE	TerraSAR-X ScanSAR
INCIDENCE ANGLE (Angle Range)	>20°<45°	PROCESSING LEVEL	RE
APPLICATION (Customer Use Case)	Vessel Detection [VDR]	PRODUCT TYPE	MGD
EMERGENCY TASKING (No time to contact maritime experts)	No	POLARISATION	HH
INFORMATION AGE [h] (Maximum acceptable delay between acquisition time and service delivery)	4	ORBIT EVALUATION	Orbit ok
DOWNLINK DELAY [h] (Estimation of downlink delay)	1	ORBIT POSITION	Beam configuration ok
AOI TYPE (Spatial relation of AoI regarding and water and terrain)	water only		
COASTAL INFLUENCE (If close to the coast, in which direction is the nearest coastline)	E of AoI		
COASTAL PROXIMITY (How far away is the coast?)	> 60 km		
COVERAGE FLEXIBILITY (Possibility to shift footprint)	possibility to shift E/W		
VESSEL TRAFFIC ESTIMATE (Estimate vessel traffic density)	low		

Abbildung 3-60: Prototypischer Wizard zur Unterstützung bei der Sensor Auswahl und – Konfiguration

3.3.2.2. Optimierung der Qualität

Maritime Informationsprodukte müssen oft in sehr kurzer Zeit erstellt werden, wobei auch Vorlaufzeiten meist auf ein Minimum reduziert sind. Diese Anforderungen begünstigen aufgrund der benötigten Reaktivität vollautomatische Dienstkonzepte. Werden die Anwendungsbereiche jedoch sensitiv oder unterliegen Sicherheitsaspekten, stellt die Einbindung von menschlichen Operatoren in die Prozesskette ein zusätzliches, oft unverzichtbares Qualitätsmerkmal dar. Zudem ist die vollständige Automatisierung ein schwer zu erreichendes und kostspieliges Ziel, vor allem, wenn zugleich hohe Qualitätsansprüche an die Informationsgüte gestellt werden. Unabhängig davon, ob innerhalb eines maritimen Dienstes eine Operator-Einbindung stattfindet oder nicht, ist es empfehlenswert, jegliche automatisch ausgegebene Information mit einem Qualitätsindikator zu versehen, sei es für den Nutzer direkt oder für den Operator zur weiteren manuellen Lagebilderstellung.

Die innerhalb des Projektes entwickelten Konzepte zielen daher darauf ab, mögliche Faktoren, die die Qualität der extrahierten Information beeinflussen könnten, zu identifizieren, diese abzuschätzen und letztlich als Berechnungsgrundlage für eben solche


Qualitätsindikatoren miteinzubeziehen. Der in der Schiffsdetektion wichtigste Parameter, die Detektionswahrscheinlichkeit oder Detektionsgüte in SAR Bilddaten, hängt zum Beispiel von verschiedenen Faktoren ab, die beispielhaft in Tabelle 3-32 gezeigt werden:

(I) Objekt Eigenschaften					
Objektgröße in Relation zur Auflösung					
klein	% ↓	mittel	% →	groß	% ↑
Objektprofil					
flach	% ↓	profiliert	% ↑	stark profiliert	% ↑
Objektverhalten					
steht still	% ↓	bewegt sich	% ↑	schnell	% ↑
(II) Sensor Eigenschaften					
Aufnahmewinkel					
steil	% →	intermediär	% ↑	flach	% ↓
Artefakte / Ambiguitäten					
keine	% ↑	gering	% ↓	stark	% ↓
(III) Externe Einflüsse / Kontext					
Wellenhöhe					
gering	% ↑	mittel	% →	hoch	% ↓
Meteorologische Effekte					
keine	% ↑	gering	% ↓	stark	% ↓
AIS Signal oder RF Signal Korrelation					
nein	% →	unsicher	% ↑	sicher	% ↑
Batymetrie					
sehr flach	% ↓	flach	% ↓	schiffbar	% →


Tabelle 3-32: Einfluss von verschiedenen Faktoren auf die Detektierbarkeit von Objekten in maritimen SAR Daten. Pfeile geben eine Indikation des Einflusses an: positiv (↑), negativ (↓) oder neutral (→)

Viele dieser Einflüsse liegen (1) zum Zeitpunkt der Lagebild-Erstellung als gegeben vor (Bsp. Sensorauflösung). Andere wiederum können (2) aus technischer Sicht automatisch extrahiert werden (Bsp. Wellenhöhe), oder (3) können durch Fusion von Informations-Ebenen bestimmt werden (Bsp. AIS Verschneidung). Diese Tatsache erlaubt es, automatisch erzeugte und empirisch überprüfbare Qualitätsindikatoren (bspw.


Detektionswahrscheinlichkeiten in %) zu bestimmen und den maritime Service-Produkten (z.B. Schiffsdetektion-Reports) beizufügen. Ist die Logik hinter solchen Werten transparent und prüfbar, können diese dem Nutzer helfen die Ergebnisse von maritimen Lagebildern nicht nur schnell entgegenzunehmen, sondern zugleich *bewerten* zu können. Wie im Einzelnen Qualitätsindikatoren erstellt und geprüft werden, war nicht Teil der Aktivitäten im Projekt. Ein Beispiel verdeutlicht jedoch den möglichen Einsatz-Rahmen eines solchen Konzeptes: In Abbildung 3-61 wird konzeptionell dargestellt, wie der Einfluss von Windgeschwindigkeiten (Wellenhöhen) die Güte der Detektion von Objekten beeinflusst. Dabei wird auch der SAR Aufnahmemodus bzw. die Objektgröße berücksichtigt.

TSX StripMap (7 m)	Vessel size (m)	Confidence Wind speed 4 ms ⁻¹	Confidence Wind speed 10 ms ⁻¹	Coverage 30 * upto 150 km
	7	75%	50%	
	15	95%	75%	
	25	95%	90%	
	40	95%	95%	
	60	95%	95%	
	80	95%	95%	
80	95%	95%		

© 2020 Airbus Defence and Space GmbH

TSX ScanSAR (18.5 m)	Vessel size (m)	Confidence Wind speed 4 ms ⁻¹	Confidence Wind speed 10 ms ⁻¹	Coverage 100 * upto 450 km
	7	wake only for fast moving vessels	wake only for fast moving vessels	
	15	75%	50%	
	25	90%	75%	
	40	95%	95%	
	60	95%	95%	
	80	95%	95%	
80	95%	95%		

© 2020 Airbus Defence and Space GmbH

TSX Wide ScanSAR (40m)	Vessel size (m)	Confidence Wind speed 4 ms ⁻¹	Confidence Wind speed 10 ms ⁻¹	Coverage 270 * upto 600 km
	7	-	-	
	15	wake only for fast moving vessels	-	
	25	50%	wake only for fast moving vessels	
	40	90%	75%	
	60	95%	90%	
	80	95%	95%	
80	95%	95%		

© 2020 Airbus Defence and Space GmbH

Abbildung 3-61: Bewertung der Detektionsgüte von Objekten auf offener See in TerraSAR-X Bilddaten abhängig von Sensorauflösung und Windstärke bzw. Wellenhöhe.

Eine solche Qualitätsangabe kann dann mit Wahrscheinlichkeitswerten aus den Detektionsalgorithmen (Schwellwert-Indikatoren, Datenrauschen) kombiniert werden.

Die Detektionsgüte, $Q_{\text{Detection}}$ könnte sich also zusammensetzen aus den Indikatoren: $Q_{\text{Algorithm}}$, $Q_{\text{External Influence}}$, $Q_{\text{Sensor/Object}}$ und $Q_{\text{Secondary Information}}$.

Dabei kann die Verkettung der Indikatoren auch komplexere Zusammenhänge abbilden. Zum Beispiel ist die Bestimmung der Länge von Objekten in SAR Bildern unterschiedlich genau. Dies hängt davon ab, ob und in welche Richtung sich das Objekt im Verhältnis zur Flugrichtung des Sensors bewegt. Fahren Schiffe genau in Azimuth Richtung werden

Objekte verschmiert und ihre automatische Längenbestimmung ist fehleranfälliger. Auch in Radar Blickrichtung (Range) ergeben sich je nach Fahrtrichtung und Geschwindigkeit Verkürzungen oder Streckungen der Objekte – virtuell im Bild. All die notwendigen Parameter, um die Informationsgüte zu bewerten, liegen für dieses Beispiel vor (Fahrtrichtung und evtl. auch Geschwindigkeit des Objektes, Flug- und Blickrichtung und Einfallswinkel des SAR Sensors).

Es ist daher empfehlenswert, jede Information, die dem Nutzer zur Verfügung gestellt wird, zusätzlich mit einem intelligent (automatisch) erstellten Qualitätsindikator zu versehen. Dies kann in Prozent (Bsp. 0 – 99 %), in Worten (Bsp. low, medium, high) oder in Ampelfarben (Bsp. grün, gelb, rot) etc. geschehen. Wichtig ist, die Logik des Qualitätsparameters in der Produktspezifikation transparent darzulegen.



Abbildung 3-62: Prototypisches Design einer Auflistung aller für ein detektiertes Objekt bereitgestellter Informationen und deren Qualitätsindikatoren. Ein klappbares Panel könnte die Informationen aus einer Sekundärquelle (AIS / RF) zeigen.

Innerhalb einer Visualisierungsplattform ist es vorteilhaft, Lagebilder so darzustellen, dass Einzelobjekte angeklickt werden können und sich dann die relevanten Informationen, die bei

der Dienstbereitstellung erhoben wurden, in einem Kontext Fenster (eine sogenannte Object Card) öffnen. Jeder Information könnte dann ein Qualitätsparameter gegenübergestellt werden. Sekundär-Informationen, wie AIS-Details des jeweiligen Schiffes könnten als zusätzliches paralleles Panel aufklappbar gestaltet werden. Eine Einschätzung der AIS match confidence ist ebenfalls empfehlenswert. Auch diese match confidence sollte intelligent bestimmt werden über räumliche Nähe einerseits und andererseits über einen automatischen Vergleich der im SAR Bild extrahierten Information und den AIS Daten des Objektes. Ergeben sich hier klare Abweichungen zum Beispiel in der Länge, kann dem Nutzer dies über eine spezielle Symbolik (z.B. rot unterlegt) angezeigt werden. Ein prototypisches Design ist in Abbildung 3-62 dargestellt.

Innerhalb einer Analyse-Plattform zur holistischen Visualisierung des maritimen Lagebildes kann ein Nutzer mit all den extrahierten Informationen zu Einzelobjekten auch bestimmte Fragestellungen beantworten oder einzelne Anomalien - vom System aufgezeigt - untersuchen und bewerten. Anomalien könnten über spezielle anpassbare Logiken mithilfe der Vielzahl an erhobenen Einzelinformationen automatisch hervorgehoben werden. Abweichungen von üblichen Mustern oder AIS Signalen, Objekte in sensiblen Gewässern oder Zusammenkünfte von Fahrzeugen sind nur einige der prominentesten Beispiele für Anomalien, die in Lagebild Plattformen dann programmierbar werden. Entscheidend ist die Erhebung der Informationen, die Bewertung der extrahierten Daten und die Überlagerung mit Sekundärinformationen (AIS). Ein einfaches Beispiel für eine solche Anomalie wäre ein Filter: „Zeige alle detektierten Objekte, die mit keinem AIS Signal in Verbindung gebracht werden können“. Bei komplexeren Anomalien müssen entsprechend verschachtelter Abfragen erstellt werden: „Zeige alle Fahrzeuge, die mit hoher Wahrscheinlichkeit Frachtschiffe oder Tanker sind, die größer als 50m sind und innerhalb einer stark befahrenen Schifffahrtsrinne keine oder kaum Fahrt machen“.

Diese fiktiven Beispiele zeigen mögliche programmierbare Anomalie-Warnungen, welche erst möglich werden, wenn alle notwendigen Parameter mit einer entsprechenden Sicherheit aus verschiedenen Quellen zur Verfügung stehen.

3.3.2.3. Optimierung der Ergonomie

Die Ergonomie eines maritimen Informationsdienstes ist besonders relevant bei der Bestellschnittstelle, sowie beim gelieferten Dienst selbst. Zur Ergonomie gehören dabei folgende Aspekte, die für einen optimierten Service betrachtet werden sollten:

a) Bestellung maritimer Satellitendienste

- Klare Übersetzung von Kundenanforderungen in spezifische Produkte
- Unmissverständliche Benennung von Service-/ bzw. Produkt-Kategorien
- Nutzerfreundliche Oberfläche (Karten-Interface) zum Platzieren von Bestellungen
- Innovative Extrapolations-Werkzeuge zur Abdeckung sich bewegender Objekte mit Satelliten-Orbits
- Intelligente Unterstützung des Nutzers bei der Bestellung, um Sensor-Fehlkonfigurationen zu vermeiden

- Transparente Systeme durch Angabe von Order-Deadlines und Lieferzeiten
- Quality Feedback Schleifen sind empfehlenswert, um Nutzererfahrungen zu erfassen und bei Bedarf Anpassungen in der Prozesskette vornehmen zu können.

b) Produktdarstellung

generell:

- Ausgabe eines Produktes, welches die vom Nutzer geforderten Informationen enthält
- Bereitstellung von Qualitätsindikatoren zu den gelieferten Objektinformationen
- Universelle Nutzbarkeit der maritimen Informationen durch Nutzung von generischen Ausgabe-Formaten (z.B. .xml) für die erstellten Produkte

Innerhalb eines Geoinformationssystems (GIS):

- Nutzung einer Grundsymbolik, welche möglichst den gängigen (z.B. IMO) Standards entspricht
- Verwendung einer Symbolik, die Formen, Größen, Farben, Linien-Stile etc. nutzt, um einem Nutzer möglichst intuitiv so viele Informationsebenen wie möglich und dabei *gleichzeitig* anzeigen zu können
- Nutzung aller üblicher GIS Operationen zur Überlagerung mit sekundären Daten-Ebenen
- Gruppierung von ähnlich gearteter Information und bei Bedarf Aus- und Einblendung gruppierter Parameter (Bsp. AIS-Informationen)

Ergonomische Aspekte, die Informations-Extraktion aus dem Satellitenbild betreffend werden in Kapitel **3.3.3.2** und **3.3.3.3** näher betrachtet.

3.3.3. AP 2130 - Optimized NRT Processing

Eine schnelle Bereitstellung und Prozessierung der Daten hat im maritimen Kontext einen besonders hohen Stellenwert. Bei der Schiffserkennung sind daher smarte Konzepte und Lösungen in einer ergonomischen Arbeitsumgebung nötig, die dem Auswerteexperten (Operator) bei der halbautomatischen Bildanalyse zur Verfügung stehen um das Informationsalter des Produktes so weit wie möglich zu reduzieren.

Informationsalter ist hier zu verstehen als die verstrichene Zeit zwischen Aufnahme des Satellitenbildes und der Auslieferung zum Beispiel eines Schiffsdetektionsberichtes an den Endnutzer. Bestimmt wird das Informationsalter im Wesentlichen durch zwei Kategorien: Der Prozess von Aufnahme des Satellitenbildes bis hin zur Verfügbarkeit der Daten beim Operator (Kapitel 3.3.3.1) sowie die eigentliche Bildanalyse und Berichtserstellung inklusive Auslieferung (Kapitel 3.3.3.2 und 3.3.3.3). Die folgenden Kapitel widmen sich der Auslegung und Optimierung einer solchen Nahe-Echt-Zeit (engl. Near-real-time, NRT) Prozesskette am Beispiel der TerraSAR-X-Mission.

3.3.3.1. Eignung weiterer SAR Sensoren zur Schiffsdetektion (ohne Zugang Direkte Empfangsstation)

Im Folgenden werden Untersuchungsergebnisse zur Eignung der SAR Sensoren von PAZ und Sentinel-1 zur Schiffsdetektion beschrieben.

PAZ

Der Radarsatellit PAZ wurde über eine Rahmenvereinbarung von Airbus Defence and Space und dem spanischen Satellitenbetreiber Hisdesat zur gemeinsamen Koordination mit den Satelliten der TerraSAR / TanDEM Mission entwickelt.

Da die drei Radarsatelliten nahezu baugleich sind und die Aufnahmequalität entsprechend angeglichen wurde, ist davon auszugehen, dass es bei der Anwendung von PAZ-Daten zur Schiffsdetektion keine signifikanten Unterschiede zur Nutzung der Systeme TerraSAR / TanDEM gibt. Mit Ausnahme des WideScanSAR-Modes, stehen prinzipiell dieselben Konfigurationsmöglichkeiten zur Verfügung. Die nahe Echtzeit-Fähigkeiten sind seitens Betreiber des PAZ-Satelliten (Hisdesat) jedoch noch etwas eingeschränkt. Die volle Funktionsfähigkeit von nahe Echtzeit-Lieferungen soll bis Ende Juni 2021 hergestellt sein. Für Kunden, die PAZ-Daten über eine eigene Empfangsstation (DRS) beziehen, gibt es keinerlei Einschränkungen.

PAZ Validierung

Die Validierung und Verifizierung der PAZ Datenqualität in Bezug auf die Qualität von TerraSAR-X / TanDEM-X, konnten generell keine signifikanten Unterschiede der Datenqualität festgestellt werden. Daher besteht die berechnete Annahme, dass sich beide Systeme gleichermaßen für die Anwendung zur Schiffsdetektion eignen.

Diese Hypothese in der Praxis tatsächlich zu verifizieren, stellt jedoch eine größere Herausforderung dar, als man zunächst annehmen würde. Im maritimen Umfeld sind die Randbedingungen für eine Vergleichsanalyse denkbar ungünstig. Sowohl die sich ständig verändernden meteorologischen und ozeanographischen Bedingungen wie Windstärke, bzw. Windrichtung, Strömungen und der sich damit dynamisch verändernde Seegang, als auch die Mobilität der Zielobjekte selbst – Schiffe sind ständig in Bewegung, ob in aktiver Fahrt, beim Treiben oder beim Pendeln um einen Anker – machen einen 1-zu-1 Vergleich zu zwei verschiedenen Zeitpunkten auf hoher See unmöglich.

Um dennoch eine Aussage über die Eignung von PAZ Daten zur Schiffsdetektion treffen zu können, wurden zwei Szenen - eine TerraSAR-X und eine PAZ Szene – vor der Ostküste Australiens, nahe dem Kohle Terminal am Hafen von Hay Point ausgewählt. In unmittelbarer Nähe des Terminals liegen rund um die Uhr zahlreiche Frachtschiffe des Typs „Bulk Carrier“ auf Reede. Die Schiffe warten hier an festgelegten Ankerplätzen bis sie zur Beladung an die Anlegestellen gelotst werden.

Dieses Szenario stellt eine, zumindest über eine relativ geringe Zeitspanne hinweg, annähernd gleichbleibende Situation in dem sonst sehr dynamischen, maritimen Umfeld dar. Dies ist eine wichtige Grundvoraussetzung für die Vergleichbarkeit zweier Aufnahmen zu verschiedenen Zeitpunkten, die möglichst eng beieinander liegen sollten.

Da beide Sensoren zur selben Konstellation gehören und die Erde auf demselben Orbit

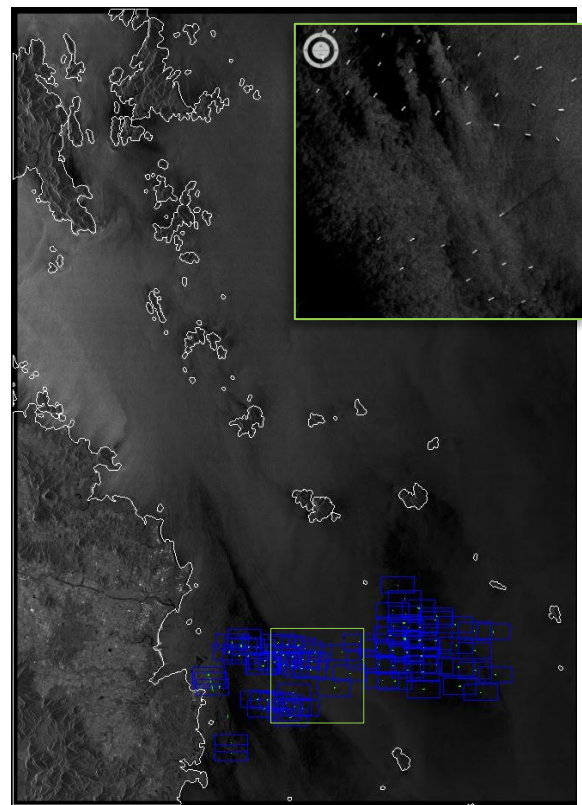
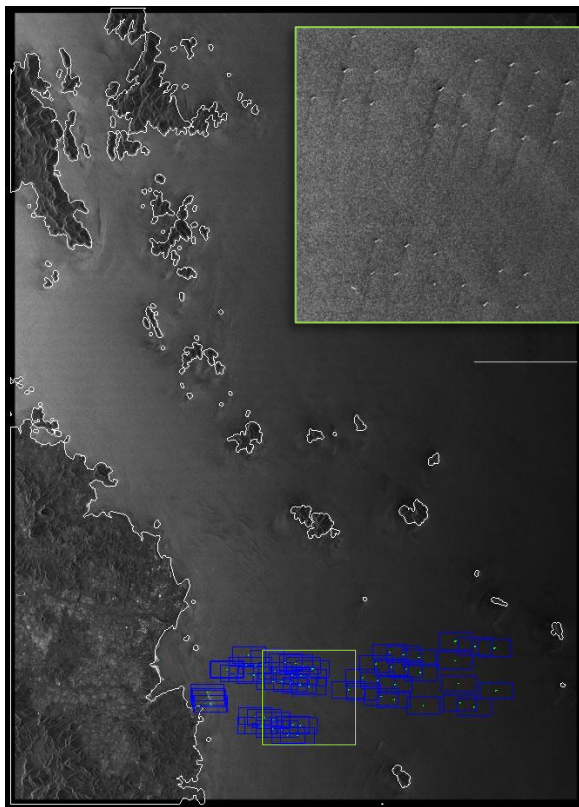
umkreisen, betragen die minimalen Zeitunterschiede zwischen PAZ und Terra-SAR-X Szenen in exakt derselben Aufnahmegeometrie etwa sieben Tage für dieses Gebiet.

Die PAZ Szene wurde am 13.12.2018 um 08:21:41 Uhr [UTC] aufgenommen, die TerraSAR-X Szene sieben Tage später am 20.12.2018 um 08:22:38 Uhr [UTC]. Die Aufnahmeparameter sind für beide identisch: Aufsteigender Orbit, HH polarisiert, 22.67° Blickwinkel im Zentrum der Szene, prozessiert als MGD_RE (Multilook Ground Range Detected; radiometrically enhanced).

Für einen neutralen und möglichst standardisierten Vergleich wurde ein CFAR basierter (Constant False Alarm Rate) state-of-the-art Schiffsdetektions-Algorithmus auf beiden Szenen angewendet. Aufgrund der unterschiedlichen meteorologischen und ozeanografischen Rahmenbedingungen zu den Zeitpunkten der Aufnahmen, musste der Schwellwert, der die Sensitivität des Detektions-Algorithmus steuert, für jede Szene individuell angepasst werden. Abbildung 3-63 zeigt einen Überblick beider Szenen sowie die Algorithmus-Ergebnisse (blaue Boxen).

TerraSAR-X

PAZ



Tiles: 90
Runtime: 48s
Threshold: 4
Results: good

Tiles: 91
Runtime: 47s

Tot. detections:	61	Threshold:	8
True positives:	54 (88%)	Results:	good
False positives:	7 (12%)	Tot. detections:	86
False negatives:	1	True positives:	69 (80%)
Sea state:	rough	False positives:	17 (20%)
		False negatives:	6
		Sea state:	smooth

Abbildung 3-63: Vergleich automatischer Schiffsdetektion (CFAR) zwischen PAZ und TerraSAR-X

Die Statistik zeigt, dass in der Prozessierung der beiden Szenen keine signifikanten Unterschiede auszumachen sind. Die Laufzeit ist beinahe identisch und die Ergebnisse sind, nach qualitativer Analyse durch einen erfahrenen SAR Bildanalysten, ebenfalls gleichermaßen solide. Die höhere Falschalarm-Rate (False positives & false negatives) in der PAZ-Szene kann mit den Randbedingungen der Szene erklärt werden. Aufgrund des geringen Seegangs ist die Rückstreuung innerhalb des Bildes heterogener als im TerraSAR-X Bild, in dem konstant höhere Windgeschwindigkeiten für eine gleichmäßig raue Wasseroberfläche über die gesamte Szene sorgen (siehe vergrößerter Ausschnitt in Abbildung 3-63). Selbst schwache Signale können so in der PAZ-Szene leichter als vermeintliche Objekte erkannt werden, wenn die relative Rückstreuung im Vergleich zur unmittelbaren Umgebung groß genug ist. Dieser Effekt kann v. a. in den dunklen Bildbereichen, wo die Wellenbewegung sehr schwach ist und somit wenig Rückstreuung in Richtung des Sensors reflektiert wird, auftreten.

Fazit

Alles in Allem eignen sich jedoch beide Ergebnisse gut als Grundlage für eine operative, semi-automatische Schiffsdetektionsanalyse. Die Eingangsfrage, ob PAZ-Daten ebenso wie TerraSAR-X-Daten zur Schiffsdetektion verwendet werden können, lässt sich somit erwartungsgemäß mit „Ja“ beantworten.

Sentinel-1

Sentinel-1-Daten eignen sich grundsätzlich ebenso gut zur Schiffsdetektion, wie andere SAR Sensoren auch. Allerdings gibt es einige Unterschiede, Einschränkungen und Vorteile, gegenüber anderen SAR Daten, die im Folgenden erläutert werden.

Einer der größten Unterschiede liegt wohl in der freien und kostenlosen Verfügbarkeit von Sentinel-1-Szenen. Die Daten der zum europäischen Erdbeobachtungsprogramm Copernicus gehörenden Sentinel-Missionen sind grundsätzlich für die Allgemeinheit kostenfrei zugänglich, d. h. es entstehen keine Beschaffungskosten.

Im Gegensatz zu vielen kommerziellen Raumfahrtprogrammen, bietet die Plattform jedoch keine Flexibilität in der Missionsplanung. Der Sensor nimmt kontinuierlich Daten in

vordefinierter Aufnahmegeometrie auf. Jede Szene ist im gleichen Modus mit gleichbleibender Auflösung, derselben Polarisierung und unveränderbarem Blickwinkelbereich aufgenommen.

Missionen wie die Airbus Radarkonstellation bieten durch ihre Flexibilität in der Aufnahmeconfiguration oft eine häufigere Wiederkehrtrate, bzw. Abdeckung eines bestimmten Ortes auf der Erde – wenn auch nicht mit identischer Aufnahmegeometrie, die z.B. für interferometrische Analysen notwendig wäre. Für maritime Anwendungen spielt eine konstante Aufnahmegeometrie eine untergeordnete Rolle im Vergleich zur maximal möglichen Abdeckung eines Ortes. Dies ermöglicht es im besten Fall mehrere Aufnahmen pro Tag vom selben Gebiet zu machen und gleichzeitig frei zwischen verschiedenen Varianten der Abdeckung, Auflösung, Blickrichtung und –winkelbereich sowie der Polarisierung wählen zu können (Vergl. Tabelle 3-31).

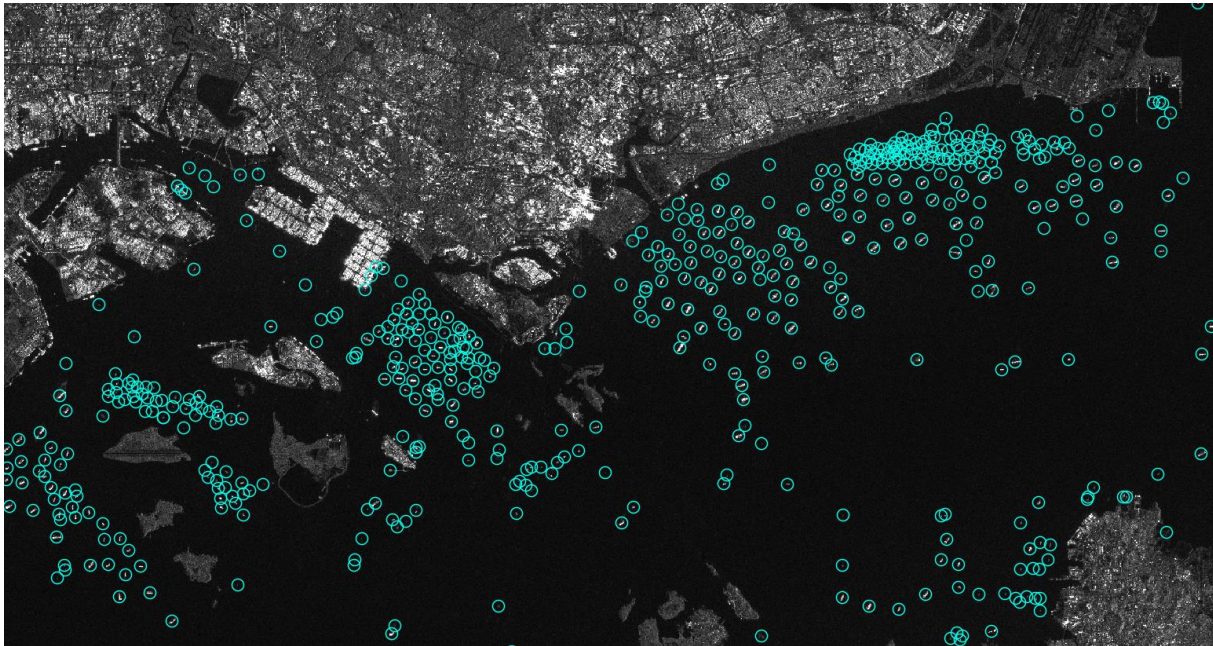
Neben diesen weichen Faktoren, ist für die Eignung eines SAR Sensors zur Schiffsdetektion v. a. die Qualität der Daten entscheidend. Die Sensorauflösung beeinflusst die Detektierbarkeit in Abhängigkeit der Objektgröße maßgebend (vgl. Kap. 3.3.1.3). Des Weiteren von Bedeutung ist die radiometrische Auflösung sowie weitere SAR spezifische Bildeigenschaften wie *speckle* und *noise*, die das Rauschverhalten im Bild bestimmen. Je verrauschter das Bild und schwächer die Auflösung, desto schwieriger ist die Detektion, sowohl für das menschliche Auge, als auch für Algorithmen und Prozessoren zur automatisierten Schiffsdetektion.

Sentinel-1 - Automatische Schiffsdetektion

Abbildung 3-64 zeigt das Ergebnis eines CFAR basierten (Constant False Alarm Rate) state-of-the-art Schiffsdetektions-Algorithmus auf einem Ausschnitt einer Sentinel-1 Szene vom 08.01.2020 über Singapur (S1A_IW_GRDH_1SDV_20200108T224756_20200108T224821_030715_03856E_1FB2.SAFE).

Die Verifizierung der Ergebnisse durch einen erfahrenen SAR Bildanalysten zeigt, dass der Algorithmus eine außergewöhnlich gute Performanz auf der Sentinel-1 Szene erzielen konnte. Die Anzahl der fälschlicherweise detektierten (false positives), bzw. der fälschlicherweise nicht detektierten (false negatives) Objekte liegt nach qualitativer Analyse jeweils unter 1 %.

Eine derartige Genauigkeit und Präzision ist mit demselben Algorithmus auf vergleichbaren Daten anderer Sensoren eher selten. Dies mag auch der Tatsache geschuldet sein, dass der Algorithmus ursprünglich mit Hauptaugenmerk auf die Verwendung von Sentinel-1-Daten konzipiert wurde, wenngleich er auch auf eine Vielzahl anderer SAR-Sensoren angewendet werden kann. Das JRC (Joint Research Center) hatte den SUMO-Algorithmus, inkl. Source-Code als „as-is release“ Anfang Jahr 2017 als Open-Source-Software freigegeben.



**Abbildung 3-64: Sentinel-1 CFAR Ergebnisse des SUMO Algorithmus über Singapur
(S1A_IW_GRDH_1SDV_20200108T224756_20200108T224821_030715_03856E_1FB2.SAFE)**

Für die automatisierte Schiffserkennung sind Sentinel-1 Szenen also ebenso geeignet, wie andere SAR-Sensoren.

Sentinel-1 - Manuelle Schiffsdetektion

Ein weiteres Beispiel zeigt, dass Sentinel-1-Daten auch für rein manuelle und visuelle Analysen herangezogen werden können, solange die vorgegebenen Eigenschaften des Sensors, wie z. B. die geometrische Auflösung von rund 20 m, für die Fragestellung geeignet sind. Abbildung 3-65 zeigt ein Schiffsunglück im Oktober 2018 im Mittelmeer, nördlich der Küste Korsikas. Die Frachtfähre „Ulysse“ war mit dem Frachtschiff „CSL Virginia“ kollidiert und hatte einen Kilometerlangen Ölteppich verursacht.

Für Monitoring-Zwecke des sich auf der Meeresoberfläche ausbreitenden Treibstoffs, sollte eine möglichst engmaschige zeitliche Abdeckung des Gebietes erfolgen. Als Ergänzung zu den verwendeten TerraSAR-X Aufnahmen, konnte eine passende Sentinel-1 Szene gefunden werden, die sich hervorragend zur Schließung von Datenlücken eignete.

Trotz der etwas schlechteren Auflösung, im Vergleich zu den verwendeten TerraSAR-X Szenen (StripMap mode in RE Prozessierung ~ 7 m), ist der Ölteppich in Abbildung 3-65 ebenso gut zu erkennen, wie die kollidierten Schiffe - zumindest bis zu einem gewissen Detailgrad.

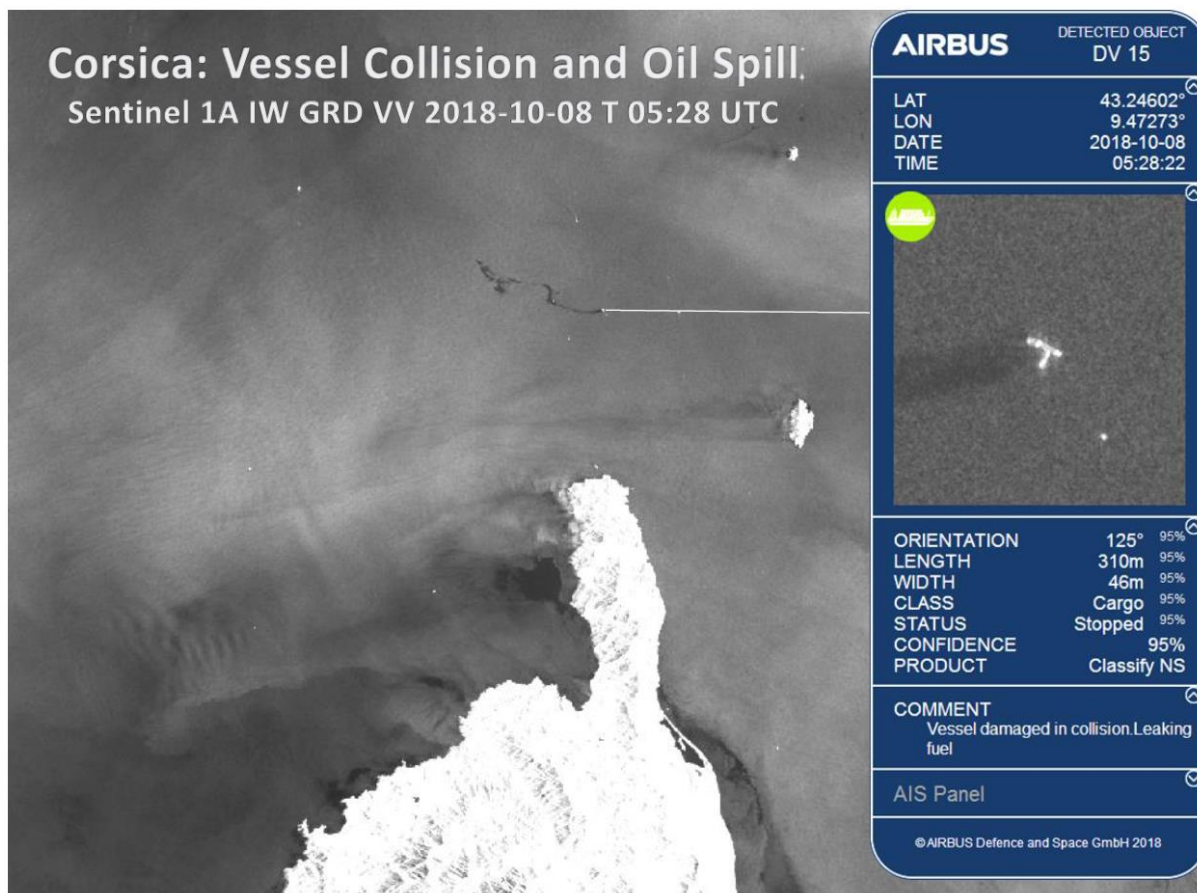


Abbildung 3-65: Sentinel-1 Beispiel für manuelle Schiffserkennung

Solche Synergien, basierend auf verschiedenen satellitengestützten Sensoren, eignen sich hervorragend für die Überwachung eines bestimmten Gebietes oder einer Situation über einen definierten Zeitraum und erfreuen sich daher immer größerer Beliebtheit in der Anwendung.

Sentinel-1-Daten eignen sich sehr gut für eine solche Synergie, solange die unveränderliche Konfiguration der Aufnahmeparameter im Einklang mit der Fragestellung steht. Die Größe der Zielobjekte z. B., sollte demnach mit der Bildauflösung des Sensors vereinbar sein (siehe dazu Abbildung 3-61 für TerraSAR-X). Mit ~ 20 m Pixelgröße ist es z. B. beinahe unmöglich 15 m große Boote zuverlässig zu detektieren.

3.3.3.2. Funktionale Anforderungen an eine Auswertesoftware zur Schiffsdetektion

Systeme zur Schiffserkennung im maritimen Umfeld zu entwickeln ist, aus verschiedenen Gründen eine große Herausforderung.

Die Weltmeere bedecken rund 2/3 der Erdoberfläche und stellen somit ein schier unendlich großes Gebiet dar. Maritime Aktivitäten umspannen daher den gesamten Globus und können u. U. für Akteure von Bedeutung sein, die sich an ganz anderen Orten der Welt befinden, als am Schauplatz der eigentlichen Ereignisse. Verspätet sich ein Frachtschiff, verliert Ladung oder wird entführt, betrifft das zumeist Reedereien, Unternehmen und Institutionen in tausenden Kilometern Entfernung. Der immense Schaden, der durch illegalen

Fischfang und Drogenschmuggel angerichtet wird, beschränkt sich i. d. R. nicht einmal auf einzelne Akteure, sondern kann diffuse Auswirkungen auf globaler Ebene mit sich bringen.

Für eine zuverlässige Informationsgewinnung und eine weltweite Abdeckung sind daher Daten aus unterschiedlichen Quellen und in verschiedenen Dateitypen und -formaten notwendig, die ein System zur Schiffserkennung verarbeiten können muss.

Daten

Das von der Internationalen Seeschiffahrts-Organisation (IMO) anerkannte Automatic Identification System (AIS) ist ein Funksystem, das durch den Austausch von Navigations- und weiteren Schiffsdaten die Sicherheit verbessert und die Steuerung des globalen Schiffsverkehrs erleichtert. Diese teils terrestrisch, teils per Satellit gewonnen AIS-Daten, gelten als Standard zur Überwachung des globalen Schiffsverkehrs. Ein Tool zur Schiffserkennung muss also zwingend in der Lage sein, AIS-Daten nutzen zu können.

Dennoch sendet nicht jedes rechtlich dazu verpflichtete Schiff AIS-Signale - ob beabsichtigt, z. B. um bei illegalen Aktivitäten unentdeckt zu bleiben, oder unbeabsichtigt, z. B. aufgrund technischer Probleme oder Beschädigung. Daher sind darüber hinaus weitere Informationsquellen für eine zuverlässige Überwachung des Schiffsverkehrs erforderlich.

Satellitendaten werden dabei wegen ihrer globalen Abdeckung und der Zuverlässigkeit der Daten sehr häufig als weitere Informationsquelle herangezogen. Besonders SAR-Sensoren eignen sich aufgrund der Tatsache, dass sie Tageslicht- und Wetterunabhängig agieren können, besonders gut dafür.

Allerdings ist der Schiffsverkehr – ebenso wie die Meere selbst – ein höchst dynamisches und komplexes Unterfangen mit sich ständig verändernden Situationen. Brauchbare Informationen zur aktuellen Lage an einem bestimmten Ort verlieren schon nach kürzester Zeit ihren Wert, insbesondere wenn die Daten nur eine Momentaufnahme der Situation widerspiegeln, wie es bei Satellitendaten der Fall ist.

Weitere Fernerkundungsdaten, wie z. B. Luftbilder, können ebenso zu einer verbesserten Informationslage beitragen wie das Einbeziehen von Wetterdaten oder Strömungssimulationen.

In jedem Fall ist die Zeit ein kostbares Gut bei der Erstellung maritimer Lagebilder und ist damit ein bestimmender Faktor in der Auslegung einer Auswertesoftware.

Elemente eines Toolkits zur Schiffsdetektion

Folglich muss ein operator-gestütztes Toolkit zur Schiffserkennung in der Lage sein, Daten möglichst schnell und effizient zu verarbeiten und den Operatoren mit Hilfe ergonomischer Tools und performanten Systemen, eine schnelle und zuverlässige Analyse und Auswertung von aktuellen Situationen im maritimen Umfeld rund um den Globus zu ermöglichen.

Die wichtigsten technischen Anforderungen an ein solches System sind:

Datenhandhabung

- Effizientes und korrektes Einlesen und Darstellen der SAR-Bilddaten (z.B. TIFF) und der nötigen Metadaten (xml) in einer Kartenansicht und einer separaten Datenansicht

- in nativer Geometrie (z.B. MGD & GEC bei TerraSAR-X)
- in originaler Auflösung und Bittiefe (z. B. 7 m, 16 Bit für TSX StripMap)
- mit möglichst genauem geographischen Bezug in korrekter Projektion (z. B. für die Darstellung als Layer über einer geeigneten Basemap; vgl. GIS-Software)
- inklusive flüssiger Navigation (pan & zoom)
- Effizientes und korrektes Einlesen und Darstellen weiterer Daten und Datenformate (z.B. GeoJSON, shp, xml, csv) in einer Kartenansicht und einer separaten Datenansicht, z.B.:
 - AIS-Daten
 - Detektionsergebnisse von externen automatischen Schiffserkennungs-Algorithmen
 - Layer für Landmasken

Bildbearbeitung

- Tools zur grundlegenden Bildbearbeitung (z.B. Histogrammanpassung) für individuelle und lokale visuelle Verbesserung der Szene, die eine optimale Auswertung der Bildinformation ermöglicht
 - Hilfreich ist ein „Dual View“ Konzept (duplizierte und synchronisierte Kartenansicht in einem zweiten Fenster), um unterschiedliche Bildeinstellungen gleichzeitig nutzen zu können. So kann das Histogramm im einen Fenster auf die Wasseroberfläche angepasst sein (z. B. um Bugwellen besser zu erkennen) und die andere auf Details der Schiffsstrukturen (z. B. um besser klassifizieren zu können)

Manuelle Detektion

- Tools zur manuellen Detektion und Bearbeitung von Objekten, insbesondere von Schiffen, und deren Vermessung (Position, Länge, Breite und Orientierung)
 - Die Objekte sollten auf der Kartenansicht, z. B. als Vektor-Layer (shp), so visualisiert sein, dass deren Ausmaße und Orientierung erkennbar sind, jedoch nicht den Bildinhalt verdecken (z.B. Polygone ohne Füllung).
 - Die Symbolik soll dabei intuitiv verständlich sein und auch möglichst viele Informationen über die Eigenschaften des Objektes widerspiegeln (siehe Datenmodell unten).
 - Die Tools sollten intuitiv, verständlich und ergonomisch auf schnelle Bearbeitung ausgelegt sein.
- Tool zur Auswahl und Bearbeitung von einzelnen und mehreren Objekten (vgl. select by Box, by Polygon,... in herkömmlichen GIS-Systemen)

Automatische Detektion

- Tools zur automatisierten Detektion über Algorithmen, die direkt aus der Software gestartet werden können (per Knopfdruck) inkl. etwaiger Konfigurationen wie Schwellwertanpassungen, etc.
 - o Algorithmus-Ergebnisse sollten validiert oder verworfen werden können.
 - o Validierte Detektionen sollen äquivalent zu manuell erstellten Objekten behandelt werden und ebenso bearbeitet und gelöscht werden können.

Datenmodell und Attributtabelle

- Die Objekte und deren Parameter sollen anhand eines geeigneten Datenmodells in einer separaten Attributtabelle geführt werden
 - o Das Datenmodell sollte mindestens folgende Parameter beinhalten:
 - Position (lat/lon)
 - Länge, Breite, Ausrichtung bzw. Orientierung, wenn nicht klar ist wo Bug und Heck sind
 - Klassifizierung (bearbeitbare Dropdown-Liste, z.B. Tanker, Cargo, Fishing,...)
 - Status (stopped, moving, moored, at anchor,...; Dropdown-Liste)
 - Maximale Intensität (Pixelwert)
 - MMSI (Maritime Mobile Service Identity) falls AIS-Signal vorhanden
 - Gesamt-Konfidenz in % (Gütemaß - wie sicher handelt es sich bei dem Objekt um ein Schiff; Dropdown-Liste)
 - Parameter-Konfidenz in % (wie oben, nur für einzelne Parameter)
 - Kommentarfeld für individuelle Anmerkungen
 - Geschwindigkeit
 - o Alle Parameter sollen über eine Editierfunktion in der Attributtabelle bearbeitet werden können. Alle Änderungen sollen auch in der Objektkarte und der Visualisierung (Symbolik auf der Karte) in Echtzeit aktualisiert werden

Objektkarte

- Eine Objektkarte verbindet die Kartenansicht mit der Attributtabelle
 - o Alle Parameter, inkl. Image Chips (=vergrößerte Darstellung)) des ausgewählten Objektes werden in einem Viewer angezeigt und können dort bearbeitet werden.
 - o Bei Mehrfachauswahl soll in der Objektkarte durch die Objekte „gebrowset“ werden können.
- Das ein- und mehrfache Auswählen und Bearbeiten von Objekten soll – ebenso wie in der Attributtabelle - auch über die Kartenansicht möglich sein.

AIS

- Eingeladene AIS-Daten sollen über intelligente Matching-Mechanismen mit bestehenden Objekten (Algorithmusergebnisse, oder manuell erstellte Objekte) automatisch korreliert und verlinkt werden.
 - o Außerdem sollen AIS-Daten auch manuell mit Objekten verlinkt bzw. entkoppelt werden können.
 - o Objekte, die eine AIS-Verlinkung besitzen, sollen entsprechend gekennzeichnet sein (z. B. Fähnchen),

SAR Geschwindigkeit

- Tool zur Geschwindigkeitsberechnung aus dem SAR-spezifischen Phänomen des Dopplereffekts
 - o Sich bewegende Objekte liegen in einer SAR-Szene versetzt zur eigentlichen Position des Objektes im Bild. Aus dieser Distanz, der Fahrtrichtung und eignen Aufnahme-spezifischen Metadaten lässt sich die Geschwindigkeit in einer gewissen Näherung berechnen.

Ambiguitäten

- Tool zur Erkennung von Ambiguitäten und Artefakten („Ghosts“)
 - o Z. B. als verschiebbares Fadenkreuz, ausgerichtet in Azimut, bzw. in Range Richtung + zwei Kreise mit Mittelpunkt im Zentrum des Fadenkreuzes und einstellbarem Radius, um Bereiche mit potentiellen Ambiguitäten der individuellen Szenen anzupassen.

Qualitätskontrolle

- Intelligente Tools zur Qualitätssicherung
 - o Filter-/Sortier- und Auswahlverfahren, die mehrere Parameter gleichzeitig berücksichtigen, um bestimmte Gruppen detailliert untersuchen zu können (z. B. alle fahrenden Tanker größer als 180 m, oder alle Schiffe kleiner als 50 m ohne AIS-Link)
 - o Finale Validierung kurz bevor der Bericht erstellt wird, z. B. durch eine Gesamtübersicht aller Image Chips. Damit können bspw. Fehldetektionen von kleinen Objekten erkannt werden, die in der Bearbeitung zuvor nicht aufgefallen sind.
 - o AIS-Matching-Konfidenz: Abgleich der manuell, bzw. automatisch bestimmten Parametern einer Detektion (z.B. Länge, Breite, Klasse, etc.) mit den entsprechenden Informationen des korrelierten AIS-Signals. Dies könnte bspw. über eine Prozentangabe oder eine Ampelfarbgebung erfolgen (Bsp.: AIS: Tanker, 100 m, gemessen: Cargo, 95 m = 95 % Konfidenz: grün, Klasse falsch: rot)

Berichterstellung

- Schnelle und unkomplizierte Erstellung (z. B. per Knopfdruck) eines geeigneten Output-Produktes (Schiffserkennungs-Bericht) mit allen nötigen Informationen

- Übersicht über alle Parameter jedes detektierten Objektes, inkl. Quicklook
- Ggfs. verschiedenen, generischen Produktformate (z. B. pdf, csv, xml)

Generell sollten die Kosten für die Entwicklung eines solchen Tools im Verhältnis zum Nutzen stehen.

3.3.3.3. Prototyp einer operator-gestützten Arbeitsumgebung für die Erstellung von SAR-basierten Schiffsdetektionsprodukten

Zu einem ganzheitlichen System zur Schiffserkennung gehören neben den eigentlichen Auswertetools auch organisatorisch-administrative Funktionen wie z. B. Anwendungen rund um das Projektmanagement. Kundeninformationen, Projektbeschreibungen und weitere Informationen rund um ein Projekt müssen ordentlich und strukturiert gehandhabt und archiviert werden. Da dies jedoch auch mithilfe anderer Programme außerhalb der eigentlichen Schiffserkennungssoftware, z. B. mit MS Office Anwendungen oder mit simplen Projektordnern, organisiert werden kann, wird an dieser Stelle nicht weiter auf das Projektmanagement eingegangen. Außerdem werden Details zur Systemarchitektur und zur Schnittstellenthematik ebenfalls nicht behandelt, da hier individuelle Anforderungen an das Toolkit von Bedeutung sind. In diesem Zusammenhang dreht es sich um Fragestellungen wie z. B. „Mit welchen konkreten Sensoren und Daten soll das Toolkit arbeiten können?“, „Welcher AIS-Provider wird verwendet?“ oder „Welche Anforderungen werden an das Berichtsformat gestellt?“ Der folgende Abschnitt fokussiert sich daher auf die nötigen Werkzeuge zur Schiffserkennung an sich.

Layout

Abbildung 3-66 zeigt eine prototypische Auslegung eines operator-gestützten Toolkits zur Schiffsdetektion, untergliedert in zwei Teile, die im besten Fall parallel auf zwei Bildschirmen oder einem Bildschirm im Breitbildformat dargestellt werden können. So behält der Operator stets den Überblick über alle wichtigen Informationen, die zu einer effizienten Analyse nötig sind.

Im „MAP VIEWER“ (vgl. Abbildung 3-67) auf dem linken Bildschirm werden Bildanpassungen, Layer-Management sowie die eigentliche Schiffsdetektion anhand verschiedener, speziell dafür entwickelter Tools, in geographischem Kartenbezug durchgeführt. Im „DATA VIEWER“ auf dem rechten Bildschirm (vgl. Abbildung 3-68) erhält der Operator Einblick in die Parameter der detektierten Schiffe, wie z. B. Länge, Breite, Richtung, Klasse, etc. sowie einen Eindruck der optischen Erscheinung der Objekte in der Szene.

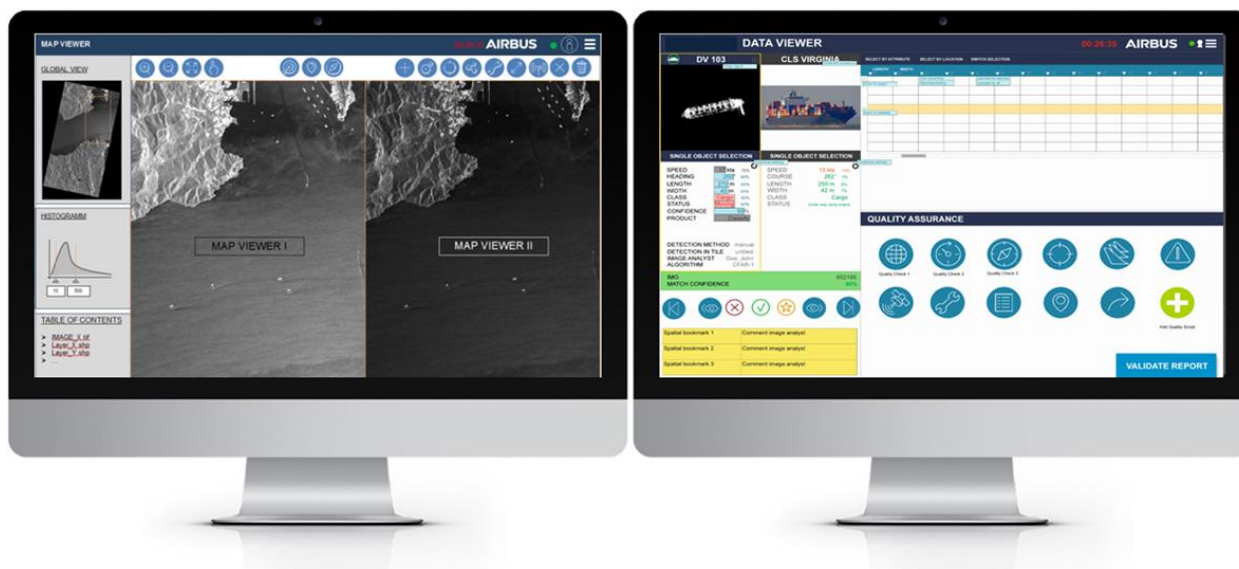


Abbildung 3-66: Toolkit Layout auf zwei Bildschirmen (Detailansicht in Abbildung 3-67 und Abbildung 3-68)

MAP VIEWER

Im Mittelpunkt des „MAP VIEWER“ Bildschirms stehen zwei Fenster (MAP VIEWER I + II), die das Herzstück der Analyse darstellen (s. Abbildung 3-67). Hier bekommt der Operator einen visuellen Einblick in die Bilddaten, sowie einen geographischen Bezug zur vorliegenden Situation vor Ort und zum Zeitpunkt der Aufnahme.

Die Zweiteilung ermöglicht es, unterschiedliche Anpassungen für dieselbe Szene vorzunehmen um gleichzeitig unterschiedliche Bildinformationen nutzen zu können. Z. B. kann das Histogramm des Bildes im linken Viewer I auf die Struktur der Wasseroberfläche justiert werden, um sich bewegende Objekte anhand der Heckwelle ausfindig zu machen. Der rechte Viewer II kann hingegen so eingestellt werden, dass die Strukturen an Deck eines Schiffes detaillierter dargestellt werden um dessen Klasse, z.B. anhand der Position der Brücke, vorhandener Kräne, Container oder Rohrsysteme, bestimmen zu können.

HISTOGRAMM

Das Bildhistogramm kann mit Hilfe von Schiebereglern oder durch direktes Eintippen der Min- und Max-Werte im „HISTOGRAMM“ Fenster auf der linken Seite eingestellt werden.

GLOBAL VIEW

Im „GLOBAL VIEW“ wird der Ausschnitt, der in den synchronisierten MAP VIEWERN I + II dargestellten Ansicht, auf der Gesamtszene dargestellt. So behält der Operator den Überblick bei der Navigation durch die Szene, auch wenn er sich in einer vergrößerten Ansicht befindet.

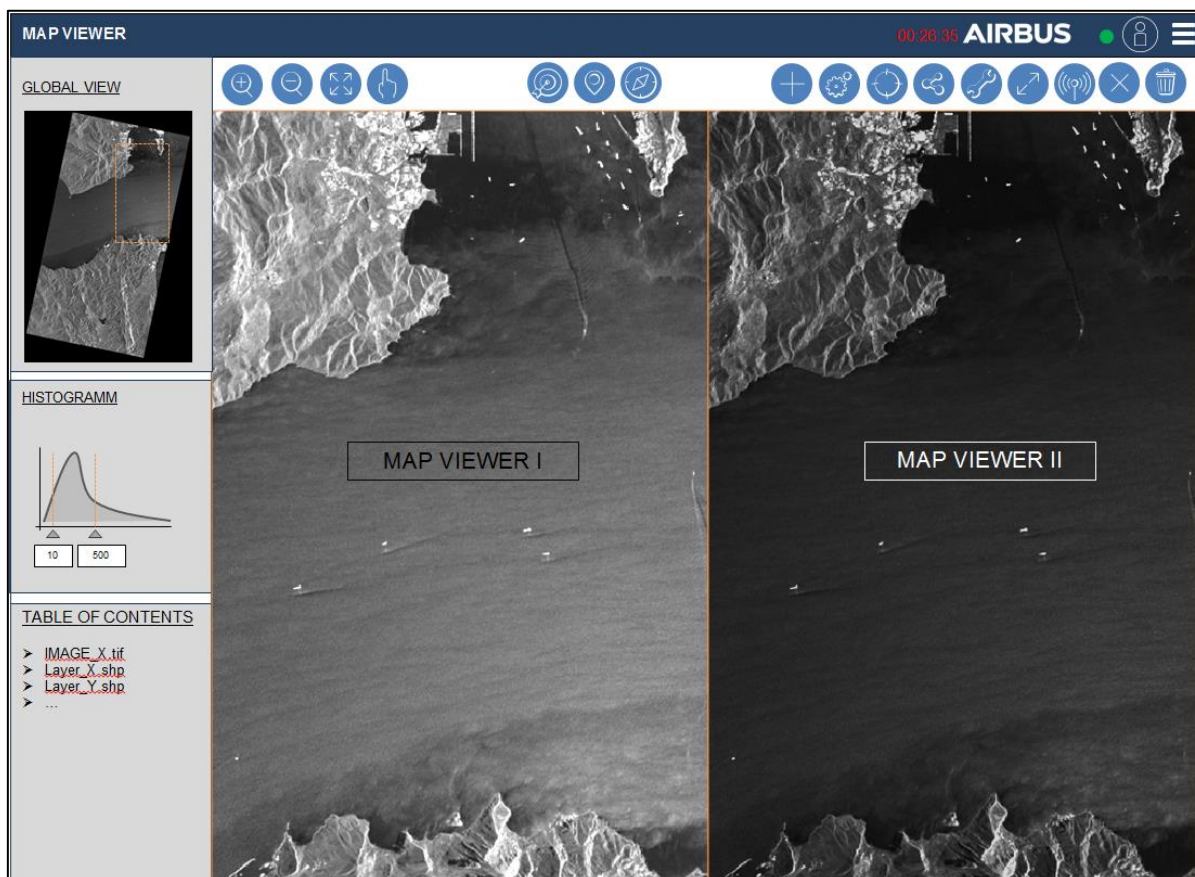


Abbildung 3-67: MAP VIEWER Entwurf (vgl. Abbildung 3-66 linker Bildschirm)

TABLE OF CONTENTS

Im „TABLE OF CONTENTS“ sind alle Layer gelistet, die über eine Checkbox aktiviert oder deaktiviert werden können und entsprechend in den MAP VIEWERN I + II dargestellt werden können. Die Anordnung der Ebenen soll über einfaches Drag & Drop bzw. Verschieben der Layer erfolgen.

Dadurch können z.B. Algorithusergebnisse und AIS-Daten übereinander auf den zugrunde liegenden Bilddaten dargestellt werden. Manuelle Detektionen können als weiterer Layer dazu geschaltet werden. Besonders wichtig ist hierbei eine geeignete Symbolik zur Darstellung der jeweiligen Vektorlayer (vgl. Kap. 3.3.2.3.).

TOOLBAR

In der Toolbar unmittelbar über den MAP VIEWERN I + II sind sowohl herkömmliche, GIS-typische Tools wie „Panning“, „Zooming“, etc. zur Navigation auf der Karte zu finden, als auch Tools zur Schiffsdetektion selbst (vgl. Kap. 3.3.2.3).

Letztere dienen in erster Linie der manuellen Erstellung und Bearbeitung von Schiffen. Dabei soll deren Geometrie über Mausklicks bestimmt werden. Mit drei aufeinander folgenden Klicks können auf diese Weise Position (lat/lon), Länge, Breite, Richtung ebenso wie Umfang und Fläche berechnet werden, indem der erste Klick das Heck, der zweite Klick den Bug und der dritte Klick die Seite markiert. Die errechneten Daten sollen unmittelbar nach Bestätigung der gezeichneten Geometrie (z.B. durch Drücken der Eingabetaste) im Datenmodell

hinterlegt und in der Attributtabelle und der Objektkarte eingetragen werden. In den MAP VIEWERN erscheint ein Pfeilsymbol, dessen Umrisse die aufgenommene Geometrie widerspiegelt.

Falls die Signatur im SAR Bild eine eindeutige Zuweisung von Bug und Heck mangels Detailtiefe nicht zulässt, soll auch ein rechteckiges Symbol ausgewählt werden können. Die Richtung soll dann entsprechend im Datenmodell nur als „Orientierung“ vermerkt werden, z.B. als „90° or 270°“ anstelle von „90°“.

Bei besonders kleinen Schiffen im Verhältnis zur Auflösung kann es vorkommen, dass auch die Orientierung nicht eindeutig festzustellen ist. Diese Signaturen erscheinen mehr als eine Anhäufung sehr weniger heller Pixel im SAR-Bild. Hierfür ist anstelle des 3-Klick-Tools ein 1-Klick-Tool sinnvoll, mit dem nur die zentrale Position als Kreis und dessen ungefähre Ausmaße über den Radius bestimmt werden können (z. B. als vorher definierbaren Radius).

Automatisch erzeugte Detektionen, die aus externen Algorithmen erstellt wurden, sollen über eine entsprechende Schnittstelle eingelesen und deren Geometrie ebenso über Polygone (Rechteck, Dreieck, etc.) mit einem Label (z.B. „algo“) dargestellt werden. Darüber hinaus sollen Schiffsdetektionsalgorithmen auch direkt über einen Button in der Toolbar gestartet werden können.

Das Editieren von bestehenden Detektionen – ob manuell, oder automatisch – soll mit der Maus durch Verschieben der Knotenpunkte in den Polygonen möglich sein.

DATA VIEWER

Der linke Teil des DATA VIEWER widmet sich der Übersicht und Navigation der erfassten Objekte. Oben in der Objektkarte, die mit einer Karte aus einem Quartett-Spiel verglichen werden kann, wird ein Quicklook des ausgewählten Schiffs dargestellt um dessen Form und Struktur analysieren zu können. Darunter sind die Parameter gelistet, die durch automatische oder manuelle Detektion bestimmt wurden. Diese sollen dort auch direkt editiert werden können. Unmittelbar rechts daneben werden im Falle einer Verlinkung mit einem AIS-Signal die zugehörigen AIS-Informationen parallel dazu gelistet. Auf diese Weise kann ein direkter Vergleich der Parameter erfolgen und somit die Konfidenz eines Matches besser bewertet werden.

OBJEKTKARTE

Über die Schaltfläche unterhalb der Objekt- und AIS-Karte kann im Falle einer Mehrfachauswahl durch die einzelnen Objekte navigiert, bzw. „gebrowst“ werden. Anhand dieses Objekt-Browsers können beispielsweise automatische Detektionen validiert oder entfernt werden oder ein Bookmark für eine spätere Bearbeitung gesetzt werden.

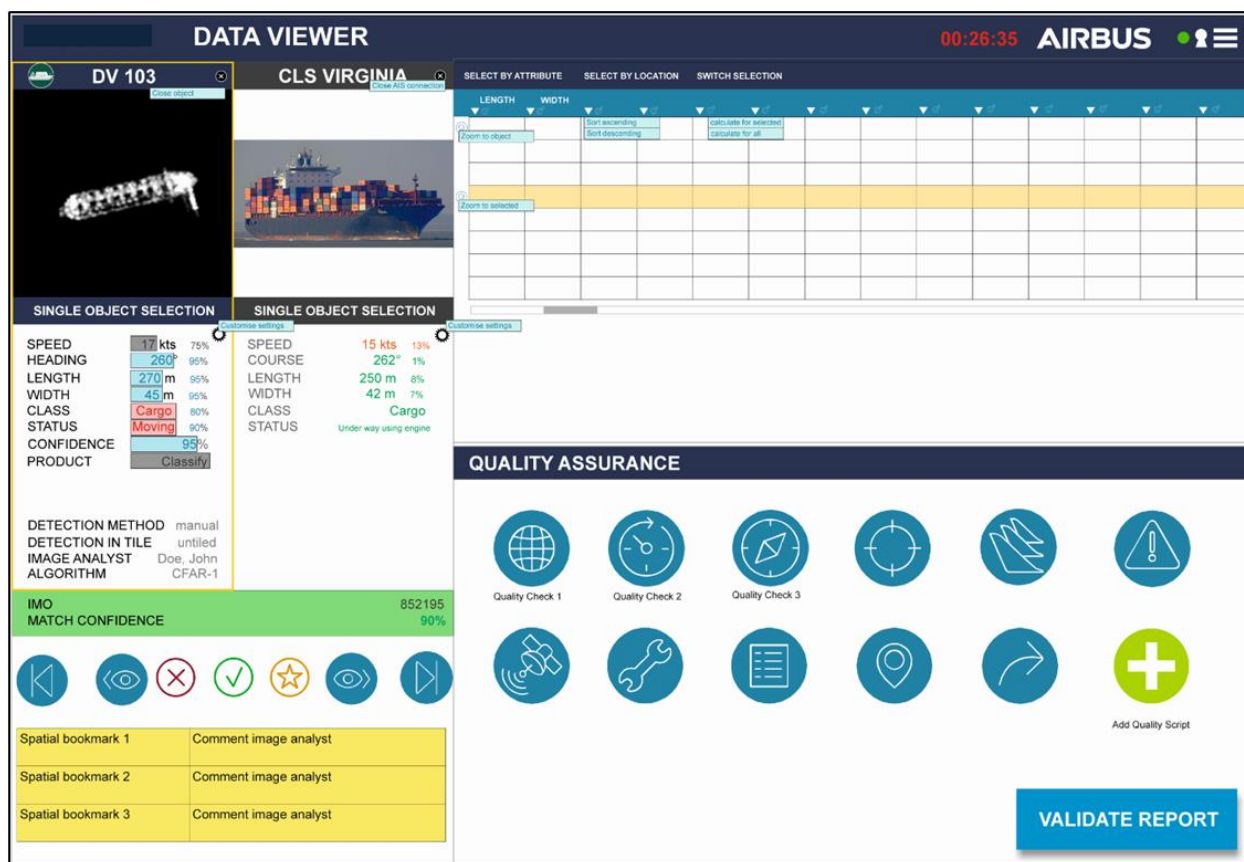


Abbildung 3-68: DATA VIEWER Entwurf (vgl. Abbildung 3-66 rechter Bildschirm)

ATTRIBUTTABELLE

In der Attributtabelle sind alle detektierten Objekte mitsamt aller zugehörigen Parameter gelistet. Über intelligente Auswahl- und Sortierfunktionen wird das Auswählen, Editieren oder Entfernen einzelner oder mehrerer Objekte bzw. Werte effizient gestaltet. Eine Auswahl, die in der Attributtabelle vorgenommen wird, soll parallel auch in den Layern auf den MAP VIEWERN I + II hervorgehoben werden und umgekehrt.

HEADER

Die blaue Schaltfläche am oberen Bildschirmrand beinhaltet neben den Informationen zum eingeloggtten Operator eine Menüfunktion (Dropdown-Liste) für verschiedene Einstellungen und Konfigurationsmöglichkeiten (z. B. die Anordnung der Fenster) und eine Countdown-Anzeige, die die verbleibende Zeit anzeigt, bis der Bericht geliefert werden muss.

QUALITÄTSKONTROLLE & BERICHTERSTELLUNG

Diverse Tools zur Qualitätskontrolle und eine Exportfunktion für den finalen Schiffserkennungs-Bericht runden das Toolkit ab. Hier sollten sich auch Informationen zum Sensor und der Aufnahmegeometrie finden lassen, die dem Operator weitere hilfreiche Informationen für die Analyse liefern. Dazu gehören in erster Linie Parameter wie Sensor, Mode, Zeitstempel der Aufnahme, Orbitrichtung, Polarisierung, Blickwinkel, etc.

Über den „Validate Report“ Knopf soll sich eine übersichtliche Darstellung aller Objekte und

Informationen öffnen, die eine finale Kontrolle erleichtert, bevor der Bericht exportiert werden kann.

3.4. Validierungskampagnen

Zur regelmäßigen Sachstandkontrolle wurden zu Beginn der Studie in Absprache mit allen Verbundpartnern Termine für 3 Validierungskampagnen festgelegt.

Die Validierungskampagnen finden zum Erreichen eines Meilensteins laut Vorhabensbeschreibung [RD-01] statt und dienen zur

- Darstellung des Vorhabenssachstandes aus der Sicht jedes einzelnen Verbundpartners
- Synchronisation des Vorhabenssachstandes zwischen den Verbundpartnern
- Integration des aktuellen Entwicklungsstandes der durch die Verbundpartner beigetragenen Software zu einem Demonstrator
- Vorstellung des Demonstrators
- Bewertung des Reifegrades des Demonstrators
- Gegenüberstellung und Bewertung des aktuellen Sachstands im Vergleich zum erreichten Meilenstein
- Reflexion des aktuellen Sachstandes im Vorhaben.

Die Koordination der Verbundtreffen erfolgte durch den Verbundleiter Airbus DS Bremen. Wegen der zu Beginn des Jahres 2020 einsetzenden globalen Corona-Pandemie und der daraus resultierenden Einschränkungen bezüglich körperlichen Zusammentreffens, konnte lediglich Validierungskampagne I im geplanten Veranstaltungsrahmen durchgeführt werden. Die Validierungskampagnen II und III wurden als Tele-Veranstaltungen ausgelegt.

Aus Sicht des Teilvorhabens IntelliMar Services konnten folgende Ergebnisse aus den Validierungskampagnen gewonnen werden.

3.4.1. Ergebnisse der Validierungskampagne I

- Teilnehmer: Vertreter aller Verbundpartner
- Termin: 08. Mai 2019
- Ort: Verbundpartner OFFIS, Oldenburg

Ein technischer Demonstrator bestehend aus Erkennungsalgorithmen der Verbundpartner DLR (2 verschiedene Implementierungen) und Airbus DS Ulm unter Nutzung des DSGs des Verbundpartners Airbus DS Bremen wurde vor Ort realisiert und mit durch den Verbundpartner OFFIS bereitgestellten Testdaten betrieben. Mit einem allen Verbundpartnern bekannten und vorab prozessierten Datensatz wurde die Funktionsfähigkeit des Demonstrators geprüft. Mittels eines Testdatensatzes, dessen Erkennungsergebnisse nur dem Verbundpartner OFFIS vorlagen, wurde die Erkennungsfähigkeit und Erkennungsqualität geprüft. Im Rahmen der Validierungskampagne I wurde entsprechend des Meilensteinplans nur die Anomalie vom Typ S105, Drastic (> 30°) change of course at

open seas, geprüft. Das Ergebnis ist aus Sicht des Verbundpartners Airbus DS Ulm für das Teilvorhaben IntelliMar Services sehr gut; es ergab sich nach minimalen Anpassungen eine 100 % Erkennungsrate im verdeckten Testdatensatz.

In der Diskussion der Ergebnisse ergaben sich folgende Punkte (Auswahl der für das Teilvorhaben IntelliMar Services relevanten Ergebnisse.)

- Selbst eine einfache Anomalie, wie die hier getestete S105 eröffnet viele Facetten in der Detailbetrachtung. Neben den Parametern wie Minimumfahrt, untere Zeitschranke, Art des Schiffs und der gefahrenen Winkelsumme können auch Aspekte wie Schlangenlinienfahrt, Umfahrung eines Hindernisses mit Rückkehr zur Originalroute für die Einschätzung der Anomalie relevant sein. Zukünftig sollen die Anomalien bezüglich ihrer Sonderfälle analysiert und möglichst berücksichtigt werden. Diese Erkenntnis führte im Teilvorhaben IntelliMar Services zu einer Remodellierung von der Betrachtung eines singulären Anomalie-Ereignisses hin zum Strom von Ereignis-Updates um Relevanzbetrachtungen im Verlauf der Erkennung durchzuführen.
- Es wurde erkannt, dass für den reibungslosen Ablauf einer Demonstration die Vorbereitungszeit bezüglich der Integration deutlich erhöht und berücksichtigt werden muss.
- Der Meilenstein I wurde erfüllt.

3.4.2. Ergebnisse der Validierungskampagne II

- Teilnehmer: Vertreter aller Verbundpartner
- Termin: 23. Juni 2020 – 25. Juni 2020
- Ort: Telefonkonferenzen

In dieser Validierungskampagne standen 2 Themen im Fokus: Zum einen sollte das erweiterte Spektrum an implementierten Anomalie-Detektoren in gleicher Weise wie in Validierungskampagne I mit einem bekannten und verdeckten Datensatz getestet werden, zum anderen wurde die Nutzung des DSGs und seiner erweiterten und geänderten API im Verbund getestet.

Folgende Ergebnisse wurden für das Teilvorhaben IntelliMar Services erzielt:

- Die Funktionsfähigkeit der Anomalie-Detektoren zu den folgenden Anomalien wurde nachgewiesen:
 - H208 (The new position / travelled distance after an AIS signal loss does not appear consistent to the previous route)
 - S104 (Small fast boat proceeding at high speed in open sea)
 - S105 (Drastic (> 30°) change of course at open seas) – Verbesserungen seit dem Stand zu Validierungskampagne I
- Die Funktionsfähigkeit der Verbindung und Anomalie-Übermittlung durch das DSG wurde nachgewiesen. Es wurde jedoch erkannt, dass sowohl an dem API selbst als auch an der Implementierung des Datenaustauschs intensiver gearbeitet werden

muss als vorgesehen war. Dies führte auch für das Teilvorhaben IntelliMar Services zu einer Verschiebung in der ursprünglichen Zeit- und Ressourcenplanung.

- Der Meilenstein II wurde erfüllt.

3.4.3. Ergebnisse der Validierungskampagne III

- Teilnehmer: Vertreter aller Verbundpartner
- Termin: Kalenderwoche 3 bis 5 des Jahres 2021
- Ort: Telefonkonferenzen

Die letzte Validierungskampagne kurz vor Ende des Gesamtvorhabens IntelliMar stellte den Rahmen für den Nachweis der Funktionsfähigkeit des Gesamt-Demonstrators laut Meilensteinplanung dar. Der große Funktionsumfang sowie der Integrationsaufwand erforderten einen deutlich größeren Zeitrahmen als bei den Validierungskampagnen I und II.

Aus der Sicht des Teilvorhabens IntelliMar Services wurde der gesamte geplante Funktionsumfang eingebracht und wie in Validierungskampagne I und II einem offenen und verdeckten Test unterzogen. Besonderes Augenmerk wurde auf die Zuverlässigkeit des Anschlusses an das DSG gelegt. Zu allen Anomalie-Detektoren verliefen die Tests fehlerfrei, auch die Integration zeigte sich erfolgreich, sodass auch der Meilenstein III als erfüllt betrachtet wird.

3.5. Wichtige Positionen des zahlenmäßigen Nachweises

Die Zuwendungen wurden größtenteils für Personalkosten verwendet. Die technischen Arbeitspakete erforderten keinerlei Sachmittel, ebenso wurden keine Fremdleistungen in Anspruch genommen. Reisekosten nahmen einen geringeren Teil der Zuwendungen ein als ursprünglich kalkuliert. Aufgrund der im Frühjahr 2020 eingetretenen Reise- und Kontaktbeschränkungen wegen des neu aufgetretenen Coronavirus SARS-CoV-2 konnten einige der geplanten Meetings und Präsentationen nicht wie geplant stattfinden und wurden virtuell durchgeführt.

3.6. Notwendigkeit und Angemessenheit der geleisteten Arbeit

Um die in den Kapiteln 2 und 3 beschriebenen Resultate zu erzielen und einer kommerziellen Nutzbarkeit einen Entwicklungsschritt näher zu kommen, war die im Rahmen der Arbeitspakete AP 1200, AP 1500 und AP 2100 geleistete Arbeit notwendig und angemessen. Die durch Kooperationspartner zur Verfügung gestellten AIS Daten waren kostengünstige und notwendige Referenzdaten um die im Rahmen von IntelliMar entwickelten Anomalien zu identifizieren. Dies ist nur in einem derartigen Kooperationskonsortium mit fachspezifischer Zuweisung der Arbeiten unter den Kooperationspartnern in einem angemessenen Kostenrahmen möglich.

3.7. Verwertbarkeit der Ergebnisse

Die im Rahmen des Arbeitspaketes AP 1500 erzielten Ergebnisse (Siehe Kapitel 3.2) sind direkt in das bestehende Software Produkt „Activity Based Intelligence (ABI)“ von Airbus Defence and Space GmbH eingeflossen und haben im Wesentlichen zu einer verbesserten Erkennung von Maritimen Anomalien und zur Erweiterung der Fähigkeiten des Produktes geführt.

Neben der Erkennung der im Rahmen von IntelliMar behandelten Anomalien sind vor allem die dahinterliegenden Technologien in das ABI Produkt eingeflossen, welche es ermöglichen in Zukunft weitere Erkennungsalgorithmen einzupflegen.

Über das Einbinden von ABI in weitere Produkte von Airbus Defence and Space GmbH haben eine Vielzahl von Software Assets Zugang zu den Technologien und Fähigkeiten aus IntelliMar.

Der im Arbeitspaket AP 1200 entwickelte Data Lake trägt ebenfalls wesentlich zur Fähigkeitenerweiterung von ABI und damit indirekt zahlreicher Softwareprodukte von Airbus Defence and Space GmbH bei.

Die Ergebnisse des Arbeitspaketes AP 2100 geben eine sehr gute Zielrichtung vor, in welche sich die Anforderungen internationaler maritimer Anwender und Kunden entwickeln. Auch zeigen die Ergebnisse in welchen Bereichen Bedarf nach neuen Lösungsansätzen besteht. Airbus Defence and Space GmbH nutzt diese Ergebnisse um zukünftige Eigenmittel zielgerichtet in die konkreten Nutzerbedarfe und –wünsche zu investieren.

3.8. Fortschritt auf dem Gebiet des Vorhabens bei anderen Stellen

Es sind keine Arbeiten - oder Fortschritte - auf dem Gebiet des Vorhabens bei anderen Stellen bekannt geworden.

3.9. Veröffentlichungen

Begleitend zur IntelliMar Studie entstand im Teilvorhaben Services eine Master-Arbeit:

Titel	Maritime Situationserkennung mittels dynamischer Bayes'scher Netze
Hochschule	Technische Hochschule Nürnberg
Studiengang	Masterstudium Software-Engineering und Informationstechnik
Bearbeiter	Steffen Haaga
Datum der Abgabe	20. Mai 2020
Note	1,0

Tabelle 3-33: Daten zur Masterarbeit, welche begleitend zur IntelliMar Studie entstand