Grant Agreement No: 687591

30/6/16

Big Data Analytics for Time Critical Mobility Forecasting

datAcron

D1.1 Requirements Analysis

Deliverable Form	
Project Reference No.	H2020-ICT-2015 687591
Deliverable No.	1.1
Relevant Work Package:	WP 1
Nature:	R
Dissemination Level:	PU
Document version:	1.0
Due Date:	30/06/2016
Date of latest revision:	29/06/2016
Completion Date:	29/06/2016
Lead partner:	UPRC
Authors:	Christos Doulkeridis, Akrivi Vlachou, Giorgos Santipantakis, Apostolos Glenis, George Vouros
Reviewers:	Georg Fuchs
Document description:	This deliverable specifies the requirements analysis for the datAcron integrated system.
Document location:	WP1/Deliverables/D1.1/Final

HISTORY OF CHANGES

Version	Date	Changes	Author	Remarks
0.1	1/3/2016		Christos Doulkeridis	First version of table of contents
0.2	7/3/2016		Christos Doulkeridis	Revised table of contents
0.3	5/4/2016		Christos Doulkeridis	Added information concerning use-cases
0.4	3/5/2016		Christos Doulkeridis	Initial templates for describing requirements
0.5	24/5/2016		Christos Doulkeridis	Finalized templates for describing requirements
0.6	1/6/2016		Christos Doulkeridis	Added requirements for WP1
0.7	8/6/2016		Christos Doulkeridis	Added requirements for WP3, based on input received by Elias Alevizos
0.8	9/6/2016		Christos Doulkeridis	Added requirements for WP2, based on input received by Nikos Pelekis
0.9	10/6/2016		George Vouros	Internal review within WP1 and homogenization
1.0	15/6/2016		Christos Doulkeridis	Version sent to all partners for review
1.5	25/6/2016		Christos Doulkeridis	Refined version after collecting input by all WPs
2.0	28/6/2016		Christos Doulkeridis	Version sent to Georg Fuchs for internal review
2.01	29/6/2016		Georg Fuchs	Internal review and proof reading

EXECUTIVE SUMMARY

This report comprises the first deliverable (D1.1) of datAcron work package 1 "System architecture and data management" with main objective to capture the requirements for the datAcron integrated system. The requirements analysis is guided by the information captured in the deliverables that describe the use-case scenarios for the maritime (D5.1) and aviation (D6.1) domains, but also takes into account the Grant Agreement. A major contribution of this deliverable is to bridge the gap between use-case scenarios and system design, by specifying functional requirements of the distinct datAcron components, paying special attention to their requirements from the data management components. Also, D1.1 aims at defining core integration and interaction hotspots among all components, figuring out specific patterns of interactions that are of crucial importance to the functionality of the overall system and which need to be considered in a meticulous way when designing the datActon integrated system.

The first part of this document provides background information regarding the specific objectives and research challenges of the project, a brief overview of the use-cases, a classification of the data sources according to their domain and modality, and an overview of the overall datAcron architecture. Then, in the second part, the functional and architectural requirements are presented for each datAcron component, and their interactions are specified in terms of inputs/outputs. Moreover, requirements are linked to the research objectives of datAcron and individual use-case scenarios, and suitable performance metrics are recorded in order to assist the validation process.

TABLE OF CONTENTS

TERMS & ABBREVIATIONS

ADS-B	Automatic Dependent Surveillance-Broadcast
AIS	Automatic Identification System
ATC	Air Traffic Control
ATM	Air Traffic Management
CMEMS	Copernicus Marine Environment and Monitoring Service
FAO	Food and Agriculture Organization
FIR	Flight Information Region
METAR	Meteorological Terminal Air Report
METOC	Meteorological and Oceanographic
MFS	Mediterranean Sea Physics Analysis and Forecast
NOAA	National Oceanic and Atmospheric Administration
RDF	Resource Description Framework
STAR	Standard Terminal Arrival Route
SID	Standard Instrument Departure

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

This document is the deliverable D1.1 of task 1.1 “Requirements Analysis” of work package 1 ”System Architecture and Data Management” of the datAcron project. It defines requirements for the datAcron integrated system, by also considering the different data sources and the detailed use-case scenarios described in deliverables D5.1 “Maritime Use Case Detailed Definition” and D6.1 “Aviation Use Case Detailed Definition”.

The deliverable goes one step further towards the detailed design of the datAcron integrated system by compiling information concerning the specification of use-case scenarios and data sources, by specifying functional requirements of the distinct datAcron components, paying special attention to their requirements from the data management components, defining core integration and interaction hotspots among all components, figuring out specific pattern of interactions that are of crucial importance to the functionality of the overall system and which need to be considered in a meticulous way when designing the datActon integrated system: As said, the objective is the design and development of the datAcron Big Data architecture, integrating all datAcron research components devised from work packages 1,2,3 and 4.

Most of the specified requirements present research challenges to the management of data, considering the different data sources, the modality of data (being data-in-motion, or data-at-rest), the results computed by the different research components, and of course the requirements of these components for the provision of data.

Each requirement is formally specified with validation criteria, while quantified validation criteria are specified in preference whenever applicable so that the main concepts of the project will all be validated. Obviously, requirements and criteria are linked to the research objectives of the project, making sure that all work packages, as well as the project as a whole, are oriented towards addressing their research objectives and towards integrating the research components, presenting a coherent overall system that will be validated and evaluated according to the operational requirements of aviation and maritime use-case scenarios.

1.1 Purpose and Scope

The requirements analysis of the datAcron project addresses the following objectives: (a) documents the specific functionality required by the datAcron integrated system and connect this functionality with specific use-case scenarios, (b) guides the specification and design of the datAcron architecture, and (c) clearly assigns roles to the individual components of the overall datAcron system, specifying also their interoperation, i.e., the required functionality to support the identified use-cases and scenarios towards achieving the project research objectives.

This report is delivered in M6 of the project. However it may evolve by identifying further needs or by refining requirements during the course of the project, as the specific experiments for validating the research components to be developed and the overall datAcron intergrated system are to be determined. This is expected to happen to a minor degree.

1.2 Approach for the Work package and Relation to other Deliverables

The current deliverable D1.1 is tightly connected to the deliverables D5.1 "Maritime use case detailed definition" and D6.1 "Aviation use case detailed definition" describing the maritime and aviation use-cases respectively. D1.1 identifies and documents the requirements of the datAcron integrated system after careful inspection and study of the detailed use-case descriptions, their implications to the functionality of the datAcron integrated system components and to the datAcron integrated system, and their relation to the research objectives of the project.

The main objective of this deliverable is to produce a set of necessary and sufficient requirements that cover all end-user needs expressed in the descriptions of the use-cases and scenarios. Moreover, at a second level, D1.1 aims at identifying requirements that will be satisfied by individual system components or by specific combinations of components. In addition, D1.1 identifies requirements that are in the intersection of use-case requirements, i.e., from both maritime and aviation domains.

It must be pointed out that D1.1 is prepared during the same time that deliverables D5.2 "Maritime data preparation and curation" and D6.2. "Aviation data preparation and curation" are prepared. Both these deliverables aim at providing more detailed descriptions of the data sources that relate to the use-case scenarios. As such, they are also important to this deliverable, and intermediate versions of D5.2 and D6.2 have been considered during the preparation of the current deliverable.

With respect to the subsequent activities and tasks of the datAcron project after the 6th month (M6), the output of D1.1 is particularly important for D1.2 "Architecture specification". The functional and architectural requirements reported in this deliverable will guide the specification and design of the datAcron integrated architecture. Furthermore, this deliverable is also linked to deliverables D5.3 "Maritime experiments specification" and D6.3 "Aviation experiments specification", specifying in detail the domain-specific experiments to be done, specifying the datasets to use per experiment, the steps to perform, what to measure and the targeted values. As a potential result, validation criteria specified in this document will be further refined according to the specifications in deliverables D5.3 and D6.3.

1.3 Methodology and Structure of the Deliverable

The overall approach for presenting the requirements of the datAcron integrated system is based on meticulous study of the use cases scenarios, and follows a "requirement-centric" methodology, where the basic structural unit is the functional requirements that stem from the use cases and scenarios defined. Each functional requirement is presented in terms of inputs, outputs, functions that operate on inputs and transform them to output, as well as validation criteria that indicate measurable goals.

The remaining of this report is structured as follows. Section 2 briefly overviews the necessary information that guides the requirements analysis, namely the research objectives of datAcron, the description of the use-case scenarios, and an overview of the data sources considered in datAcron. Section 3 describes the architecture of datAcron as well as key issues relevant to the architecture. Section 4 presents the architectural requirements of the datAcron integrated system, in terms of interoperability between the different software components, thus identifying the core integration points between work packages. In addition, it documents the functional

requirements of the datAcron integrated system, thus clearly presenting its operation in relation with the specific use-case scenarios. Section 5 aggregates the information related to requirements and organizes it in different ways, with respect to research objectives and use-case scenarios.

2 BACKGROUND

This section provides the necessary background information that is a prerequisite for performing and understanding the requirements analysis. It replicates information from the Description of Action and from other deliverables, but it is necessary for making this document self-contained.

In the first place, a brief overview of the research objectives of datAcron is presented: Requirements must be linked to these objectives. Then, an overview of the two use cases is presented for the maritime and aviation domains, respectively. Guided by these use-cases and detailed scenarios, the specification of analysis requirements is performed. Finally, the data sources that will be used in the datAcron project are reviewed, as their characteristics have implications on the specification of the requirements.

2.1 Research Objectives

The specific research objectives (O.x) of the datAcron project together with specific research challenges (O.x.y) as documented in the Description of Action comprise the following:

O.1 Scalable integration and management of data from disparate and heterogeneous sources

datAcron will implement ready-to-go integration connectors for the seamless integration of multiple, voluminous and heterogeneous data-in-motion and data-at-rest data sources. Towards this objective, datAcron will deliver components for integrating and summarizing multiple streaming data sources as they flow into the system databases and processing components - following the in-situ processing paradigm- producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing. The objective is to produce streaming data synopses at a high-rate of compression, without compromising the detection and forecasting accuracy of entities activities.

Summarized streaming data will be semantically integrated with archival data, as well as with detected and forecasted trajectories and events via the development of novel solutions for supporting the efficient integration, management and querying of spatio-temporal data.

- O.1.1 : Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates
- O.1.2 : Automatic, real-time semantic annotation and linking of data towards generating coherent views on integrated cross-streaming and archival data
- O.1.3 : Efficient distributed management and querying of integrated spatio-temporal data

O.2 Real-time detection and forecasting of trajectories

datAcron will develop a trajectory detection and forecasting component, involving the development of novel algorithms for the cross-streaming real-time detection of trajectories and algorithms for short- and long-term forecasting / prediction of trajectories, supporting outlier detection, taking advantage of data analytics results over archival data and recognized complex events.

In close relation to these objectives, datAcron will develop advanced data analytics methods and tools over contextually enhanced trajectories of moving entities, exploiting integrated data-in-motion and data-at-rest.

O.2.1 : Cross-streaming, real-time detection of the trajectories of moving entities

O.2.2 : Data analytics over the trajectories of moving entities

O.2.3 : Short- and long-term real-time forecasting of trajectories

O.3 Real-time event recognition and forecasting.

datAcron will develop adaptive complex event recognition and forecasting technology that is able to benefit from data-in-motion and data-at-rest from multiple, disparate, voluminous data sources. Towards this objective datAcron will produce methods for real-time event recognition under uncertainty, with noisy and fluctuating data, exploiting a library of complex event patterns suitable for recognition and forecasting, and for adapting the event patterns in dynamic environments. Concerning events forecasting, datAcron will further advance event recognition algorithms to indicate the probability of a forecasted event, as well as the probability of when an event will happen, making these algorithms resilient to the lack of veracity in the data.

O.3.1 : Real-time event recognition and forecasting algorithms that take full advantage of the data provided

O.3.2 : Methods for adapting event patterns in dynamic settings

O.3.3 : Resilient real-time event recognition and forecasting algorithms addressing lack of veracity of data

O.4 Real-time interactive analytics.

datAcron aims at developing interactive scalable Visual Analytics (VA) methods and tools that are able to handle efficiently both archival and streaming spatio-temporal data, with varying levels of resolution and quality. Specifically, this objective includes the development of methods for data exploration and assessment of data quality; for interactive pattern extraction from data; for user-guided model building and validation; for building situation overview and situation monitoring.

O.4.1 : VA methods for data exploration and assessment of data quality considering data-in-motion and data-at-rest from multiple sources

O.4.2 : VA methods for interactive pattern extraction from data-in-motion and data-at-rest coming from multiple sources

O.4.3 : VA methods for user-guided model building and validation

O.4.4 : VA methods for building situation overview and situation monitoring

O.5 Validation and evaluation of the datAcron system and individual components on the surveillance of moving entities in the ATM and marine domains.

The validation and evaluation of the integrated datAcron prototype system and of the constituent methods will be performed according to the specific scenarios specified by the use case partners in close collaboration with the use case interest groups, in the ATM and maritime domains. In both domains, there will be different scenarios of data growth, also defining cases where data sources provide data at different levels of quality and veracity; partners will specify the exact data sources to be used, will determine data sources, while defining the detailed validation and evaluation methodologies per scenario of use, with the expected impact specified for the relevant metrics. Visual Analytics methods will be evaluated by objective user performance and subjective user satisfaction quantitative

criteria. For selected critical tasks, eye-tracking evaluation of problem solving activity will be performed for identifying common mistakes and suboptimal problem solving strategies.

All constituent methods and the integrated datAcron prototype will be capable of handling very large numbers of moving entities at sea and in the air across large geographical areas.

The aim is to validate and evaluate

- The datAcron integrated prototype system to support data-intensive tasks in different scenarios of data growth, data quality and veracity of data in the domains of focus.
- The individual components of the datAcron prototype system to satisfy the evaluation criteria and key-performance indicators, specified for their purposes.

Beyond the above-mentioned research objectives, according to the Description of Action there are specific objectives concerning the interaction between components, interaction with the users and in-situ data processing:

- All analytics components can take full benefit of the computations of others, also taking advantage of interlinking between their results. Thus, the trajectory detection and forecasting methods can benefit from events detected or forecasted and vice-versa. Similarly for the visual analytics methods;
- Users can interact and explore data, via integrated data views, being supported for decision-making;
- Advanced processing of data close to the data sources (following the in-situ data processing paradigm) must be performed.

2.2 Overview of Use-cases

In this deliverable, after carefully studying the individual use-case scenarios, the requirements stemming from these scenarios are decoded, common or overlapping requirements between different scenarios are identified, and classification of the required functionality is provided.

Concerning the maritime domain, as defined in D5.1, in order to support datAcron challenges, several scenarios involving fishing activities have been considered. These scenarios highlight the needs for continuous (real-time) tracking of fishing vessels and surrounding traffic, require contextually enhanced data analytics, including for instance cluster and spatial analysis as well as motion pattern detection. Scenarios have been specified so as to stress datAcron data management components and analytics methods in terms of velocity, veracity, variety and volume of data, and provide a complete support for trajectory and event detection, prediction and visualization. From an operational point of view, scenarios concern fishing security and control. For each scenario, D5.1 describes the specific user information needs, as related to the Maritime Situational Indicators specified in the deliverable. The specific scenarios defined for the maritime use-case are listed in Table 1.

As far as the aviation use case is concerned, as D6.1 defines, the aim is to increase predictability (or, equivalently, reduce uncertainty). The Big Data approach applied to trajectory prediction can help to this improvement, applying data-driven analytics techniques (i.e., learning from historical data), taking into account the Big Data characteristics of data from the various sources. The two scenarios selected for the use case concern Flow Management and Flight Planning.

Scenario	Description
SC11	Collision avoidance
SC12	Vessel in distress / Man overboard
SC21	Monitoring maritime protected area (from illegal fishing)
SC22	Fishing pressure on areas
SC31	Detection of migrants / refugees and human trafficking
SC32	Illicit activities

Table 1: Maritime use-case scenarios documented in deliverable D5.1.

Both scenarios are split in increasingly complex smaller scenarios which are related to datAcron components and technical work packages, so the coverage of the different datAcron developments can be easily mapped. Flow Management scenarios' main objective is to allow better planning of the demand and capacity balance, which will lead to fewer delays. Flight planning scenarios' main objective is to enhance the trajectory prediction to avoid plans prone to great deviations over the day of operation. Both scenarios leverage the analysis of historic data related to Flight Plans, Contextual ATM data, Surveillance data, Weather data, Flow Management (regulations). The specific scenarios defined for the aviation use-case are listed in Table 2.

Scenario	Description
FP01	Real trajectory reconstruction
FP02	Real trajectory enrichment
FP03	Event recognition in trajectories
FP04	Event forecasting in trajectories
FP05	Data set preparation
FP06	Trajectory clustering
FP07	Trajectory prediction - preflight
FP08	Trajectory prediction - preflight schedule based
FP09	Trajectory prediction - real time
FP10	Trajectory comparison
FM01	Regulation prediction
FM02	Demand and capacity prediction
FM03	Resilience assessment

Table 2: Aviation use-case scenarios documented in deliverable D6.1.

2.3 Overview of Data Sources

The specific sources that domains aim to exploit for realizing the scenarios specified are detailed and categorized according to their modality: Being data-at-rest (archival data) or data-in-motion (streaming data). Subsequently, we recall the data sources prescribed per domain, as well as the modality of the source.

Sources are identified by the initial of their domain (M/A), being surveillance (S), contextual (C), regulations (R), weather (W), reported-events (RE), events recognized (E), or trajectories recognized and/or forecasted (T), other (O).

2.3.1 Maritime domain

Table 3 shows the categorization for maritime data sources in datAcron.

2.3.2 Aviation domain

Table 4 shows the categorization for aviation data sources in datAcron.

	Data source	**Modality**
MS1	Automatic Identification System (AIS) datasets, (kinematic and static messages) by coastal receivers	Streaming spatio-temporal data
MS2	Automatic Identification System (AIS) datasets (kinematic and static messages), by satellite receivers	Streaming spatio-temporal data
MS3	Automatic Identification System (AIS) datasets (kinematic and static messages), by coastal receivers	Compressed Streaming spatio-temporal data
MS4	Automatic Identification System (AIS) datasets (kinematic and static messages), by satellite receivers	Compressed Streaming spatio-temporal data
MC1	EU regulated fishing areas	Archival spatial data
MC2	FAO and ICES fishery statistical areas	Archival spatial data
MC3	Community Fishing register	Archival data for vessels
MC4	European Marine Observation and Data Network	Archival spatial data (coastal maps, ports, fishing areas, etc.)
MC5	Marine protected areas NATURA 2000	Archival spatial data
MC6	Environmental biodiversity datasets	Archival spatial data
MC7	World Port Index	Archival spatial data
MC8	Routes	Archival spatio-temporal data (it includes a sequence of points/areas)
MW1	METOC data from different sources (MFS, CMEMS, ECMWF, Andriatic forecasting system etc)	Low-pace spatio-temporal streaming data that may be considered archival in datAcron terms
MRE1	Events reports (blacklists from IUU fishing vessels, worldwide threat to shipping reports, Interpol reports, piracy reports)	Archival textual data maybe with spatio-temporal information
MT1	Trajectories (detected / forecasted, open or close)	Streaming spatio-temporal data
ME1	Events (recognized /predicted)	Streaming spatio-temporal data

Table 3: Overview of maritime data sources.

	Data source	**Modality**
AS1	Radar tracks (IFS) from ATC providers	Streaming spatio-temporal data
AS2	ADS-B messages broadcasted by airplanes	Streaming spatio-temporal data
AO1	Airline schedule	Archival static data
AO2	Flight plans	Archival spatiotemporal data
AR1	Flow Management Data (related to regulation data)	Archival spatio-temporal data related to sectors and flights, delays
AC1	Sector Configuration	Archival Spatial Data
AC2	Airports database	Archival Spatial Data
AC3	Aircrafts database	Archival Spatial Data
AW1	Metar data	Low-pace spatio-temporal streaming data that may be considered archival in datAcron terms
AT1	Trajectories (detected / forecasted, real or synthetic, open or close)	Streaming spatio-temporal data
AE1	Events (recognized /predicted)	Streaming spatio-temporal data

Table 4: Overview of aviation data sources.

3 OVERALL VIEW OF THE datAcron ARCHITECTURE

3.1 The datAcron Concept

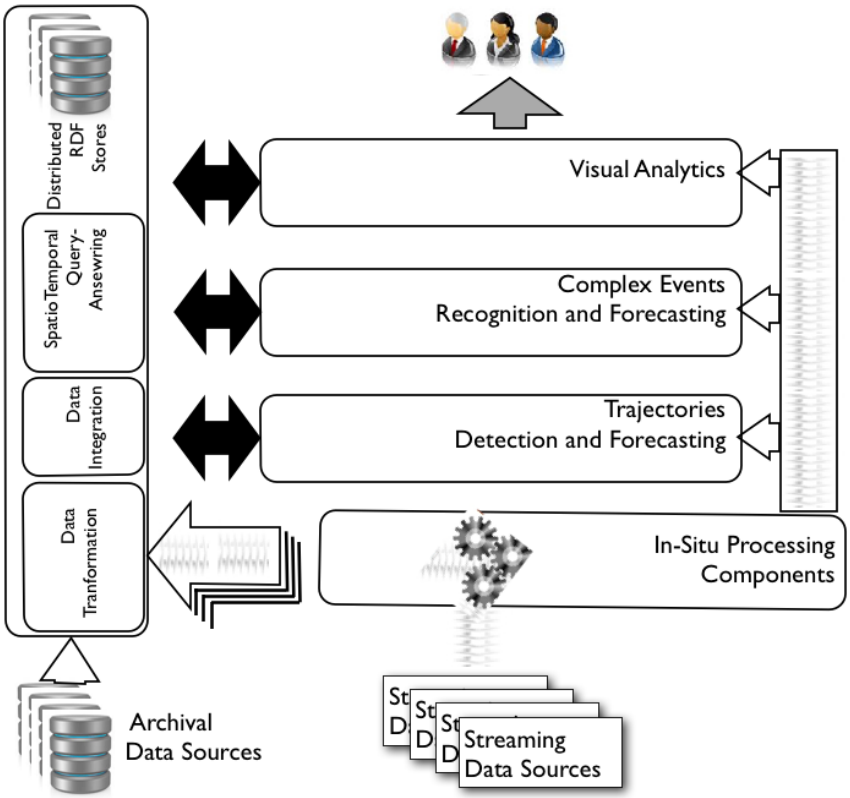

Figure 1: datACRON overall architecture.

The datAcron concept is demonstrated by the overall architecture shown in Figure 1 (reported in the Description of Action [2]), providing insights and evidence on how the above-mentioned research challenges and objectives are going to be addressed towards providing a coherent Big Data solution. In particular, the targeted functionality is separated to work packages as follows:

- WP1: In-situ processing components and distributed RDF store, including data transformation, data integration and spatio-temporal query answering
- WP2: Detection and forecasting of trajectories
- WP3: Recognition and forecasting of complex events
- WP4: Visual analytics

Specifically:

- The **data sources** are multiple streaming data sources, as well as archival data sources.
- The **in-situ processing components** aim to compress and integrate where appropriate data-in-motion from streaming sources in communication efficient ways computing single and multi-streaming data synopses at high rates of data compression – without affecting the quality of analytics – capitalizing on low-level primitive operators (e.g. selections and projections, joins to cater for cross-stream processing as close to the data as possible, etc) that are applied directly on the data streams.
- The **data transformation components** aim to convert data in (a) single and multi-streaming data synopses, (b) archival data and (c) results from the datAcron higher levels analytics components to a common form.
- The **data integration component** interlinks semantically annotated data using link discovery techniques for automatically computing correspondences between data from disparate sources. This produces integrated data views, including data and their correspondences. Integrated data are provided to the analytics components in real-time, while they are also stored in parallel stores.
- The **spatiotemporal query-answering component** provides parallel query processing techniques for spatio-temporal query languages. Interlinked data (processed and compressed data-in-motion and linked data-at-rest) are stored in parallel RDF stores, using sophisticated RDF partitioning algorithms in domain specific, spatial and temporal dimensions.
- The **data analytics components** include trajectory and complex event recognition and forecasting, as well as visual analytics. These consume the data provided by the data integration component: Synopses computed by the bottom layer, being integrated (where necessary) with archival data.

The data analytics components also use internal stores for frequent and fast data write/read, well tuned to their requirements and to the rest of the architecture, so as to provide real-time results.

The key issues for the datAcron architecture are as follows:

- The data synopses computed near to the sources aim to largely reduce at a high compression rate the streaming data that the data management and analytics layers have to manage. However, access to the raw streaming data is still an option for the analytics components.
- The data synopses computed from multiple streams can already be integrated at the lower processing components (near to the sources). Data synopses and archival data are transformed into a common form according to the dataAcron RDFs schema, are integrated (where necessary) and are pipelined to the rest of the analytics components directly, in real-time. This alleviates the need for analytics components to access datAcron stores frequently.
- "Raw" streaming data are not stored as they enter the system: Persistent storage concerns data synopses, semantically annotated and integrated to archival data, trajectories and events detected. The datAcron stores will provide advanced query answering services for other system components and human or software clients to access these data, according to their requirements on integrated data views.

The above architecture has certain benefits:

- All data from streaming and archival data sources, as well as trajectories and events computed by analytics components can be semantically integrated by discovering links between respective instances, providing semantically-rich coherent views of data. Doing so, datAcron seamlessly annotates trajectories and events with semantic information, and it links these among themselves as well as with the rest of archival and cross-streaming data.
- All analytics components can take full benefit of the computations of others, also taking advantage of interlinking between their results. Thus, the trajectory detection and forecasting methods can benefit from events detected or forecasted and vise-versa. Similarly for the visual analytics methods.
- Users can interact and explore data via integrated data views, being supported for decision-making.

4 ARCHITECTURAL REQUIREMENTS

In this section, the basic requirements for the datAcron integrated system are documented as they have been identified by the use-case scenarios, and by also taking into consideration the research and other project objectives detailed in the Description of Action.

4.1 datAcron Requirements on Big Data

The datAcron use-cases and their individual peculiarities pose requirements relevant to the most typical issues and challenges encountered when managing Big Data [3]. Several of these challenges are also reported in the well-known Beckman report [1] about database research self-assessment.

- *Volume*: The amount of data managed and analyzed by datAcron is extremely large and also getting larger every minute, as new data are produced with varying rates. Data ingested by the datAcron integrated system include surveillance data, weather data, contextual data, as well as other data from diverse sources related to the maritime and aviation domains. Some of these sources are particularly voluminous, as for example is the case for weather forecasts that are described by multiple variables at high resolution, and are updated as the time evolves. In addition, historic data (data-at-rest) are to be integrated in datAcron, in order to be exploited for pattern discovery and for the predictive analytics operations over trajectories and complex events. Last but not least, the integration/interlinking task performed in datAcron provides valuable new information in the form of links between related entities, which also contributes to the increase of the amount of managed data.

- *Velocity*: In datAcron, several incoming sources are streaming data sources, whose data must be linked with data originated from other sources in real-time. Managing multiple data streams of varying rate is another Big Data challenge. Due to the individual challenges posed by the specific use-cases in datAcron, processing near to the data sources (in-situ processing) is advocated in the context of datAcron, which is going to accelerate the processing rate of data by "moving" processing as close to the data source as possible. Another critical issue related to the maritime use-case and surveillance data (esp. data received from satellites) is that quite often data arrive delayed in the system, and this discrepancy needs to be handled appropriately, because delayed data may be extremely useful for determining the accurate movement of a vessel.

- *Variety*: Data in datAcron comes in different representations, often semi-structured and in various forms (text and binary data). To cope with the issue of data variety, datAcron relies on a common RDFS schema capable of representing the different concepts, their relations and their properties. By addressing the data variety issue, datAcron can readily combine data, by means of data integration/interlinking, from different sources into this commonly accepted schema, so as to enable the subsequent application of complex mining and machine learning tasks, including prediction algorithms.

- *Veracity*: Several data sources used in datAcron provide noisy data that need to be meticulously cleaned. For instance, it has been observed that almost 35% of AIS messages

contain some form of error, and such errors need to be detected and corrected whenever possible. This poses another requirement for the datAcron integrated system, namely how to effectively deal with inherent noise in the raw data, in order to provide coherent, integrated views of data that are going to be exploited by data analytics and machine learning algorithms.

Another dimension that should be taken into account when pinpointing requirements is the acceptable latency of the computations: datAcron specifies acceptable latency in three levels: *strategic*, *tactical* and *operational*. Thus, all requirements come with their acceptable latency characterization. The latency levels considered in the context of datAcron are provided in Table 5 for clarity.

Latency level	**Time**
Operational	in milliseconds
Tactical	in few seconds
Strategic	tens of seconds or minutes

Table 5: Latency levels in datAcron and their respective temporal duration.

4.2 Requirements for Data Management

In datAcron, data integration is viewed in the most general way, at the level of interlinking different entities: Entities can be specific points in space and/or time, to specific moving objects, other spatial and/or temporal entities, events, or even trajectories, being detected and/or forecasted. In addition, interlinking concerns computing arbitrary, but in-advance well specified relations among entities, beyond the mere identification of their equality: Such relations may be spatial relations (e.g., point within polygon), temporal relation (time point close to another time point), spatio-temporal relations (a point in a trajectory is within a polygon, or two trajectories crossing each other), similarity relations based on specific features of entities (e.g., vessels with similar characteristics, trajectories with high similarity) and combinations of the above (e.g., vessels with similar characteristics, similar trajectories, or frequently crossing trajectories). Subsequently, integration and interlinking are used interchangeably: The aim is to compute a coherent view on data independently of their provenance.

Furthermore, a notable feature of datAcron is that it advocates in-situ processing, close to the data sources, in order to efficiently process data streams of high rates. The in-situ processing entails challenging optimizations related to (distributed) stream processing, for example reduction of processed data, low communication, as well as distributing the computation to multiple nodes. Therefore, a significant issue in the context of the datAcron project is the interplay between in-situ processing and stream processing.

Another important feature of the datAcron data management components is the ability to support advanced querying operations over integrated spatio-temporal data at scale. This issue raises multiple research challenges associated with querying and processing large RDF graphs, which are additionally of spatio-temporal nature. In essence, this raises the need for unified

querying using spatio-temporal predicates together with predicate concerning RDF properties and relations.

Based on the above, the following list of requirements are relevant to the datAcron data management components (WP1):

- R1.1: Real-time integration/interlinking of spatial and/or spatio-temporal entities
- R1.2: Interplay of in-situ and stream processing components
- R1.3: Integration/interlinking over stored data
- R1.4: Spatio-temporal RDF querying of integrated data
- R1.5: Retrieval of spatio-temporally constrained subsets of integrated data

4.2.1 Requirements for the integrated datAcron system

In this subsection, we record a set of architectural requirements for the integrated datAcron system. This set of requirements is derived from the functional requirements of individual components integrated in the datAcron system. The functional requirements of these components are analyzed in detail in the subsequent subsections.

It should be clarified that WP1 encompasses two different processing engines, one for stream (real-time) processing and one for batch processing. The stream processing engine implements the part of the required functionality that needs to be performed in real-time, i.e., at *operational* latency level. Instead, the batch processing engine is responsible for functionality related to integration and querying the RDF data store, and this is performed at *tactical* and *strategic* latency level.

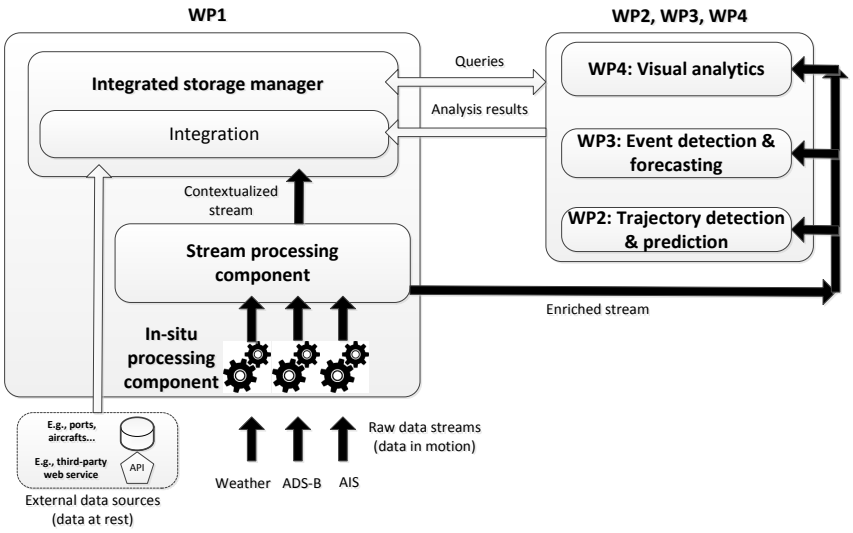

Figure 2: Inputs/Outputs of data management components: black arrows represent streaming data, while white arrows depict interactions that are not necessarily stream-based.

Figure 2 depicts four (4) components that correspond to the research work packages of datAcron, and their interactions from the perspective of WP1.

Inputs WP1 receives input from the following types of main data sources[1]:

- *surveillance data*:
  - maritime use-case: AIS data from coastal receivers (MS1) and satellite receivers (MS2)
  - aviation use-case: radar tracks (AS1) and ADS-B messages (AS2)
- *weather data*: in particular weather forecasts, both related to the conditions at sea, as well as operational weather data at ground-air level and at airports:
  - maritime use-case: METOC data from different sources (MW1)
  - aviation use-case: Metar data (AW1)
- *contextual data*:
  - maritime use-case: EU regulated fishing areas (MC1), FAO and ICES fishery statistical areas (MC2), Community Fishing register (MC3), European Marine Observation and Data Network (MC4), Marine protected areas NATURA 2000 (MC5), Environmental biodiversity datasets (MC6), World Port Index (MC7), and Routes (MC8)
  - aviation use-case: Sector Configuration (AC1), Airports database (AC2), Aircrafts database (AC3)
- *other data sources*: Airline schedule (AO1), Flight plans (AO2)
- *regulations*: Flow Management Data (AR1)
- *trajectories* detected or forecasted:
  - maritime use-case: trajectories (MT1)
  - aviation use-case: trajectories (AT1)
- *events* recognized, predicted or reported:
  - maritime use-case: events (ME1) and maritime events reports (MRE1)
  - aviation use-case: events (AE1)

In addition, WP1 receives input from other WPs regarding *analysis results* that need to be stored and integrated with existing data in the integrated store. Analysis results comply to the following description:

- *WP2 analysis results:* Mainly comprise detected and forecasted trajectories of moving objects. Each trajectory is going to be represented by means of spatio-temporal information (typically a sequence of spatio-temporal points), and an identifier that uniquely determines the moving object (vessel or aircraft) that it concerns. In addition, other kinds of analysis results are going to be stored, such as detected clusters of trajectories, sequential patterns, etc. Furthermore, the involved objects should be available (i.e., the objects whose trajectories conform the cluster or satisfy the sequential pattern), while these patterns should be linked to already existing entities.

[1] The notation introduced in Section 2.3 (Tables 3 and 4) is used to identify data sources.

- *WP3 analysis results:* Every detected event will be associated with the (possibly instantaneous) temporal interval in which it was detected. Moreover, for each event, the list of involved moving objects will be specified and possibly the coordinates of the area in which the complex event was recognized. Based on this information, the detected events are going to be interlinked to already existing entities (e.g., trajectory segments, contextual data, weather data, etc.) in the integrated store, in order to maintain readily available information about which event is associated with which entities.
- *WP4 analysis results:* As a result of visual analyses, input data is enriched and expanded. This includes adding new attributes, such as cluster ids or area associations, to input data on different levels of granularity (single events, trajectories, moving object contributing multiple trajectories), new spatial and temporal objects (e.g., areas of interest, semantically annotated time intervals), as well as new entity relations (RDF triplets) and refined parametrization for models used in online processing.

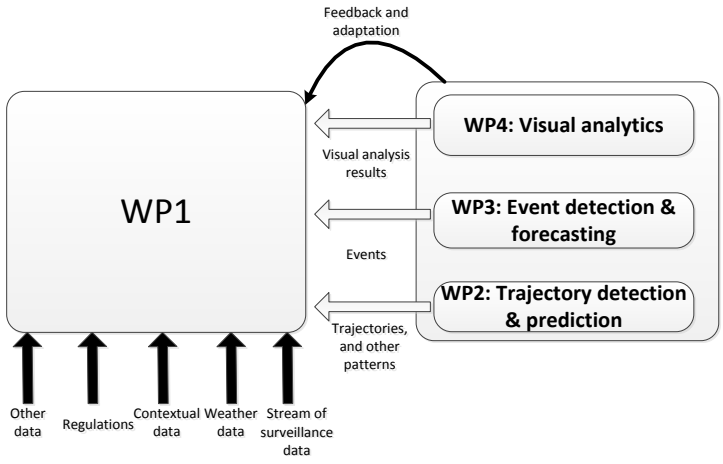

Figure 3: Inputs to data management component.

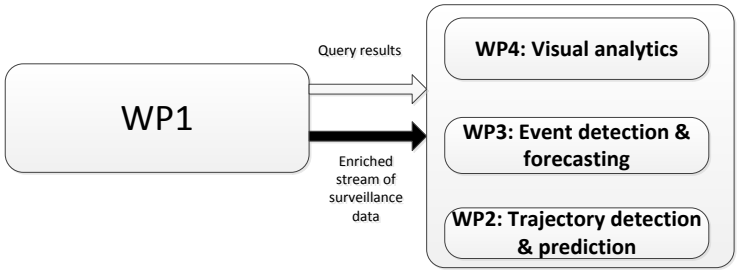

Figure 4: Outputs of data management component.

All afore-described inputs to WP1 are converted to RDF according to a specific schema and are integrated/interlinked, resulting to a coherent RDF graph.

Finally, WP1 takes as input in the form of feedback (mainly) from WP4 the parameterization of the in-situ processing component, as well as input in the form of parameters that adapt the behavior of the real-time interlinking functionality. A graphical illustration of all the above inputs to WP1 is depicted in Figure 3.

Outputs WP1 provides output within the integrated datAcron system, namely to all other WPs (WP2–WP4), as follows:

- An enriched stream of moving object positions. This stream includes positional information, enriched with weather and additional contextual information that is associated to the positions of moving objects (e.g., relations to spatial areas of interest, etc.).
- An interface for invoking batch processing tasks over the data in the integrated store, thus allowing other WPs to retrieve integrated data based on different criteria, including spatio-temporal constraints, information about moving objects (e.g., types of moving objects, their characteristics, etc.), weather forecasts, specific configurations, sectors, regulations, flights and flight plans, etc.

A graphical illustration of all the above outputs of WP1 is depicted in Figure 4.

4.2.2 Requirement 1.1: Real-time integration/interlinking of spatial and/or spatio-temporal entities

This requirement addresses the need to integrate surveillance data with contextual and weather data in real-time, as the stream of positions of moving objects is consumed. In more detail, the position of a moving object (vessel or aircraft) is going to be interlinked with spatial or spatio-temporal entities in real-time. Typically, this position comes from streaming surveillance data[2]. In the maritime use-case, the main source will be a stream of AIS messages. In the aviation use-case, surveillance data coming from ADS-B or radar tracks are going to be used.

Inputs The first input is a stream of positions of moving objects, $\{x, y, t\}$ or $\{x, y, z, t\}$, for maritime and aviation respectively.

The following entities comprise the inputs necessary for implementing the desired functionality of integration/interlinking in real-time:

- *Weather data:* Include weather-related information, in the form of forecasts for specific spatial areas and a future temporal instance or temporal interval. A potentially different set of weather-related variables of interest is deemed more useful for each domain. As an example, typical variables of interest for the aviation use-case include: wind speed, temperature, and geopotential height.
- *Contextual data:* The following input sources have been identified from the use-case scenarios, and constitute contextual data sources:
  - sector data (in the case of aviation), defined as geometric 3D regions that constitute a partitioning of the air space
  - regulations (in the case of aviation), defined by appropriate ATC variables
  - areas of interest (in the case of maritime), defined as spatial 2D polygons describing protected areas.

[2]However the position can also come from other sources, such as positions that belong to a model-based predicted trajectory.

Outputs The output is going to be positions of moving objects found in the input stream interlinked with spatial or spatio-temporal entities. In other words, some of the recorded moving object positions in the input stream are going to be augmented with information relevant to a specific spatial or spatio-temporal entity, such as weather or contextual data.

Functions The task of interlinking aims to identify the following relations between moving objects and spatial or spatio-temporal entities:

1. Point (vessel) in polygon (area).
2. Point close (haversine distance) to polygon.
3. Point towards to polygon. In addition to the above, the vessel heading is necessary here.
4. Point close to point (points may be vessels or ports).
5. Point towards point (vessel moving towards other vessel; heading is necessary here).
6. Polygon overlap.
7. Point (vessel) close to poly-line (border, shipping lane, etc).
8. Point towards poly-line.
9. Poly-line (vessel trajectory) close to poly-line (vessel trajectory).
10. Poly-line (vessel trajectory) close to parallel poly-line (vessel trajectory). In addition to the above, the heading is necessary here.
11. Poly-line (vessel trajectory) crossing polygon (area).

Performance Requirements and Validation Criteria In terms of performance, the main requirement is that the interlinking task must take place in real-time as the stream is being processed. It is important to minimize any latency or delay imposed on the stream, due to the interlinking functionality. As such, in the context of datAcron, the integration/interlinking needs to produce an output stream of at least the same stream rate (positions per second) as the input stream of moving object positions. Therefore, the acceptable latency level is characterized as *operational*.

Another important aspect is accuracy of interlinking, i.e., which percentage of relations were really identified and led to interlinking. The primary objective here is to achieve 100% accuracy, namely to successfully interlink all positions of moving objects with any spatial or spatio-temporal entity, as long as any relation (from the list of relations mentioned above) holds. As a secondary objective, it is going to be investigated how much performance gain can be attained by relaxing the requirement for accuracy. This trade-off is particularly interesting in practice, due to the real-time constraint imposed. In several cases, it is possible that a small loss in accuracy can be tolerated, if this translates to significant performance gains.

Link to Research Objectives This requirement relates mainly to objective [O.1] "Scalable integration and management of data from disparate and heterogeneous sources" and the specific research challenge [O.1.2]: "Automatic, real-time semantic annotation and linking of data towards generating coherent views on integrated cross-streaming and archival data".

Link to Use-case Scenarios The real-time integration/interlinking task is very important in the context of datAcron, as it supports most of the use-case scenarios. Practically all scenarios are based on the links discovered between surveillance data (moving objects) with weather and contextual data, as shown in the following list:

- SC11 Collision avoidance
- SC12 Vessel in distress / Man overboard
- SC21 Monitoring maritime protected area (from illegal fishing)
- SC22 Fishing pressure on areas
- SC31 Detection of migrants / refugees and human trafficking
- SC32 Illicit activities
- FP01 Real trajectory reconstruction
- FP02 Real trajectory enrichment
- FP03 Event recognition in trajectories
- FP04 Event forecasting in trajectories
- FP05 Data set preparation
- FP06 Trajectory clustering
- FP07 Trajectory prediction - preflight
- FP08 Trajectory prediction - preflight schedule based
- FP09 Trajectory prediction - real time
- FP10 Trajectory comparison
- FM01 Regulation prediction
- FM02 Demand and capacity prediction
- FM03 Resilience assessment

Data needed from other datAcron components In general, this requirement does not explicitly need data from other datAcron components. It operates at the lowest level of the architecture, namely over the streaming data, and performs the task of interlinking in real-time. Still, it is foreseen that its operation can be parameterized based on external information coming from any datAcron component. The parameterization practically establishes an adaptive behavior of the real-time interlinking task, where the linking can be performed only on weather data, or only on contextual data, or both. This parameterization is going to be useful when evaluating specific use-case scenarios, where the focus is going to be on link discovery for some particular data only. Table 6 briefly describes how the adaptive behavior is going to be achieved, based on external input.

Data needed from other datAcron components	Data modality	Acceptable latency
From any component: a parameter indicating which source should be used in the link discovery process	Key-value pairs corresponding to data sources with boolean values	Does not apply, as these parameters are going to change on demand

Table 6: Overview of data needed from other datAcron components in order to support requirement R1.1.

4.2.3 Requirement 1.2: Interplay of in-situ and stream processing components

The in-situ processing component aims at accelerating the processing of high-rate data streams by placing the processing element close to the source of the stream. This is mostly beneficial for positioning data as created by AIS messages in the maritime use case or surveillance data from ADS-B or radar in the aviation use case. Close to the source does not necessarily imply that the processing component is physically placed on an aircraft or vessel, although those options might be possible theoretically, but at the boundary of the datAcron system where the data is made available.

Inputs The major input for the in-situ components is the stream of moving object positions (object-id, timestamp, position) as in the streaming component. Some of the tasks such as local pattern discovery and trajectory compression can merely be executed on these inputs in the in-situ components. However, most of the tasks will require further input to be provided by the streaming component, hence requiring bi-directional communication between streaming and in-situ processing components:

- *Parameters*: most trajectory based algorithms require parameterization that often depends on the application and user needs. For example, the granularity of stop detection in trajectories is parameterized via spatio-temporal parameters. Further parameters depend on the use case, as for example areas of interest in collision avoidance of fishing monitoring use cases. Often, setting the appropriate parameters requires an explorative visual analysis of the data already available. This implies that there also must be a communication path (possibly via the streaming component) back from WP4 to the in-situ components.
- *Contextual information*: One major benefit of the in-situ processing components will be that they can apply predictive models at a very early stage on the incoming data, in order to monitor and generate alarms in low-latency. Complex models will mostly be based not on moving object data only, but will also depend on contextual information such as weather data, flight plans, or cross-stream information.
- *Predictive models*: In both WP2 and WP3, predictive models will be generated that either allow generate predictions for trajectories or for events, respectively.

Outputs The in-situ components can generate different types of output:

- Preprocessed/compressed data streams. This is achieved by executing trajectory processing algorithms from WP2 in the in-situ component.
- Stream of events, by applying event and pattern detection algorithms from WP3 in the in-situ components.

- Forecasted Events: the in-situ application of predictive event models from WP3 will lead to early alarms with low-latency
- Concept drift alerts: in some cases, the in-situ components should be capable of performing model monitoring, for example, to check whether the predictions made of an event forecasting model from WP3 actually hold on the input data stream. Another example is monitoring an outlier model: if the ratio of detected outliers crosses a given threshold, the model will be considered as out-dated due to a concept drift in the environment.
- Model updates: specific learning algorithms will be developed that allow to learn models in-situ in a distributed fashion, such that the global model can be generated from the composition of local (in-situ) models. Model monitoring will allow to check whether the communication of a local model is beneficial enough for the accuracy of the global model. If so, the in-situ component will output a local model to be composed into a global model in WP3.

Functions The functions of the in-situ components basically transform the inputs to the desired outputs as described above, while relying on algorithmic components developed in WP2 and WP3.

- Stream compression
- Event detection
- Event forecasting
- Concept drift monitoring
- Model building

Performance Requirements and Validation Criteria The following validation criteria will be applied to evaluate the performance of the in-situ components.

- Reduction in size and rate of the input streams to be processed by the streaming component. The more processing can be achieved in-situ, close to the data source, the less the streaming component must process. Compression and event detection already performed in-situ will reduce the data rate that has to be processed by the streaming component, or, in other words, given that the data rate processed by the streaming components is fix, the benefit of the in-situ components can be measured as an increase in the data rate achieved by the whole datAcron system.
- Reduction in the latency in the detection of events. Event detection performed in-situ at an early stage in the processing flow can be compared against the delay incurred when performing event detection after the stream processing component.
- Reduction in latency of forecasted events: similar to the case of event detection, in-situ application of predictive models from WP3 can reduce the latency of event forecasting, e.g. providing earlier alarms in collision avoidance.
- Latency of concept drift detection and model building. In-situ learning can help to accelerate the tasks of concept drift detection and model building.
- Accuracy of concept drift detection and model building. This validation criterion compares the accuracy of these tasks in comparison with a centralized model building approach.

Link to Research Objectives The in-situ processing component is generally related to the overall research objective [O.1.1] "Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates" as it aims to accelerate the processing rate of data by processing the data already close to the data source. It will be most beneficial in particular with respect to the research objectives [O.3.1] "Real-time event recognition and forecasting algorithms that take full advantage of the data provided", and [O.3.2] "Methods for adapting event patterns in dynamic settings" as it can help to accelerate predictions and detection of patterns by online learning at a very early stage.

Link to Use-case Scenarios As in-situ processing is a preceding task to the stream processing, relevant use cases are a subset of the use cases addressed in requirement R1.1.

- SC11 Collision avoidance
- SC12 Vessel in distress / Man overboard
- SC21 Monitoring maritime protected area (from illegal fishing)
- SC22 Fishing pressure on areas
- FP01 Real trajectory reconstruction
- FP03 Event recognition in trajectories
- FP04 Event forecasting in trajectories

Data needed from other datAcron components Table 7 presents details on the data needed from other datAcron components, in order to realize the functionality of this particular requirement. The main source of data is the parameterization of the in-situ processing component based on feedback received by other datAcron components. Based on the specific use cases that need to be supported in datAcron, it is foreseen that visual data exploration and visual analytics are going to guide the parameterization, each time according to the individual use case and user needs. However, the feedback mechanism can be generic enough in order to support receiving feedback also from other datAcron components, if such a need arises during the project.

Data needed from other datAcron components	Data modality	Acceptable latency
From Visual Analytics: parameters needed for configuring operation	Alphanumeric or numeric data in key-value form	Does not apply, as these parameters are going to change on demand

Table 7: Overview of data needed from other datAcron components in order to support requirement R1.2.

4.2.4 Requirement 1.3: Integration/interlinking over stored data

This requirement targets the need of the datAcron project to integrate trajectories and events, when they have been detected, with stored data. Also the interlinking of data from archival sources with the enriched stream and other sources (e.g., flight plans). The information that is going to be linked with the trajectory or event comes from different sources of data, which are not

explicitly related to the trajectory or event data respectively. Examples of such sources of data that will be used for interlinking include weather information based on weather forecasts, and contextual information about spatial areas of interest that are related to the trajectory, information about the configuration of the airspace crossed by a trajectory, or various characterizations of points in a trajectory (e.g., at the start airport). Also, interlinking of events with relevant trajectories of moving objects is also examined in this requirement.

The objective of interlinking trajectories and events is to provide links or associations of trajectory data that consist of recordings of positions of moving objects in space and time with relevant weather or contextual data, in order to associate and enrich analysis results with useful implicit information. As an example, based on empirical evidence, it is indicated that there exists a strong correlation between the actual trajectory followed by a moving object and the weather conditions in the neighboring areas. However, the weather information is not always readily available together with analysis results, so datAcron is going to explore the potential of linking detected trajectories with such data, thereby providing insightful information for more advanced analysis tasks.

Inputs The main input is a specific trajectory of a moving object (vessel or aircraft) or a detected event. Other inputs necessary for the interlinking task include all stored data which need to be associated it with a trajectory or event. The following input sources have been identified from the use-case scenarios:

- *Weather data* in the form of forecasts for specific spatial areas and a future temporal instance or temporal interval. A potentially different set of weather-related variables of interest is deemed more useful for each domain. As an example, typical variables of interest for the aviation use-case include: wind speed, temperature, and geopotential height; whereas in the maritime use case instead of altitude, sea state (i.e., wave height and direction) and precipitation information are more relevant.
- *Contextual data* from different sources:
  - sector data (in the case of aviation), defined as geometric 3D regions that constitute a partitioning of the air space
  - areas of interest (in the case of maritime), defined as spatial 2D polygons describing protected areas, etc.
- *Regulations* (in the case of aviation), defined by appropriate ATC variables.
- *Other data sources*, namely (in the case of aviation) information related to flight plans.

Outputs The output of this requirement is going to be a trajectory or event associated with external data, weather or contextual, that affect it.

Functions Based on the spatio-temporal representation of a trajectory or event, and the spatio-temporal representations of weather or contextual data, a matching process is going to take place in order to identify relevant data for the trajectory or event at hand respectively. In most cases, the matching is going to be performed based on spatio-temporal criteria.

In the case of associating the trajectory of a flight with a flight plan, each trajectory that corresponds to a flight is going to be associated with the last available flight plan prior to take-off. In this case, the matching is going to be performed by using a unique identifier for the aircraft (ICAO and/or aircraft ID) whose trajectory is under inspection, the date and time of the flight, as well as the origin and destination.

Performance Requirements and Validation Criteria In terms of validation of this requirement, the main metric is going to be the number of trajectories and events that were linked with weather and contextual data. This number will be contrasted to the number of trajectories and events for which such external data exist, in order to quantify the completeness of the link discovery task, i.e., which percentage of trajectories (events) that could have been linked were actually linked. It is important to clarify that for the correct computation of this metric only the trajectories and events for which there exist weather and contextual data should be taken into account, as clearly it is not possible to enrich a trajectory (event) in the absence of related weather and contextual data.

In terms of performance, the task of integration/interlinking over stored data will take place in the RDF store, after the trajectory or event has been detected. As such, it is not an operation that is going to be performed on the stream of positions of moving objects. Still, the time necessary to perform the enrichment is significant, and the aim is to accomplish this task with latency level *tactical*.

Link to Research Objectives This requirement is directly relevant to objective [O.1] "Scalable integration and management of data from disparate and heterogeneous sources", since the interlinking of trajectories and events is part of the integration task. In particular, this requirement relates to the specific challenge [O.1.2] "Automatic, real-time semantic annotation and linking of data towards generating coherent views on integrated cross-streaming and archival data".

Link to Use-case Scenarios The requirement of integration/interlinking over stored data is necessary for all use-case scenarios since it is the basis for more complex analytics that need to be performed in datAcron, from prediction and forecasting to complex event detection. It should be emphasized that the specific requirement is significant also for real-time operation, such as *FP09 Trajectory prediction - real time*, because the integration over stored data is going to be exploited during real-time operations.

- SC11 Collision avoidance
- SC12 Vessel in distress / Man overboard
- SC21 Monitoring maritime protected area (from illegal fishing)
- SC22 Fishing pressure on areas
- SC31 Detection of migrants / refugees and human trafficking
- SC32 Illicit activities
- FP01 Real trajectory reconstruction
- FP02 Real trajectory enrichment
- FP03 Event recognition in trajectories
- FP04 Event forecasting in trajectories
- FP05 Data set preparation
- FP06 Trajectory clustering

- FP07 Trajectory prediction - preflight
- FP08 Trajectory prediction - preflight schedule based
- FP09 Trajectory prediction - real time
- FP10 Trajectory comparison
- FM01 Regulation prediction
- FM02 Demand and capacity prediction
- FM03 Resilience assessment

Data needed from other datAcron components	Data modality	Acceptable latency
From WP2: Detected and predicted trajectories	The representation of a trajectory	Tactical
From WP2: Detected clusters and patterns	The representation of a cluster or pattern	Tactical
From WP3: Detected and forecasted events	The representation of an event	Tactical

Table 8: Overview of data needed from other datAcron components in order to support requirement R1.3.

Data needed from other datAcron components Table 8 describes the data required from other datAcron components. For the task of integration over stored data, it is necessary to receive input from WP2 and WP3 about trajectories and events respectively, whenever they have been detected or predicted. This data is going to be integrated with stored information as already described above. The latency level is *tactical*, indicating the time interval in which a detected trajectory or event must be fetched. In terms of data modality, the representation of each detected/predicted object is going to be used, and this representation is going to be specified by the respective work package.

4.2.5 Requirement 1.4: Spatio-temporal RDF querying of integrated data

This requirement refers to the need for scalable querying of spatio-temporal RDF data from the distributed RDF store, based on filtering criteria that include spatial, temporal and other RDF predicates.

Inputs The spatial and temporal constraints are represented as ranges of spatial dimensions and a temporal interval respectively. Optionally, RDF predicates are given as input, aiming at retrieval only of those RDF data that satisfy the predicates. Also, the information (properties) that should be retrieved within a specific area for some period of time, and complies with the input constraints (spatio-temporal and other predicates). As an indicative example, this information could be: vessel IDs, vessel type, and vessel flag, within a specific area for some period of time. In addition, this requirement also covers the need for querying RDF data using other predicates, apart from spatio-temporal filtering criteria. As an example, vessel IDs of a specific vessel type located within a spatio-temporal constraint could be requested.

Outputs The entities that comply with the input constraints, along with requested fields of information.

Functions The system shall first perform validity checks on the given input constraints. These validity checks shall verify the correctness of the constraints; for example, in case of a spatio-temporal box $[x_1, x_2] \times [y_1, y_2] \times [t_1, t_2]$ defined on dimensions X, Y, and T, it should always hold that: $x_1 \leq x_2$, $y_1 \leq y_2$, and $t_1 \leq t_2$.

Only in case the validity checks are successful, the system continues with processing the request in the distributed RDF store, otherwise no processing is performed. During the actual processing, efficient access methods will be employed to selectively access only part of the underlying data, while eagerly pruning unnecessary data to the result. The exact access methods will be defined in the design phase, however it is foreseen that both spatio-temporal indexing and RDF access methods will be in place to support efficient distributed access. The resulting objects that satisfy the given input criteria will also be accompanied with all fields required, before being returned as results.

Performance Requirements and Validation Criteria Efficiency and scalability are the main performance requirements with respect to the size of the integrated RDF data.

Efficiency will be quantified as the percentage of accessed data over the total amount of data in the distributed RDF store. This also relates to the amount of pruning achieved, namely what is the proportion of data accessed for producing the result.

Scalability will be related either to (a) the size of stored data, or (b) the number of physical nodes available. In the ideal case, a linear behavior is expected to be observed as the size of integrated RDF data grows, given a fixed hardware setup.

In terms of processing time, the exact processing time relates to the amount of stored RDF data and the available hardware. Still, the target is to deliver both an efficient and scalable solution.

Link to Research Objectives This requirement is directly related to research objective [O.1] "Scalable integration and management of data from disparate and heterogeneous sources". In particular, the focus of this requirement is on the management of data, after it has been integrated from different sources. A key issue is scalability, since it is expected that the volume of the integrated data is going to be high, far surpassing the capabilities of a centralized approach and calling for distributed solutions. As a result, one main challenge is to support efficient querying of spatio-temporal RDF data over a distributed RDF store. Thus, the present requirement relates to research objective [O.1.1] "Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates", in terms of producing a scalable processing framework. Another key issue is the complexity of the underlying data, which include numerical data, textual data, as well as spatio-temporal data. Efficient querying of data of such variety is also challenging and will be addressed as part of the research challenge [O.1.3] "Efficient distributed management and querying of integrated spatio-temporal data".

Link to Use-case Scenarios The specific requirement stems from the following use-case scenarios that require access to integrated spatio-temporal data:

- SC11 Collision avoidance
- SC12 Vessel in distress / Man overboard

- SC21 Monitoring maritime protected area (from illegal fishing)
- SC22 Fishing pressure on areas
- SC31 Detection of migrants / refugees and human trafficking
- SC32 Illicit activities
- FP01 Real trajectory reconstruction
- FP02 Real trajectory enrichment
- FP03 Event recognition in trajectories
- FP04 Event forecasting in trajectories
- FP05 Data set preparation
- FP06 Trajectory clustering
- FP07 Trajectory prediction - preflight
- FP08 Trajectory prediction - preflight schedule based
- FP09 Trajectory prediction - real time
- FP10 Trajectory comparison
- FM01 Regulation prediction
- FM02 Demand and capacity prediction
- FM03 Resilience assessment

Data needed from other datAcron components Table 9 describes the data required from other datAcron components. In principle, the task of querying spatio-temporal RDF data is independent of other datAcron components, in the sense that any data that have already been integrated (regardless from their original source) are going to be available for querying. However, for supporting spatio-temporal querying of trajectories and events, it is necessary to receive input from WP2 and WP3 about trajectories and events respectively, which will be subsequently integrated, as described in requirement R1.3. The latency level is *tactical*, indicating the time interval in which a detected trajectory or event must be fetched. In terms of data modality, the representation of each detected/predicted object is going to be used, and this representation is going to be specified by the respective work package.

4.2.6 Requirement 1.5: Retrieval of spatio-temporally constrained subsets of integrated data

This requirement covers the need to obtain integrated views (subsets) of the data stored in the datAcron store, where each subset is defined by a spatio-temporal constraint. Emphasis should be given to the fact that the subset of retrieved data is *integrated*, and also that it is interlinked with data coming from disparate sources. More concretely, the retrieved subset of data corresponds to trajectories of moving objects, where a trajectory (or parts of the constituents of a trajectory) has been enriched with data from other sources, such as weather, protected areas, sectors, configurations, and flight plans. The obtained subset of integrated data refers to

Data needed from other datAcron components	Data modality	Acceptable latency
From WP2: Detected and predicted trajectories	The representation of a trajectory	Tactical
From WP2: Detected clusters and patterns	The representation of a cluster or pattern	Tactical
From WP3: Detected and forecasted events	The representation of an event	Tactical

Table 9: Overview of data needed from other datAcron components in order to support requirement R1.4.

a specific spatial area and is temporally-constrained, thereby providing a very informative view of any situation in different space/time granularities that can be used for different purposes in the datAcron project.

Inputs The spatial and temporal constraints represented as a spatio-temporal box. For the maritime use-case, this input constraint can be represented as: $\{[x_1^i, x_2^i] \times [y_1^i, y_2^i] \times [t_1^i, t_2^i]\}$, where the letter "i" in the exponent is used to discriminate this range of values as input constraint. For the aviation use-case, the input constraint also includes the fourth dimension: $\{[x_1^i, x_2^i] \times [y_1^i, y_2^i] \times [z_1^i, z_2^i] \times [t_1^i, t_2^i]\}$ In the following, we focus in the 3D case (x, y, t) merely for simplifying the notation, as the generalization to 4D (x, y, z, t) is straightforward.

Outputs A subset of the integrated data stored in datAcron, represented as an RDF graph and encoded in the form of RDF triples, which the most typical storage format used for RDF. All data will be provided in a form that is independent of the internal representation of each WP and will be specified as part of the datAcron integrated system architecture in D1.2.

Functions Given the input spatio-temporal constraints, the complete RDF graph used to store the integrated data in datAcron is filtered in order to keep only those spatio-temporal resources that comply with the input constraints. Compliance with the input constraints is defined more accurately in the following.

- First, assume the case of spatio-temporal objects, i.e., objects that are directly associated with spatio-temporal data. For instance, this category includes positions of moving objects, trajectories, weather forecasts, sector configurations, etc. These objects can be further classified in two categories, based on whether their spatio-temporal information is an exact position in space and time ($\{x, y, t\}$) or interval-based ($[x_1, x_2] \times [y_1, y_2] \times [t_1, t_2]$).

  If the position is an exact position, it is straightforward to determine whether the object complies with the input constraint and should be returned, it suffices to check if the following inequalities hold: $x_1^i \leq x \leq x_2^i$ and $y_1^i \leq y \leq y_2^i$ and $t_1^i \leq t \leq t_2^i$.

  If the position is an interval, then returned objects must either be entirely contained in the input constraint or have an overlap with the input constraint. As an example, the part of a trajectory that has a spatio-temporal overlap with the input constraint is retrieved as result, even though the trajectory itself is not entirely contained in the constraint.

- Next, the case of RDF data which are not directly related to spatio-temporal information is considered. Examples of such data include types of vessels and aircrafts, their identifiers, as well as their characteristics. This data must also be returned in the retrieved subset of data, given that there are paths in the RDF graph connecting these data to the spatio-temporal objects in the subset.

Performance Requirements and Validation Criteria The main validation criterion for the present requirement is the correctness and completeness of the retrieved subset of integrated data. Namely, that (a) no spatio-temporal object that satisfies the spatio-temporal input constraint is missed (false negative), and that (b) no spatio-temporal object that does not satisfy the spatio-temporal input constraint is returned (false positive).

In terms of performance, the main measure is going to be the time required (response time) to produce the subset of integrated data from the complete integrated set of data. It is expected that the time required is going to be dependent on the size of the complete integrated set of data. Ideally, an at most linear increase in the response time with the size of complete dataset is desired.

Link to Research Objectives The present requirement relates to objective [O.1] "Scalable integration and management of data from disparate and heterogeneous sources". It is an indicator of the ability of datAcron to create integrated views of data coming from streaming and archival sources, which are also spatially and temporally focused in specific spatial areas and temporal intervals respectively.

Link to Use-case Scenarios This requirement originates explicitly from use-case scenario *FP05 Data set preparation*, as described in the aviation use-case. However, it is expected that other use-cases will also benefit from it. For example, certain analysis tasks are expected to require access to a spatio-temporally constrained snapshot of the integrated data, in order to perform the associated task, e.g., model building, clustering, forecasting, etc.

Data needed from other datAcron components	**Data modality**	**Acceptable latency**
From WP2: Detected and predicted trajectories	The representation of a trajectory	Does not apply, whenever a trajectory is detected/predicted
From WP2: Detected clusters and patterns	The representation of a cluster or pattern	Does not apply, whenever a cluster/pattern is detected
From WP3: Detected and forecasted events	The representation of an event	Does not apply, whenever an event is detected/forecasted

Table 10: Overview of data needed from other datAcron components in order to support requirement R1.5.

Data needed from other datAcron components Table 10 describes the data needed from other datAcron components. This case is identical to requirement R1.4, therefore it is not

analyzed in further detail.

4.3 Requirements for Trajectories Detection and Forecasting

A key operation in datAcron is the management of trajectory data, in particular trajectory detection and forecasting. Trajectory detection is a task that aims to construct trajectories from raw positioning information. During this process, many challenges need to be confronted, including the computation of synopses from surveillance data, as well as the reconstruction of trajectories based on these synopses. Trajectory forecasting is a complex task that includes both short-term prediction and long-term prediction, based on the targeted use-case. To successfully accomplish the forecasting task, various constituent processing components must be built, including pattern detection and discovery, trajectory clustering, as well as building predictive models by machine learning algorithms.

Based on the above, the following list of individual requirements are relevant to WP2:

- R2.1: Computation of trajectory similarity and clustering
- R2.2: Pattern discovery
- R2.3: Prediction of trajectories and locations
- R2.4: Computation of surveillance data synopses, reconstruction of trajectories by data synopses

4.3.1 Requirements for the integrated datAcron system

Based on the functional requirements, which are analyzed in detail in the subsequent subsections, we record a set of architectural requirements for the integrated datAcron system.

Inputs WP2 receives input from the following types of main data sources[3]:

- *surveillance data*:
  - maritime use-case: AIS data from coastal receivers (MS1) and satellite receivers (MS2)
  - aviation use-case: radar tracks (AS1) and ADS-B messages (AS2)
- *weather data*: in particular weather forecasts, both related to the conditions at sea, as well as operational weather data at ground-air level and at airports:
  - maritime use-case: METOC data from different sources (MW1)
  - aviation use-case: Metar data (AW1)
- *contextual data*:
  - maritime use-case: EU regulated fishing areas (MC1), FAO and ICES fishery statistical areas (MC2), Community Fishing register (MC3), European Marine Observation and Data Network (MC4), Marine protected areas NATURA 2000 (MC5), Environmental biodiversity datasets (MC6), World Port Index (MC7), and Routes (MC8)

[3]The notation introduced in Section 2.3 (Tables 3 and 4) is used to identify data sources.

  - aviation use-case: Sector Configuration (AC1), Airports database (AC2), Aircrafts database (AC3)

- *other data sources*: Airline schedule (AO1), Flight plans (AO2)
- *regulations*: Flow Management Data (AR1)
- *synthetic trajectories*: generated based on existing model-based prediction algorithms used in the aviation domain
- *events* recognized, predicted or reported:
  - maritime use-case: events (ME1) and maritime events reports (MRE1)
  - aviation use-case: events (AE1)

In addition, WP2 receives input from other WPs according to the following descriptions:

- *WP1's interlinked data*: WP2 uses the batch processing tasks of WP1 over the data in the integrated store, thus retrieving integrated data based on different criteria, including spatio-temporal constraints, information about moving objects (e.g., types of moving objects, their characteristics, etc.), weather forecasts, specific configurations, sectors, regulations, flights and flight plans, etc.
- *WP1's enriched stream of surveillance data*: WP2 is recipient of this stream that includes enriched positional information (moving object positions) from surveillance data sources.
- *WP3's analysis results*: WP2 receives the initially detected events as well as their interlinked version after they have been processed by the integration module of WP1.

A graphical illustration of inputs to WP2 from other datAcron components is depicted in Figure 5.

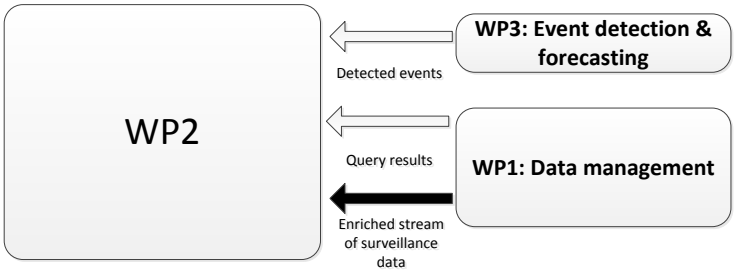

Figure 5: Inputs to trajectories detection and forecasting component.

Outputs WP2 provides output within the integrated datAcron system as follows:

- *Synopses*: The first output of WP2 is a derived, reduced (compressed) stream of critical points from the stream of surveillance data that results to semantic trajectories, progressively becoming data-at-rest that can be used for offline processing, mining, etc.
- *Predictions*: The second output of WP2 consists of detected and forecasted locations and trajectories of moving objects.

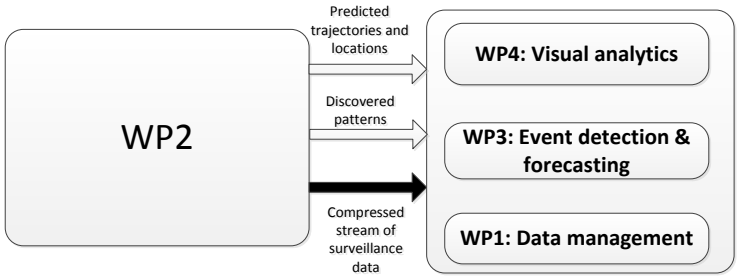

Figure 6: Outputs of trajectories detection and forecasting component.

- *Patterns*: The third output of WP2 comprises of all kind of analysis results, such as clusters (of points and/or trajectories), sequential patterns, etc. Furthermore, the involved objects (i.e. the objects that conform the cluster or satisfy the sequential pattern) are returned, while these patterns are forwarded to other WPs so as to either readily used or interlinked to already existing entities

A graphical illustration of all the above outputs of WP2 is depicted in Figure 6.

4.3.2 Requirement 2.1: Computation of trajectory similarity and clustering

This requirement includes operations that need to be performed on either raw or semantically enriched trajectories in the context of datAcron. More specifically, the goal of this requirement is twofold: First, given a pair of (semantic) trajectories the goal is to quantify their similarity score. This operation should account both the spatio-temporal properties of the trajectories but also their semantics. Based on such similarity score, the second goal is to be able to cluster (semantic) trajectories, namely to detect groups of (semantic) trajectories which comply to certain events at specific places, time speed or altitude.

Inputs

- A pair or a set of (semantic) trajectories depending on whether we want to detect groups or simply measure the similarity score between two specific trajectories.
- Constraints in terms of events, area, time, speed, altitude, etc.

Outputs

- A similarity score, and (optional) a structure of their matching components
- Groups of semantic trajectories

Functions

- The algorithm that measures the similarity between two (semantic) trajectories should handle the two aircrafts or vessels as two sequences, identify their matching components and accumulate their individual scores.

- The clustering algorithm first filters the semantic trajectories w.r.t. the given constraints and subsequently utilizes the afore-mentioned similarity function to cluster semantic trajectories w.r.t. the various dimensions included in the constraints' list.

Performance Requirements and Validation Criteria Regarding the trajectory similarity the latency should be at the *tactical* level, while the clustering function at the *strategic* level. Both functions will eventually be evaluated by either internal and/or external clustering evaluation metrics depending on the availability of ground truth in the available datasets

Link to Research Objectives This requirement relates to [O.2.2] "Data analytics over the trajectories of moving entities", since both the computation of similarities between trajectories as well as the trajectory clustering task are basic structural units for more advanced data analytics over moving object data.

Link to Use-case Scenarios The computation of similarities between trajectories directly relates to use-case *FP10 Trajectory comparison*, but also to use-case *FP06 Trajectory clustering*, since most clustering algorithms are based on some similarity measure. Obviously, trajectory clustering is also at the heart of use-case scenario *FP06*. Last but not least, this requirement is relevant for event recognition in trajectories (use-case scenario *FP03 Event recognition in trajectories*), by identifying groups of trajectories with similar behavior.

In a nutshell, the following use-cases are relevant to this requirement:

- FP03 Event recognition in trajectories
- FP06 Trajectory clustering
- FP10 Trajectory comparison

Data needed from other datAcron components Table 11 describes the necessary data from other WPs, in order to perform the required tasks. Both the trajectory similarity computation as well as the trajectory clustering need trajectory data from the data management component (WP1). This data is integrated and interlinked with weather and contextual information, thereby providing trajectories with rich semantics to WP2, which can be exploited to improve the quality or effectiveness of analytics over trajectory data. Moreover, events detected by WP3 (and their interlinked version) are also needed, as they can form valuable sources for the relevant operations with this requirement.

Data needed from other datAcron components	**Data modality**	**Acceptable latency**
From WP1: retrieval of trajectories enriched with weather and contextual data in order to compute similarities	Integrated RDF data	Tactical
From WP3: detected events and their interlinked versions	The representation of an event	Tactical

Table 11: Overview of data needed from other datAcron components in order to support requirement R2.1.

4.3.3 Requirement 2.2: Pattern discovery

This requirement addresses the needs for pattern discovery in datAcron, such as route patterns, "hot-spot" spatial areas, and fishing areas.

- Route patterns: The goal of this requirement is to detect 'routes', namely the paths whereupon the movements of all or specific types of vessels (e.g., fisheries) take place, so as the latter to be monitored (i.e. when they enter/exit the routes, or whether they exhibit certain behavior).
- "Hot-spot" spatial areas: To identify spatial areas where a high number of fishing vessels are located frequently.
- Fishing areas: To discover the spatio-temporal sequential patterns where the fishing activity takes place.

Inputs

- The semantic trajectories for which we want to detect their corresponding "route patterns".
- 1) Spatio-temporal constraint, 2) Fishing vessel positions within the constraint.
- 1) Spatio-temporal constraint, 2)"Hot-spot" spatial areas, 3) Fishing vessel trajectories within the constraint.

Outputs

- The frequent paths (i.e. sequences of points or areas) wherefrom the given set of trajectories pass
- Spatial areas, identified as "hot-spot" areas
- Spatio-temporal sequential patterns

Functions

- Devise suitable clustering algorithms to detect frequent 'hot spot' points or areas shared by subsets of trajectories that therefore constitute the waypoints defining a corresponding set of semantic trajectories; subsequently, apply appropriate sequential pattern mining algorithms to detect common sequences of movement episodes linking these frequent points or areas.
- Retrieve all fishing vessel positions that comply with the spatio-temporal constraint and apply clustering algorithms in order to detect spatial areas of high vessel density (which indicate that this is a fishing area). The clustering algorithm will produce groups of points (vessels), located in nearby locations. Each such group represents a candidate fishing area. To output fishing areas, each group of vessels will be represented as a polygon defined as the convex hull of the respective set of points.
- Given the "hot-spot' spatial areas, the goal is to apply a sequential pattern mining algorithm so as to discover frequent sequential patterns of the involved trajectories.

Performance Requirements and Validation Criteria All requirements in the current category should be available with latency at the *strategic* level, since their functions imply iterative operations with possible multiple passes from the involved data objects. Similarly with the clustering requirements, the current functions will be evaluated by either internal and/or external clustering evaluation metrics depending on the availability of ground truth in the available datasets, as well as by the maximization of the coverage of the detected patterns.

Link to Research Objectives This requirement relates to [O.2.2] "Data analytics over the trajectories of moving entities", which includes pattern discovery from trajectory data.

Link to Use-case Scenarios This requirement is expected to support the use-case scenarios related to *SC21 Protection of ecological areas* and *SC22 Fishing pressure*. Both use-cases are dependent on the ability to effectively discover patterns of moving objects and their trajectories.

Data needed from other datAcron components All pattern discovery tasks prescribed in this requirement require input data from WP1, either positions of fishing vessels or trajectories of fishing vessels, which satisfy a given user-defined spatio-temporal constraint. Also, events detected by WP3 are also necessary, since they may lead to the discovery of patterns that would not have been discovered otherwise. The retrieved data is going to be used by pattern detection algorithms, in order to produce the required patterns according to the use-case at hand. The acceptable latency level for data retrieval from WP1 and WP3 is *tactical*, since the pattern discovery task itself is expected to be long-running, thus it is labeled as *strategic*. Due to the *strategic* latency of the pattern discovery task, higher latency from WP1 and WP3 can be tolerated in this requirement, in case the amount of retrieved data (determined by the spatio-temporal constraint) is high. Table 12 summarizes the data needed from other work packages.

Data needed from other datAcron components	Data modality	Acceptable latency
From WP1: fishing vessel positions and/or fishing vessel trajectories satisfying a spatio-temporal constraint	Integrated RDF data	Tactical
From WP3: detected events and their interlinked versions	The representation of an event	Tactical

Table 12: Overview of data needed from other datAcron components in order to support requirement R2.2.

4.3.4 Requirement 2.3: Prediction of trajectories and locations

This particular requirement refers to the need for prediction of future locations and trajectories of moving objects.

With respect to prediction of the position (location) of a moving object in the future, despite the generic nature of the requirement for location prediction, specific instantiations of the problem are going to be studied in the context of datAcron, namely the prediction of:

- which vessels (such as cargos, tankers, ferries) will cross the areas where fishing vessels are fishing

- the position of suspicious vessels to optimize helicopter and coast guards intervention
- similarly to the maritime domain, in the aviation domain we want to predict the future position of aircrafts which is a prerequisite for event forecasting.

With respect to trajectory prediction, the following cases are distinguished:

- In case of a collision and escape of the responsible vessel, the user wants to predict the trajectory of the fugitive.
- For a given flight plan a forecasted trajectory will be obtained and compared with the real one finally flown.
- For a given airline schedule a forecasted trajectory will be obtained and compared with the real one finally flown.
- For a given flight plan and the current surveillance data arriving to the platform a forecasted trajectory will be obtained and updated continuously.

Based on the use-case scenarios, for the trajectory prediction task, two separate cases must be considered: (a) pre-flight prediction that can be performed before the actual flight takes place, and (b) real-time prediction that is performed continuously as the moving object (aircraft) changes its position.

Inputs A set of inputs is defined in order to perform the required location prediction task:

- A moving object (or a set of moving objects) along with its current position.
- The prediction time (temporal window from "now").
- Historical patterns (optional).

Moreover, for specific location prediction tasks, additional inputs are required. To determine vessels crossing a fishing area, it is necessary to have as additional input the fishing areas themselves.

For trajectory prediction:

- Vessel ID or flight plan or airline schedule
- Current movement information
- Weather forecasts
- Prediction time (temporal window from "now")
- Historic patterns of moving object (vessel or aircraft), including real historic (semantic) trajectories and corresponding (historic) flight plans

Outputs For the location prediction task, a moving object (or a set of moving objects) and its predicted future locations. For the trajectory prediction task, the predicted future trajectory of a moving object (vessel or aircraft).

Functions The algorithm uses the current movement information of the moving object (vessel or aircraft), which is defined as its position, speed, and heading. Optionally, historic patterns of the particular moving object or of other moving objects can be exploited, when present. The result of the algorithm is the predicted future location (or trajectory) at the prediction time.

The algorithm uses the flight plan of the aircraft, the weather forecast and its historic data and/or behavior and predicts its future trajectory.

The algorithm uses the airline schedule of the aircraft, the weather forecast and its historic data and/or behavior and predicts its future trajectory.

The algorithm uses the continuously updated aircraft's state and the dynamically changing flight plan, the weather forecasts and its historic data and/or behavior and continuously predicts its future trajectory from the current point till the arrival.

Performance Requirements and Validation Criteria The accuracy of the prediction in terms of the Sum of Square Errors (SSE) between the real location (trajectory) and the predicted one. The acceptable latency level is defined as *tactical*.

Link to Research Objectives This requirement is directly related to research challenge [O.2.3] "Short- and long-term real-time forecasting of trajectories".

Link to Use-case Scenarios The following use-case scenarios are directly or indirectly related to the prediction task.

- SC11 Collision prevention
- SC12 Vessel in distress / MOB
- SC31 Migrants/ Human trafficking
- FP04 Event forecasting in trajectories
- FP07 Trajectory prediction - preflight
- FP08 Trajectory prediction - preflight schedule based
- FP09 Trajectory prediction - real-time

Data needed from other datAcron components Table 13 describes the data necessary for realizing this requirement. Two cases can be distinguished: real-time prediction and offline prediction. In the real-time case, it is necessary to have access to the stream of surveillance data (i.e., moving object positions), and this access needs to be performed at the level of *operational* latency. Also, the detected events from WP3 need to be accessed for realizing this requirement also at the level of *operational* latency. In the offline case, access to integrated trajectories is needed, enriched with weather, contextual and other data sources, as well as access to any historic patterns available.

4.3.5 Requirement 2.4: Computation of surveillance data synopses, reconstruction of trajectories by data synopses

This requirement addresses the need to support the following functionality:

Data needed from other datAcron components	Data modality	Acceptable latency
From WP1: stream of surveillance data	Compressed streaming spatio-temporal data	Operational
From WP1: historic patterns, as well as trajectories enriched with weather, contextual, and other data	Integrated RDF data	Tactical
From WP3: detected events	The representation of an event	Operational

Table 13: Overview of data needed from other datAcron components in order to support requirement R2.3.

- *Reconstructing trajectories from surveillance data* will provide representation in time of the original (raw) positions: 3D (x, y, t) for vessels, and 4D (x, y, z, t) for aircrafts. In effect, distinct sequences of timestamped positions per moving object will be obtained, after excluding any inherent noise detected in the streaming positions due to e.g., delayed arrival of messages, duplicate messages, sea drift, discrepancies in GPS measurements, etc.
- *Maintaining trajectory synopses from surveillance data* will offer representation in time of summarized positions of each moving object (3D for vessels, 4D for aircrafts). In effect, this will accept positional stream(s) and will track major changes along each object's movement. Given that vessels and aircrafts normally follow planned routes (except for adverse weather conditions, congestion situations, accidents, etc.), this process will instantly identify events as "critical points" along each trajectory strictly characterized from mobility features, such as stop, turn, climb, or descent. Therefore, an approximate, summarized trajectory (synopsis) may be maintained consisting of critical points only, effectively discarding redundant locations along a "normal" course.
- *Communication gaps*: are identified that may indicate distress situations. For example, one such indicator of distress in the maritime use-case is when AIS is switched off.

Inputs In trajectory detection and summarization, data from the following main sources may be used:

- raw streams of surveillance data concerning moving object positions for the maritime use-case (AIS) and the aviation use-case (ADS-B and/or radar tracks)
- static databases of maritime information (vessels, ports)
- static database of aviation information (aircrafts, airports).
- The following information is needed as input, in order to identify communication gaps:
  - Vessel ID
  - Historical trajectory
  - Elapsed time since last AIS signal
  - Parameter: A temporal threshold τ (a time interval) for issuing a communication gap, namely, if no signal is received for more than τ then this is considered a gap

Outputs

- For trajectory reconstruction: the updated trajectory
- For maintenance of trajectory synopses: a derived stream of critical points detected for each moving object
- For communication gaps: a notification that a vessel with the given vessel ID has no known position at open sea, because contact is lost

Functions

- *Trajectory constructor*: This will provide a sequence of time-ordered points that reconstruct the trajectory per sailing vessel or flying aircraft. The output may also include calculated spatio-temporal features, such as instantaneous speed and heading, traveled time, distance, etc. as computed from the incoming positions. These estimates should not to be confused with similar original measurements emitted by the moving objects; but such derived estimates may be compared with original measurements, if needed in a particular use case.
- *Trajectory compressor*: For the task of maintaining trajectory synopsis, summarized trajectories will be produced (as a derived stream). Each such trajectory synopsis with contain detected critical points based on mobility features characterizing the observed course of a vessel (stop, turn, communication gap, speed change, slow motion, etc.) or an aircraft (e.g., Top of Climb, Top of Descent, Step Climb, Cruise Speed change, etc.).
- The trajectory synopsis should include the last known position as a sign of communication gap.

Performance Requirements and Validation Criteria For the *trajectory reconstruction* task: percentage (%) of raw messages unassigned to trajectories (classified as noise).

For the *maintenance of trajectory synopses*:

- *Compression ratio*: This is the percentage (%) of positions retained in the approximate trajectory synopsis over the raw ones originally obtained. The higher this ratio, the more compressed and lightweight the resulting synopses.
- *RMSE*: Root Mean Square Error can be used to measure approximation quality, effectively comparing the original reconstructed trajectory with its approximate representation in the synopsis. The less this error (expressed in distance units, like meters), the lower the information loss due to summarization (hence, the higher the fidelity of the resulting synopsis).

The acceptable latency level for both tasks should be *operational*, so as to provide trajectories and their synopses in near real-time, keeping up with the arrival rate of the incoming positional stream(s).

Link to Research Objectives This requirement mainly addresses objective [O.2.1]: "Cross-streaming, real-time detection of the trajectories of moving entities". However, maintenance of trajectory synopses is also related to [O.1.1]: "Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates", as explicitly mentioned in its last subtask.

Link to Use-case Scenarios This requirement originates explicitly from use-case scenario *FP01 Real trajectory reconstruction*, as described in the aviation use-cases. However, it is expected that other (maritime or aviation) use-cases will also benefit from it. For example, certain analysis tasks are expected to require reconstructed trajectories or lightweight synopses in order to perform the associated task, e.g., collision avoidance, vessel in distress / MOB, real trajectory enrichment, event recognition in trajectories, etc.

Data needed from other datAcron components Table 14 describes the data needed form other datAcron work packages for this requirement. First, access to the raw streaming surveillance data is necessary, in order to compute the synopsis. Second, access to static data relevant to moving objects (vessels and aircrafts) is needed for the trajectory reconstruction task. In addition, other static data necessary are those about ports and aircrafts, as they are indicators of the start/end of a trajectory. All this data must be available at *operational* latency level, in order to avoid any delays on the trajectory compression operation.

Data needed from other datAcron components	Data modality	Acceptable latency
From WP1: stream of surveillance data	Streaming spatio-temporal data	Operational
From WP1: mostly static data related to vessels, ports, aircrafts, and airports	Archival data	Operational

Table 14: Overview of data needed from other datAcron components in order to support requirement R2.4.

4.4 Requirements for Complex Event Recognition and Forecasting

In datAcron, adaptive complex event recognition and forecasting technology is going to be developed that is able to benefit from integrated data-in-motion and data-at-rest from multiple, disparate, voluminous data sources. To achieve this objective datAcron is going design methods for real-time event recognition under uncertainty, with noisy and fluctuating data. With respect to event forecasting, datAcron is going to predict meaningful events for the maritime and aviation use-case scenarios that will occur in the future.

The list of requirements relevant to WP3 include two main requirements, each focusing on one of the application domains of datAcron (maritime and aviation), and aiming at capturing the needs for complex event detection and forecasting in each domain:

- R3.1: Event detection and forecasting in the maritime domain
- R3.2: Event detection and forecasting in the aviation domain

4.4.1 Requirements for the integrated datAcron system

Based on these functional requirements, which are analyzed in detail in the subsequent subsections, we record a set of architectural requirements for the integrated datAcron system.

Inputs WP3 receives input from the following types of main data sources[4]:

- *surveillance data*:
  - maritime use-case: AIS data from coastal receivers (MS1) and satellite receivers (MS2)
  - aviation use-case: radar tracks (AS1) and ADS-B messages (AS2)
- *weather data*: in particular weather forecasts, both related to the conditions at sea, as well as operational weather data at ground-air level and at airports:
  - maritime use-case: METOC data from different sources (MW1)
  - aviation use-case: Metar data (AW1)
- *contextual data*:
  - maritime use-case: EU regulated fishing areas (MC1), FAO and ICES fishery statistical areas (MC2), Community Fishing register (MC3), European Marine Observation and Data Network (MC4), Marine protected areas NATURA 2000 (MC5), Environmental biodiversity datasets (MC6), World Port Index (MC7), and Routes (MC8)
  - aviation use-case: Sector Configuration (AC1), Airports database (AC2), Aircrafts database (AC3)
- *other data sources*: Airline schedule (AO1), Flight plans (AO2)
- *regulations*: Flow Management Data (AR1)
- *synthetic trajectories*: generated based on existing model-based prediction algorithms used in the aviation domain
- *events* reported:
  - maritime use-case: maritime events reports (MRE1)

In addition, WP3 receives input from other WPs according to the following descriptions:

- *WP1's interlinked data*: WP3 uses the batch processing tasks of WP1 over the data in the integrated store, thus retrieving integrated data based on different criteria, including spatio-temporal constraints, information about moving objects (e.g., types of moving objects, their characteristics, etc.), weather forecasts, specific configurations, sectors, regulations, flights and flight plans, etc.
- *WP1's enriched stream of surveillance data*: WP3 is recipient of this stream that includes enriched positional information (moving object positions) from surveillance data sources.
- *WP2's synopses*: WP3 uses the summarized trajectories produced by WP2 in order to detect complex events.

A graphical illustration of inputs to WP3 from other datAcron components is depicted in Figure 7.

[4]The notation introduced in Section 2.3 (Tables 3 and 4) is used to identify data sources.

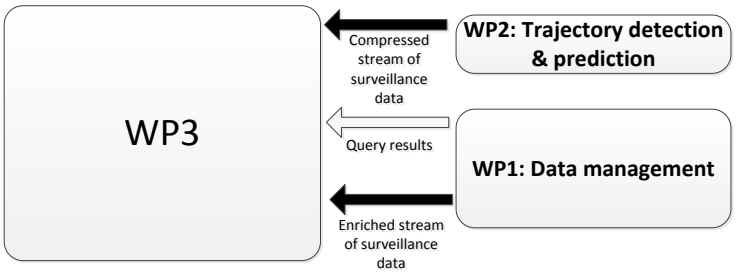

Figure 7: Inputs to complex event recognition and forecasting component.

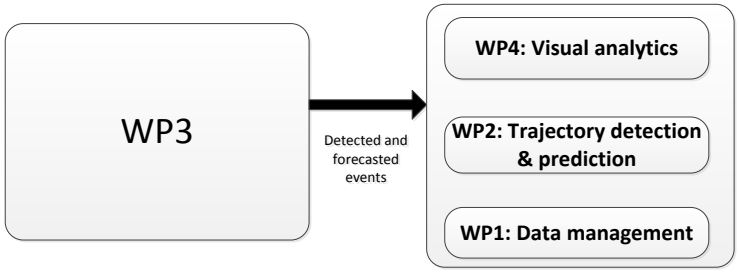

Figure 8: Outputs of complex event recognition and forecasting component.

Outputs WP3 provides output within the integrated datAcron system as follows:

- *Complex Events:* The main output of WP3 is a stream of complex events whose detection is based on a knowledge base of rules that combine compressed surveillance data (e.g., trajectory synopses), weather data, contextual data (e.g., protected zones, sector configurations), and other data (e.g., flight plans)
- *Forecasted Complex Events:* WP3 will also produce streams of complex events that are predicted to happen in the future.

A graphical illustration of all the above outputs of WP3 is depicted in Figure 8.

4.4.2 Requirement 3.1: Event detection and forecasting in the maritime domain

This requirement targets the need to identify events of interest with respect to areas, vessels and trajectories in the maritime domain. With respect to vessels, the goal is to identify a vessel's characteristics, such as its type or whether it has been black-listed. Such vessel characteristics are provided mostly as static information. If enough data (including ground truth) is available, then some of these characteristics may be inferred. With respect to areas, the goal is to identify areas with high fishing pressure. In order to achieve this, we first need to characterize the trajectories of vessels moving in such areas, e.g. determining whether a vessel is actively fishing. In order to evaluate area and trajectory characterization, the availability of ground truth is required.

Inputs The following pieces of information are considered as input to the present requirement:

- Vessel characteristics (such as type and size)
- Vessel "history" (e.g., whether it has been black-listed)
- Trajectory synopses (and possibly raw surveillance data)
- Protected areas
- Fishing areas
- Suspicious areas
- Regulations about areas (e.g., when fishing is permitted, how much pressure an area can sustain)
- List of known trafficking routes and areas (polylines/polygons)
- Weather data

Outputs

- Vessel moving and fishing in protected area
- Vessel moving towards protected area
- Vessel moving along known route
- Areas with high fishing pressure

Functions

- For the identification of vessels moving towards areas, the algorithm should try to estimate a vessel's future position and determine whether this future position lies inside an area that is "active" (other fishing vessels in it)
- For the identification of vessels fishing in protected areas, the algorithm should determine whether a vessel is inside such an area and then determine if its movement indicates a fishing activity
- For the identification of vessels moving along a route, the algorithm should compare a vessel's trajectory with the route
- For the identification of high pressure fishing areas, the algorithm should "cluster" nearby vessels simultaneously fishing, determine their pressure on the area and check if it exceeds given limits.

Performance Requirements and Validation Criteria If ground truth is provided, the F1 score ($2 \times (Precision \times Recall) \div (Precision + Recall)$) should act as a validation metric. For simple tasks, like determining whether a vessel is inside an area, the expected latency is at the *tactical* level. For the more complex tasks, such as estimating whether an area is under heavy fishing pressure, the expected latency is at the *strategic* level.

Link to Research Objectives This requirement relates to the following research challenges:

O.3.1 : Real-time event recognition and forecasting algorithms

O.3.2 : Methods for adapting event patterns in dynamic settings

O.3.3 : Resilient real-time event recognition and forecasting algorithms addressing lack of veracity of data

However, it should be noted that one of the goals of WP3 is to establish appropriate definitions for the Complex Event patterns. This constitutes a target of WP3. Therefore, at this early stage of the project, WP3 cannot provide more specific requirements, since specific requirements themselves require specific pattern definitions.

Link to Use-case Scenarios All the maritime scenarios, with the exception of *SC12 Vessel in distress*, for which not enough information is available:

- SC11 Collision avoidance
- SC21 Monitoring maritime protected area (from illegal fishing)
- SC22 Fishing pressure on areas
- SC31 Detection of migrants / refugees and human trafficking
- SC32 Illicit activities

Data needed from other datAcron components Table 15 describes the necessary data the WP3 requires from other datAcron components to ensure its smooth operation. First, access to contextual and weather data stored in WP1 is necessary, as this data have spatial or spatio-temporal information that is extremely useful for identifying specific types of events. Then, access to the compressed surveillance data produced by WP2 is necessary, as this is the main source of information about the current position and movement of vessels.

Data needed from other datAcron components	Data modality	Acceptable latency
From WP1: contextual data and weather data	Archival spatial or spatio-temporal data	Tactical
From WP2: synopses	Compressed Streaming spatio-temporal data	Operational

Table 15: Overview of data needed from other datAcron components in order to support requirement R3.1.

4.4.3 Requirement 3.2: Event detection and forecasting in the aviation domain

This requirement targets event detection and forecasting in the aviation domain. With respect to flow management, the goal is to predict regulations that need to be imposed, detect affected flights and detect capacity/demand imbalances. With respect to flight planning, the goal is to detect a number of significant events for an aircraft's trajectory.

Inputs

- METAR data
- NOAA data
- IFS data
- ADS-B data
- Sector configuration
- Network data (GIPV,CFMU)
- Synthetic trajectories
- Aircraft Identification
- Flight Plan data

Outputs

- Regulations
- Demand and Capacity evolution and imbalance monitoring.
- Inconsistency between imbalance and regulation measure recognition
- Terminal Boundary Crossing Point
- Hold Entry
- Hold Exit
- Fly-by
- Fly-over
- STAR Entry
- SID Entry
- Aircraft not following planned route
- The above as forecasted events

Functions

- *Demand and Capacity evolution and imbalance monitoring:* detect excess of demand vs capacity
- *Inconsistency between imbalance and regulation measure recognition:* evaluate and forecast the system capacity to assume imbalance under nominal conditions
- *Terminal Boundary Crossing Point:* Indicates the point at which the trajectory crosses from one FIR into another. A named reference to the FIR being entered may also be identified.

- *Hold Entry:* Indicates that the associated trajectory point is a point at which the flight is expected to enter into a holding.
- *Hold Exit:* Indicates that the associated trajectory point is a point at which the flight is expected to exit from planned holding.
- *Fly-by:* the number of degrees of change between the current DTK and the upcoming DTK can provide turn anticipation for fly-by waypoint
- *Fly-over:* a fly-over waypoint is a waypoint that must be crossed vertically by an aircraft
- *STAR/SID entry:* SIDs and STARs aim to deconflict potentially conflicting traffic by the use of specific routings, levels and checkpoints
- *Aircraft not following planned route:* Partial or full lateral path change according to the initial flight plan. The typical reason of rerouting will be ATC or weather, and normally it is followed by a flight plan update containing the route.

Performance Requirements and Validation Criteria F1 score will be used as an accuracy metric. For the flight planning events, the latency should be at the *tactical* level. For the flow management events, at the *strategic* level.

Link to Research Objectives This requirement relates to the following research challenges

O.3.1 Real-time event recognition and forecasting algorithms.

O.3.2 : Methods for adapting event patterns in dynamic settings.

O.3.3 : Resilient real-time event recognition and forecasting algorithms addressing lack of veracity of data.

However, it should be noted that one of the goals of WP3 is to establish appropriate definitions for the Complex Event patterns. This constitutes a target of WP3. Therefore, at this early stage of the project, WP3 cannot provide more specific requirements, since specific requirements themselves require specific pattern definitions.

Link to Use-case Scenarios All aviation scenarios are relevant to this requirement:

- FP01 Real trajectory reconstruction
- FP02 Real trajectory enrichment
- FP03 Event recognition in trajectories
- FP04 Event forecasting in trajectories
- FP05 Data set preparation
- FP06 Trajectory clustering
- FP07 Trajectory prediction - preflight
- FP08 Trajectory prediction - preflight schedule based
- FP09 Trajectory prediction - real time

- FP10 Trajectory comparison
- FM01 Regulation prediction
- FM02 Demand and capacity prediction
- FM03 Resilience assessment

Data needed from other datAcron components Table 16 describes the requirements of WP3 in terms of data needed from other datAcron components. Similarly to the previous requirement, access to contextual and weather data stored in WP1 is necessary, as this data have spatial or spatio-temporal information that is extremely useful for identifying specific types of events. In addition, access to the compressed surveillance data produced by WP2 is necessary, as this is the main source of information about the current position and movement of aircrafts.

Data needed from other datAcron components	Data modality	Acceptable latency
From WP1: contextual data and weather data	Archival spatial or spatio-temporal data	Tactical
From WP2: synopses	Compressed Streaming spatio-temporal data	Operational

Table 16: Overview of data needed from other datAcron components in order to support requirement R3.2.

4.5 Requirements for Visual Analytics

The following requirements are (transitively) relevant to WP4:

- R1.1: Real-time integration/interlinking of spatial and/or spatio-temporal entities
- R1.3: Integration/interlinking over stored data
- R1.4: Spatio-temporal RDF querying of integrated data
- R1.5: Retrieval of spatio-temporally constrained subsets of integrated data
- R2.1: Computation of trajectory similarity and clustering
- R2.2: Pattern discovery
- R2.3: Prediction of trajectories and locations
- R3.1: Event detection and forecasting in the maritime domain
- R3.2: Event detection and forecasting in the aviation domain

4.5.1 Requirements for the integrated datAcron system

Visual Analytics does not represent a single, specific analysis technique but rather a methodological approach to gain insight into large, complex, noisy and often conflicting data, to develop and test hypotheses, and to build and understand complex analytical models. The key aspect is the collaborative work between the computer and the human analyst, whereby the human expert imparts background knowledge about the current analysis task's context and reasoning on the overall analytical process.

As such, WP4 Visual Analytics expands upon automated analyses developed and applied in the context of WP2 and WP3. A focus thus certainly lies on the tactical and strategic levels of acceptable analysis latency. Models deployed for fully automated, real-time data processing should definitely be informed and validated through Visual Analytics, but typically real-time monitoring and alerting presumes the existing suitable a model with parametrization (cf. Section 4.2.3) that is applied without direct human guidance.

Therefore, visual analytics approaches will operate on the same data types and structures identified in the preceding Sections 4.3–4.4. This includes the raw data and contextualized data (WP1) as applicable for model building, as well as analytical models and their current parameterizations (WP2, WP3) for model understanding, verification, and refinement.

Inputs WP4 receives inputs in terms of data sources from the following main data sources:

- The data accessible through the integrated datAcron system, specifically, the various outputs defined in Sections 4.3.2, 4.3.3, 4.3.4, and 4.3.5
- Models and their parameterizations from WP2
- Models and their parameterizations from WP3

Outputs As a result of visual analyses, input data is enriched and expanded as follows:

- Additional attributes expanding the input data (e.g., cluster IDs, association with specific spatial or temporal regions of interest) at different levels: per raw event, per complex event, per trajectory point, per trajectory, per moving object (which may contribute multiple trajectories)
- New spatial, temporal, and spatio-temporal objects (e.g., regions of interest, semantically annotated time intervals or cycles, complex events, model trajectories)
- Additional relations between entities persisting analysis insights (e.g., RDF triplets)
- Sets of parameters for automated processes (WP2, WP3)

Functions

- Visualization techniques that enable a human analyst to observe and reason about spatial, temporal, and spatio-temporal data (events, trajectories, context data) and patterns contained therein.
- Visual interfaces that facilitate the direct interaction with data selections, algorithms, and their parametrization with immediate visual feedback regarding the impact of parameter changes.

Performance Requirements and Validation Criteria Visual Analysis tools are inherently creative tools so quantitative evaluation is not applicable. Developed techniques must however be scalable to accommodate typical data sizes encountered in the context of the identified use cases.

5 CLASSIFICATION OF REQUIREMENTS

In this section, a classification of the requirements documented in the previous sections is presented, aiming to provide a comprehensive view of the requirements and, particularly, their correspondence to research objectives and use-case scenarios.

5.1 Data Management

Requirement	Description	O.1.1	O.1.2	O.1.3	O.3.1	O.3.2
R1.1	Real-time integration/interlinking of spatial and/or spatio-temporal entities		X			
R1.2.	Interplay of in-situ and stream processing components	X			X	X
R1.3	Integration/interlinking over stored data		X			
R1.4	Spatio-temporal RDF querying of integrated data	X		X		
R1.5	Retrieval of spatio-temporally constrained subsets of integrated data			X		

Table 17: Mapping of data management requirements to research objectives

Table 17 shows the correspondence between the data management requirements and research objectives. R1.1 addresses the need for real-time integration/interlinking and is relevant to the specific research challenge [O.1.2]: "Automatic, real-time semantic annotation and linking of data towards generating coherent views on integrated cross-streaming and archival data".

R1.2 relates to the research objective [O.1.1] "Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates" as it aims to accelerate the processing rate of data by processing the data already close to the data source. Moreover, R1.2 is going to support research objectives [O.3.1] "Real-time event recognition and forecasting algorithms that take full advantage of the data provided", and [O.3.2] "Methods for adapting event patterns in dynamic settings", as it can help to accelerate predictions and detection of patterns by online learning at a very early stage.

R1.3 also relates to the specific challenge [O.1.2], since it addresses integration over stored data. R1.4 and R1.5 cover the research challenge [O.1.3] "Efficient distributed management and querying of integrated spatio-temporal data". Furthermore, R1.4 addresses the research objective [O.1.1] in terms of producing a scalable processing framework over integrated data.

Table 18 depicts the mapping between use-case scenarios and data management requirements. The rationale for mapping each individual requirement with a use-case is described in detail for each individual requirement earlier in the text.

Scenario	Description	R1.1	R1.2	R1.3	R1.4	R1.5
SC11	Collision avoidance	X	X	X	X	
SC12	Vessel in distress / Man overboard	X	X	X	X	
SC21	Monitoring maritime protected area (from illegal fishing)	X	X	X	X	
SC22	Fishing pressure on areas	X	X	X	X	
SC31	Detection of migrants / refugees and human trafficking	X		X	X	
SC32	Illicit activities	X		X	X	
FP01	Real trajectory reconstruction	X	X	X	X	
FP02	Real trajectory enrichment	X		X	X	
FP03	Event recognition in trajectories	X	X	X	X	
FP04	Event forecasting in trajectories	X	X	X	X	
FP05	Data set preparation	X		X	X	X
FP06	Trajectory clustering	X		X	X	
FP07	Trajectory prediction - preflight	X		X	X	
FP08	Trajectory prediction - preflight schedule based	X		X	X	
FP09	Trajectory prediction - real time	X		X	X	
FP10	Trajectory comparison	X		X	X	
FM01	Regulation prediction	X		X	X	
FM02	Demand and capacity prediction	X		X	X	
FM03	Resilience assessment	X		X	X	

Table 18: Mapping between use-case scenarios and data management requirements.

5.2 Trajectories Detection and Forecasting

Table 19 shows the correspondence between the trajectories detection and forecasting requirements and research objectives.

R2.1 relates to [O.2.2] "Data analytics over the trajectories of moving entities", since trajectory similarity and clustering are basic structural units for more advanced data analytics over moving object data. R2.2 also relates to [O.2.2], since pattern discovery is included in the data analytics operations performed on trajectory data.

Clearly, R2.3 is directly related to research challenge [O.2.3] "Short- and long-term real-time forecasting of trajectories", as its main focus is on the prediction of the future position of a moving object and of the trajectory of a moving object in the future.

Finally, R2.4 mainly addresses objective [O.2.1]: "Cross-streaming, real-time detection of the trajectories of moving entities". However, maintenance of trajectory synopses is also related to [O.1.1]: "Producing a scalable, fault-tolerant framework for cross-streaming data integration, collection, and processing, producing data synopses at high compression rates", because it is also related to the compression of trajectory data and the creation of data synopses.

Table 20 depicts the mapping between use-case scenarios and trajectories detection and forecasting requirements. The rationale for mapping each individual requirement with a use-case is

Requirement	Description	O.1.1	O.2.1	O.2.2	O.2.3
R2.1	Computation of trajectory similarity and clustering			X	
R2.2	Pattern discovery			X	
R2.3	Prediction of trajectories and locations				X
R2.4	Computation of surveillance data synopses, reconstruction of trajectories by data synopses	X	X		

Table 19: Mapping of trajectories detection and forecasting requirements to research objectives

described in detail for each individual requirement earlier in the text.

5.3 Complex Event Recognition and Forecasting

Table 21 shows the correspondence between the complex event recognition and forecasting requirements and research objectives. Both requirements R3.1 and R3.2, targeting the maritime and aviation domain respectively, relate to the following research challenges:

O.3.1 : "Real-time event recognition and forecasting algorithms"

O.3.2 : "Methods for adapting event patterns in dynamic settings"

O.3.3 : "Resilient real-time event recognition and forecasting algorithms addressing lack of veracity of data"

The rationale is that all three research challenges relevant to event detection and forecasting are going to be addressed in both domains (maritime and aviation) under study.

Table 22 depicts the mapping between use-case scenarios and complex event recognition and forecasting requirements. The rationale for mapping each individual requirement with a use-case is described in detail for each individual requirement earlier in the text.

5.4 Visual Analytics

As established in Section 4.5, Visual Analytics expands upon the requirements identified for automated analysis and model building tasks of WP2 (Trajectories Detection and Forecasting) and WP3 (Complex Event Recognition and Forecasting). Therefore, the mappings between requirements and objectives (Tables 19 and 21, respectively) as well as between use-case scenarios and associated requirements (Tables 20 and 22, respectively) are equally applicable.

Scenario	Description	R2.1	R2.2	R2.3	R2.4
SC11	Collision avoidance			X	
SC12	Vessel in distress / Man overboard			X	X
SC21	Monitoring maritime protected area (from illegal fishing)		X		
SC22	Fishing pressure on areas		X		
SC31	Detection of migrants / refugees and human trafficking			X	
SC32	Illicit activities				
FP01	Real trajectory reconstruction				X
FP02	Real trajectory enrichment				
FP03	Event recognition in trajectories	X			
FP04	Event forecasting in trajectories			X	
FP05	Data set preparation				
FP06	Trajectory clustering	X			
FP07	Trajectory prediction - preflight			X	
FP08	Trajectory prediction - preflight schedule based			X	
FP09	Trajectory prediction - real time			X	
FP10	Trajectory comparison	X			
FM01	Regulation prediction				
FM02	Demand and capacity prediction				
FM03	Resilience assessment				

Table 20: Mapping between use-case scenarios and trajectories detection and forecasting requirements.

Requirement	Description	O.3.1	O.3.2	O.3.3
R3.1	Event detection and forecasting in the maritime domain	X	X	X
R3.2	Event detection and forecasting in the aviation domain	X	X	X

Table 21: Mapping of complex event recognition and forecasting requirements to research objectives

Scenario	Description	R3.1	R3.2
SC11	Collision avoidance	X	
SC12	Vessel in distress / Man overboard	X	
SC21	Monitoring maritime protected area (from illegal fishing)	X	
SC22	Fishing pressure on areas	X	
SC31	Detection of migrants / refugees and human trafficking	X	
SC32	Illicit activities	X	
FP01	Real trajectory reconstruction		X
FP02	Real trajectory enrichment		X
FP03	Event recognition in trajectories		X
FP04	Event forecasting in trajectories		X
FP05	Data set preparation		X
FP06	Trajectory clustering		X
FP07	Trajectory prediction - preflight		X
FP08	Trajectory prediction - preflight schedule based		X
FP09	Trajectory prediction - real time		X
FP10	Trajectory comparison		X
FM01	Regulation prediction		X
FM02	Demand and capacity prediction		X
FM03	Resilience assessment		X

Table 22: Mapping between use-case scenarios and complex event recognition and forecasting requirements.

References

[1] Daniel Abadi, Rakesh Agrawal, Anastasia Ailamaki, Magdalena Balazinska, Philip A. Bernstein, Michael J. Carey, Surajit Chaudhuri, Jeffrey Dean, AnHai Doan, Michael J. Franklin, Johannes Gehrke, Laura M. Haas, Alon Y. Halevy, Joseph M. Hellerstein, Yannis E. Ioannidis, H. V. Jagadish, Donald Kossmann, Samuel Madden, Sharad Mehrotra, Tova Milo, Jeffrey F. Naughton, Raghu Ramakrishnan, Volker Markl, Christopher Olston, Beng Chin Ooi, Christopher Ré, Dan Suciu, Michael Stonebraker, Todd Walter, and Jennifer Widom. The beckman report on database research. *Commun. ACM*, 59(2):92–99, 2016.

[2] The datAcron consortium. Grant agreement, number – 687591 – datacron. *European Commission, H2020-ICT-2015*.

[3] H. V. Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M. Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Commun. ACM*, 57(7):86–94, 2014.