International Journal on Advances in Software, vol 6 no 3 & 4, year 2013, http://www.iariajournals.org/software/

Complex Event Processing for Decision Support in an Airport Environment

Gabriel Pestana[1], Sebastian Heuchler[3], Augusto Casaca[1], Pedro Reis[2], and Joachim Metter[3]

[1]INESC-ID/INOV/IST, Lisboa, Portugal
gabriel.pestana, augusto.casaca@inesc-id.pt

[2]ANA-Aeroportos de Portugal, Lisboa, Portugal
pereis@ana.pt

[3]BIJO-DATA GmbH, Heldburg, Germany
sheuchler, jmetter@bijodata.de

***Abstract*— A new approach is proposed to the surveillance of Security and Safety occurrences concerning mobile objects in an airport environment, in particular to monitor aircrafts, vehicles and staff at the manoeuvring area for all weather conditions. A middleware platform receives localization information from the different mobile objects in the airport and merges that information through data fusion in the platform. The system outputs are shown in a high-resolution Graphical-User interface, providing a collaborative environment with the relevant information to the airport stakeholders. The outputs can be used by the stakeholders to take decisions on the best way to improve security and safety and also on the optimization of airport operational procedures in compliance with existing business rules. In this paper, the proposed system architecture follows an event-driven approach based on streams of occurrences processed in real-time. Therefore, it is suited for decision support. We will illustrate our approach by monitoring events occurred in an airport environment.**

Keywords-Mobility management; Situation awareness; Safety and security business rules; Location based services.

I. INTRODUCTION

In the airport environment, about 90% of critical events relate to accidents and incidents during ground handling services assisting parked aircrafts. The need for coordination of multiple activities occurring simultaneously requires, therefore, a continuous control of all ground movements, in particular during taxi operations. However, the current lack of context awareness and controllability is frequently identified as a causal factor for business rule infringements.

Without a solution capable of providing, in real-time, information related to the surveillance of operational occurrences, airport stakeholders (e.g., Airport Authority, Ground Handlers, Airlines) have not a reliable view of the overall situation to take well informed, in-time decisions [1] [2]. To assist airport stakeholders in their daily decision-making process we need an event-driven solution to combine data from multiple sources, capable of identifying meaningful events and responding to them as quickly as possible [3]. For instance, automatic detection of events related to overspeeding, safety infringements, unauthorized movements in restricted access areas, or any other location-based occurrence related to airport resources, staff or passengers.

Such capabilities would provide airport stakeholders with a new way to detect and analyse events in real-time, enabling them to adjust control actions according to the severity level of the observed event. However, to reach this level of operational intelligence and to avoid stakeholders to be drowned in data and be left without actionable information, they need to be assisted with data integration and fusion capability of unrelated events, based on business rules and policies. In this paper, the proposed system architecture follows an event-driven approach based on streams of occurrences in an airport environment. In particular, we will reference the running SECAIR project, whose platform will be deployed at Faro airport, Portugal.

The SECAIR project [4], is an European R&D project partially funded in the Eurostars program, brings a new approach to the surveillance of ground movements at the manoeuvring areas in the airport. It combines different localization technologies to detect and analyse movement patterns inside the airport terminal and at the apron area. The project relies on the development of an event observer system, which is capable of automatically identifying events and generating alarms in real-time. This means that a sliding window of one second is used to continuously provide streaming data to update the position of each surveyed object. A middleware platform provides data fusion to determine the localization of objects, which is determined by radio tracking techniques and video technology.

The middleware is part of a larger platform that, on the whole, will manage events related to movement patterns or hazardous situations. And because the project operates with different localization technologies simultaneously, multiple objects are surveyed, causing a very high volume of fine-grained data, which must be processed to determine movement patterns.

Current software architectures of decision support systems cannot deal efficiently with the processing of continuous event streams. Existing approaches focus on knowledge processing, but do not explicitly target the problems associated to real-time event processing. [5].

To test the capabilities of the system, a set of business scenarios addressing airport operational requirements were defined in close collaboration with ANA-Aeroportos de Portugal - the main Portuguese airport's management company, based on the following needs:

- Traceability of vehicles and Ground Support Equipment with automatic detection of unauthorised incursions into restricted access areas;
- Tracking and controlling of ground handling operations;
- Surveillance of aircraft ground movements within the apron area;
- Provision of context awareness about on-going operations at the apron area, triggering safety and security alerts with different levels of severity;
- Support the decision making process by providing a reliable view of the overall situation whenever a safety or security event is reported;
- Ensure that event notifications are sent to airport stakeholders based on their roles/operational needs.

The paper is organized as follows: Section II introduces the process followed for the specification of the system requirements. Section III presents the main software components within the multi-tier architecture designed for the SECAIR system. Section IV presents the system implementation. Section V describes the use case of the project together with the operational scenarios defined for testing the system. Section VI reports on related work. Finally, conclusions are included in Section VII.

II. SYSTEM REQUIREMENTS SPECIFICATION PROCESS

An airport, which is usually classified as a critical infrastructure, needs to encompass functionalities for an unambiguous surveillance of surface traffic caused by aircrafts and vehicles, without reducing the safety level [6].

In the airport environment, to efficiently coordinate ground movements caused by aircrafts, passengers and cargo, decision makers must be able to respond to an increasingly complex range of events. To reach such level of continuous data integration for decision-making, airport stakeholders must have a graphical informational cockpit capable of communicating large amounts of relevant information in an intuitive way. This is a feature typically assigned to corporate spatial dashboards.

Indeed, when combining human understanding with data visualization techniques it is possible to track the situation awareness and simultaneously get a well picture of the business operational performance. This is particularly true when the visualization of key performance indicators (KPI), describing how business is performing, are correlated with the representation of on-going events as point features over a cartographic layout. To cope with such goals, core operational requirements were specified in close collaboration with airport stakeholders, enabling us to express the requirements in terms of features that the SECAIR system should satisfy.

Within the SECAIR project the collaboration of airport stakeholders during the requirements specification phase followed the recommendations of the IEEE 42010 standard. [7]. The standard states that a well-formed requirement is a statement of the system functionality that must be met or possessed by the system to solve stakeholder's objectives, and that is qualified by measurable conditions and bounded by constraints. In such approach, user requirements also generate a structured collection of information that embodies the requirements of the system mapped to the stakeholders concerns.

Figure 1 presents the schema of the system requirements specification (SyRS) process adopted within the SECAIR project for capturing safety and security requirements. The approach provided a "black box" description of what the system should do, in terms of the system's interactions or interfaces with its external environment. Therefore, the SyRS was essentially used as a technique to discover and document, through elicitation sessions with airport stakeholders, the system capabilities and its required behaviour. Besides distinguishing between requirements and their attributes (conditions and constraints), the SyRS also helped identifying, for each operational scenario, the corresponding business indicators used to validate whether the results (i.e., movement path of a surveyed object) are within acceptable thresholds.

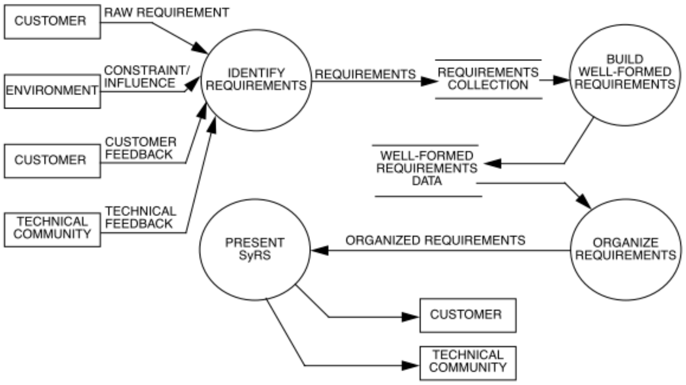


Fig. 1. System Requirements Specification process.

Afterwards, technical requirements, expressed as constraints placed on the SECAIR system, were analysed having in view the technical limits of each underlying technology. Besides these technical requirements, environmental influences affecting the system were also considered and classified into categories (e.g., political, standards and organizational policies). SECAIR conducted a research in each of these categories to ensure that the system conforms to all regulations that influence the airport sector.

The combination of all these artefacts was used to shape the set of operational scenarios that hold a high potential to demonstrate the benefits of the system developed under the SECAIR project.

For instance, in order to optimize gate to gate operations, the important issue to be considered is related with aircraft assisting tasks and the surveillance of ground movements in order to manoeuvre safely and efficiently on the movement area.

Concerning the objectives of SECAIR, in this paper, we are just considering the apron area and the service roads in the airport. The apron is a defined area intended to accommodate aircraft for purposes of loading and unloading passengers, mail and cargo, fuelling and parking. Its design should take into account safety procedures for aircraft manoeuvring, taking into account the specified clearances and following the established procedures to enter, move within and depart from apron areas.

By analyzing the activities and interactions that occur between aircraft and ground vehicles, we conclude that most of them take place in the apron areas. It is a wide variety of complex operations, including the handling of aviation fuel, the movements of vehicles, aircrafts and airport staff with different tasks to perform. They are all concentrated in a restricted area and with a short turnaround time, increasing the possibility of a potential conflict. Most of the time the activities performed in the apron are of vital importance for the safety of an aircraft during its subsequent flight.

Regarding the aircraft servicing mentioned above, we can refer to the following aircraft servicing operations involving ground vehicles and equipments:

- Passenger, baggage and cargo loading/unloading;
- Galley service;
- Fuelling service;
- Provision of compressed air for engine starting;
- Aircraft maintenance;
- In some cases, electric power and air conditioning.

As illustrated in Figure 2, there is a wide variety of ground operations, which contribute exponentially to conflicts and increase the risk of accidents/incidents in the apron area. Figure 2 outlines some of the vehicles used to assist parked aircrafts. The coordination of the movements of all those vehicles, sometimes with aircrafts parked at adjacent stand areas requires an effort to avoid delays and to comply with safety procedures. Besides being a very constrained area, in extreme situations (e.g., rush hours, bad meteorological conditions) the risk of operational inefficiencies at the stand area can compromise airport procedures, leading to a dysfunction of the airport and, eventually, compromising the required level of safety. Such stressing situation tends to increase the need to accurately monitor all ground movements within different areas in the airport air side, namely inside the stand area. However, without a system capable to continuously and accurately track all ground movements (i.e., vehicles, equipment and persons) inside restricted access areas, infringements to safety rules might not be noticed by apron controllers.

In the apron area, the most common type of incidents and accidents fall into the following categories:

- Ground equipment driven into aircraft
- Unmanned equipment rolls into aircraft
- Aircraft rolls forward/backward
- Towing vehicle strikes aircraft
- Aircraft contacts object/equipment.

Concerning the service roads, every effort should be made to plan air side service roads so that they do not cross runways and taxiways. Several solutions can be found to minimize the possibility of conflicts between aircrafts and vehicles/equipments, and one of them is considering road tunnels avoiding the crossings at taxiways. However, in all situations, vehicle drivers must comply with aerodrome regulations and take due care and attention to avoid collisions between vehicles and aircraft and other related hazards.

The SECAIR project addresses these concerns by monitoring all ground movements and by triggering alert messages for each detected infringement. The localization of each moving object is represented as point features, labelled with a colour code to call the attention of the controller (at the situation room) whenever a safety infringement is detected.

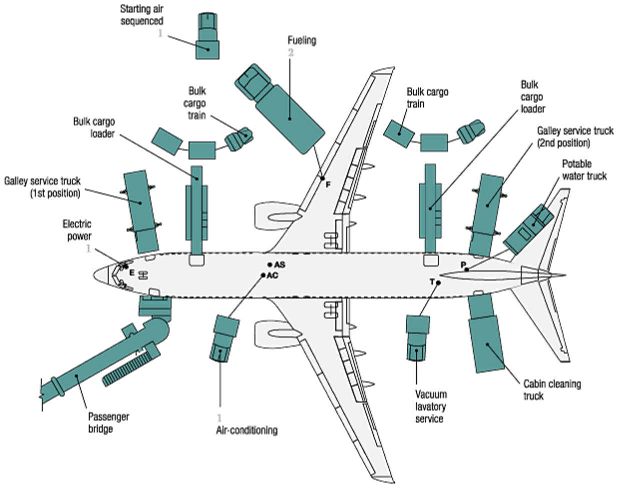


Fig. 2. The B-777 being serviced during a turnaround with the help of ground systems and mobile equipment. Source: Boing 777 Airplane Characteristics for Airport Planning.

Integrated airport operations planning, advanced surveillance techniques, ground-based safety nets and new runway management tools are amongst the improvements that will allow the aircraft to be served more efficiently from gate to gate. The SECAIR project will contribute to reduce ground hazards that affect flight safety, reduce aircraft ground damage, reduce personnel injuries and also, in the security domain, prevent acts of unlawful interference.

III. SYSTEM ARCHITECTURE

An event-driven solution typically consists of event observers (i.e., localization technologies) and event consumers. The SECAIR system operates by observing a set of events that happen in the external environment. Because localization technologies are continuously emitting data, they are particularly suited for Complex Event Processing (CEP). As outlined in Figure 3, an event starts at the Communication tier, with the sensing of a fact (e.g., safety occurrence) that is converted into a data stream and sent to the Data Fusion Algorithm (DFA) at the middleware of the Application tier. To this end, any update to the position of each surveyed object is provided as a single time streaming data.

Location-based data tend to be strongly correlated in both time and space. For instance, position and speed data measured by one localization technology is highly correlated to the data collected by another adjacent localization technology. Similarly, readings observed at one time instant are highly indicative of the readings observed at the next time instant. This is particularly relevant because airport stakeholders are not interested in individual readings in time or individual devices in space, but rather in application-level concepts of temporal and spatial granularities.

In this paper, we use the term spatio-temporal to designate data related to both time and space dimensions. Since the project is closely related with location-based data, the trigger to most events derives from the movement of the observed objects. For each reported position, the system might require access to additional data about the object in order to analyse the event in conformity with the role of the object or to consider changes in the object status, changes in the object descriptive data or even changes in the spatial context where the object is located. The complexity to support such in-time actions increases when the system has to process data from multiple technologies while considering a set of business rules to take appropriate actions.

A. *Overview*

The generic architecture of the SECAIR system is shown in Figure 3. At the periphery of the system we have the Event Observers and the Event Consumers. The former corresponds to the devices and localization technologies, whereas the latter receives event notifications and presents them to the end-users so that they can react accordingly.

The SECAIR system implements a client-server architecture structured into three tiers (see Figure 4). The Communication tier operates with heterogeneous wireless localization technologies (sensors), each one collecting data about the location of the observed objects. Each device of the adopted localization technologies is responsible to continuously generate location-based data streams; for some technologies the period is less than one second. However, the sensor data is too fine-grained and do not meet airport stakeholder concerns as they are not interested in individual sensors data, but in application-level concepts of spatio-temporal granularities. Therefore, domain data events are required.

At the Application tier, the middleware is responsible to integrate and process incoming data from the Communication tier, delivering event streams with reliable location data to the Business Logic. This is performed based on a data fusion process that computes positioning data to provide accurate and reliable location data about the observed objects. Prior to the data fusion process, a set of location-based data for the same object has to be integrated. But after the data fusion process, a single computed position per object is provided, expressing a pre-processed set of events.

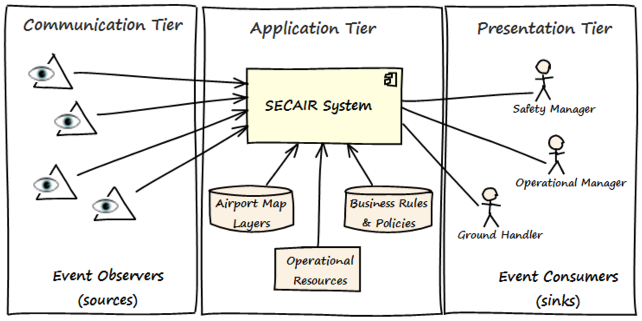


Fig. 3. A high-level view of the SECAIR system.

In order to be understood and processed, events need to be integrated with the business context, generating domain data events. Domain data events can be derived by mapping raw sensor data to domain concepts. For instance, domain data events correlate data of sensors located in one specific road segment, and evaluate speed limit infringements caused by a vehicle, or an alert caused by a tagged passenger for an unauthorized entrance into a restricted access area.

At the Application tier, there is logic that operates by interpreting a set of business rules to derive composite events from the events that have occurred. This means that one of the goals is to timely process generic data, not necessarily event notifications, tied together by spatio-temporal relationships, in order to perform a diagnosis based on existing business rules and organization policies. Such composite events are presented to decision-makers using the Client Application at the Presentation tier and might characterize a situation that is undesirable for the decision maker.

B. *Communication Tier*

The Communication tier ensures that the data observed by each device is timely transmitted to the Application tier at the server side. It is also aware about data communication requirements, including the wireless network required to cover the operational areas. The selected localization technologies acting as event observers are:

- The Stand-alone Global Navigation Satellite System (GNSS), used together with a Wi-Fi communication device, to collect and transmit, every second, the coordinates of the vehicle position;
- The Ultra Wide Band (UWB) system to provide immunity to multipath propagation and precision range measurement capability. The IEEE 802.15.4a UWB standard implements precision location measurements when the monitored objects are close to large metallic infrastructures;
- The Video Surveillance and Tracking System (VSTS) consisting of multiple video cameras installed at predefined locations to fully cover the target area. The video data collected by each camera is processed by the VSTS sub-system to detect, track and classify the foreground objects within the area of interest;
- The Radio Frequency (RF) localization system consisting of mobile devices and antenna units mounted in the area of interest. It measures the position of a mobile de-

vice attached to the observed object (e.g., passenger or staff) in the area of interest.

There is no single technology, which can provide satisfactory performance in all environments and scenarios; therefore, various localization technologies have to collaborate in order to deliver a flexible localization system, instead. Sensor data fusion will combine sensor data from different localization technologies to outperform any individual systems working alone.

C. *Application Tier*

At the server side, the Application tier is segmented into three conceptual areas. The middleware area to hold the Data Integration and Data Fusion, the Business Logic area holding the software components responsible for the system operational intelligence and the system operational database managed by Microsoft SQL Server 2008.

Middleware

The middleware is responsible to continuously provide a calculated position of each observed object to the Business Logic. The positions of the observed objects are continuously transmitted and are coherently integrated using data-fusion techniques to address multipath effects reduction and improve quality of location (QoL). This approach, besides reducing installation costs, also contributes to increase location accuracy, achieving a better coverage range with the same amount of equipment. [8]

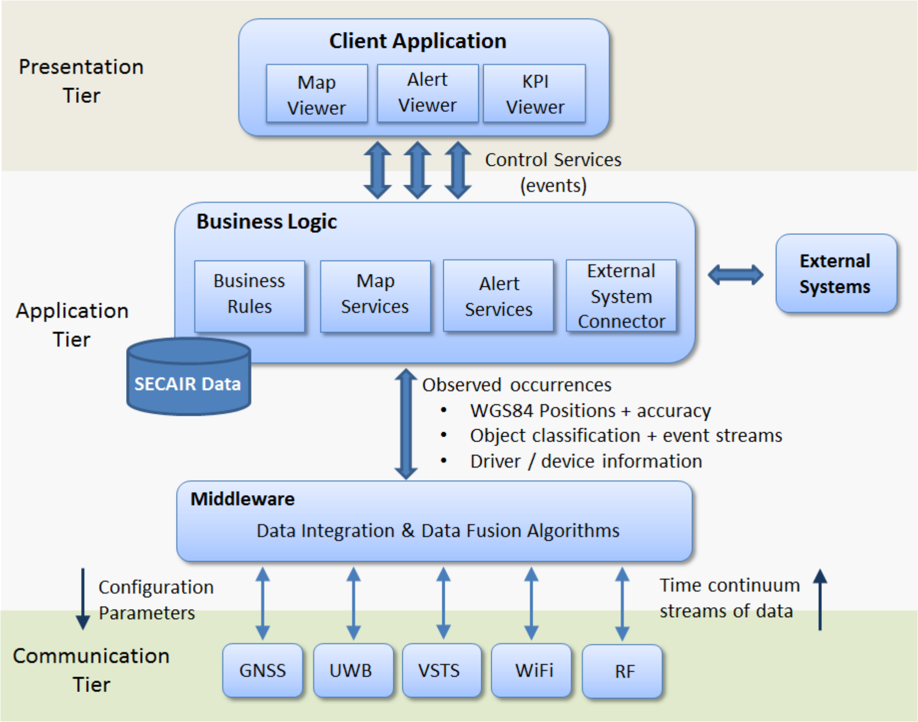


Fig. 4. The architecture of the SECAIR system.

Business Logic

For each data stream triggered by the Middleware, a set of actions are required to correlate additional data about the observed object with existing business rules and metadata about the spatial-context to determine domain events. This is accomplished at the Business Logic by four core software components. A short description of each software component is presented next.

- **Business Rules**, establishes the link between the definition and the execution of all business rules within the system, enabling organizational policies and the repeatable decisions associated with those policies, such as restricted area incursions, to be defined, deployed, monitored and maintained centrally at the server side. When changes occur at the business level, this service also assists in discovering the set of existing rules that are influenced by those changes. The Business Rules services interact with the operational database to store incoming events with the right classification.
- **Map Services**, this component is responsible to manage the spatial-context that characterizes the airport environment where the observed objects operate. Within the project scope, the airport layout is represented with a set of overlapped layers in a standard format [9]. To efficiently support the spatial database workload and the degree to which spatial functions are required, a geographic information system (GIS) engine was specifically designed. This GIS engine copes with challenging requirements related to scalability and real-time representation of multiple moving objects and dynamic changes to the spatial context, without

compromising the overall performance of the system. Depending on the nature of the detected event, the Alert Services will interact with the GIS to generate an alarm to be broadcast to each connected client application. A log record of all events is stored for historical data analysis purpose.

- **Alert Services**, for each business rule infringement a proper alert is generated by mapping location-based information related to the observed object (event) with domain concepts. This means that the data for each observed object has to be analysed to determine if a composite event (e.g., severe safety infringement) occurred or if a business metric needs to be updated. For each event being detected by the system, a semantic meaningful alert message will be triggered, with the corresponding relevance and severity risk. A dendrogram with weighted nodes is used to structure relationships between business indicators. Granular indicators are at the bottom (leaf) level and derived indicators (usually more aggregated) are at the nodes of the dendrogram. For instance, a business user can configure the system to inform about how many stand areas incursions were performed by a driver in a specific time period or day of the week.
- **External System Connector**, this software component is responsible for handling the interoperability with external systems, for instance, to collect data related with flight schedules, resources and assigned tasks. With such approach, location based data for each observed object can be coherently correlated with metadata from external sources, enabling the surveillance and track of events to be performed according to business logic/rules [10]. The Application tier, being responsible for implementing airport business logic, seeks ground for the coexistence and balance between the dual trends of the airport industry: increased demand for air travel and strengthened aviation safety and security [11].

Depending on the business rule being infringed, a specific event (alert message) is triggered to the end-users at the Presentation tier. This is done by creating a subscription offered as a public endpoint by the system. The Business Logic sends the requested data, either as a stream of updates (event-based queries) or as a chunk of current state data (instant-queries). The first are triggered on a certain event, e.g., an object moving into a specific area. The Business Logic can create an event subscription (“tell me about objects moving into a specific area”) to be notified on that event (“an object moves into an area”) and perform specified actions accordingly (“alert: object moved into restricted area”). This kind of subscription may be triggered often, or never, depending on how the event occurs. On the contrary, the result of an instant-query is always returned immediately and is not dependent on any event. This kind of query is useful to retrieve the current state of an object. For instance, “give me a list of all objects, which are currently in a certain area” or “tell me the current battery status of an object”.

The data structure used to define each object position is presented by the class Position. The coordinates of the object position are presented as a point feature dataset that is used by the GIS engine to determine the location of each point feature (i.e., object location) over the airport layout. The airport layout is represented by a set of overlapped thematic map layers (also known as feature datasets, see Figure 5), some of them are polygon features representing operational areas with a predefined set of metadata to store specific business rules (see Figure 6). All layers have a common metadata structure; however, some polygon layers (e.g., Serviceroads, Stand, Taxiways, etc.) also have specific metadata attributes relevant for spatial context-semantic data analysis. These metadata are used by the Business Logic to enforce the application of predefined safety and business rules.

The GIS engine performs a topological point-in-polygon overlay operation to determine which points (i.e., IDObj) from the Position feature dataset are contained within the polygons of the airport layout. For each intersected layer, the Business Rule component interacts with GIS engine to check if any of the specified business rules is infringed. For instance, the Serviceroad layer includes attributes to specify the speed limit, category and status of each roadway segment within the airport, but the Stand layer just includes a status attribute to identify if a specific stand is open (i.e., with no parked aircraft), closed (i.e., with a parked aircraft) or deactivated (e.g., in maintenance). When a point is inside a specific roadway segment polygon, that information is passed to the Business Rules component to validate for operational rule infringements.

The QoL attribute is used to determine the accuracy of the reported position. This means that points reported with a QoL value higher than 10 m are labelled in red to indicate lack of accuracy in the IDObj position, QoL with values between 7.5 m and 10 m are labelled in yellow to indicate that the system is not able to assure the exact position of the IDObj and QoL with a value lower than 7.5 m are labelled in green to indicate that the IDObj position is accurate. The field tests performed so far were able to achieve position accuracy with a QoL between 0.5 m and 3 m.

A typical example of a safety rule infringement refers to speed limit, e.g., an object of type “vehicle” circulating at 30 Km/h within a roadway segment with a speed limit metadata of 25 Km/h will trigger a speed limit infringement alert message. In the same way any vehicle moving inside an open stand area will trigger a stand area incursion alert message. The IDObj provided within the Position data structure is used by the Business Rules to obtain more information about the IDObj (e.g., to which airport operator it belongs or if it is a priority object such as a “Follow-Me vehicle”). For each new position reported by each object, every second, the Business Rules might need to validate the IDObj business data before triggering an alert message.

As presented in Figure 4, additional business data about the IDObj are provided by the External Systems component. A data interoperability connection, established between the SECAIR system and the existing airport systems, enables the decision support capabilities of the SECAIR system to access some operational data (e.g., flight data and resource data) and correlate those data with the metadata from the airport layers and the data from the monitored IDObj. This computation is performed at the Business Logic in less than one second for each position reported by the middleware for each IDObj. In this way the SECAIR system is able to validate for infringements to some business rules defined by each airport.

The system surveillance capabilities also includes collision avoidance to prevent, in real-time, ground damage to aircraft, equipment and potential injury to staff, operating within close proximity during ground handling operations.

To cope with such requirement (i.e., detection of the likelihood of collision trajectories), in the current version of the SECAIR system each object is represented as a point feature with a dynamic geofence around the object. For an aircraft, this geofence is designated as a clearance level, represented by a safety circle with a radius defined according to the specifications presented in ICAO Annex 14 [12] (i.e., wingspan and length of the fuselage). As such it is sufficient to generalize aircrafts to point locations instead of considering their real dimension. For vehicles, the geofence is designated as a protection area, corresponding to a rectangle at the front and another at the rear of the vehicle. The length of these two rectangles is determined by the vehicle category (e.g., Passenger Bus, High-Loader, Catering, Refuelling vehicle, etc.).

When any of these safety buffers (i.e., geofences) intersect, an alert for possible collisions between moving objects or other infrastructures is automatically triggered. For instance, when the vehicle protection area intersects the aircraft clearance a warning is triggered to indicate the driver to move to a safety distance. This functionality although in a preliminary version has been successfully tested when objects interact, namely for vehicles at the proximity of moving aircrafts or for interactions between Ground Service Equipment (GSE). For aircrafts parked at the stand area, the clearance level is not validated during ground handling operations. However, it will trigger a safety alert for any interaction of a parked aircraft with a moving aircraft (i.e., intersection of two clearance levels).

Collision avoidance between vehicles revealed to be a challenge difficult to accomplish because in most areas within the airport it is physically possible for a vehicle to move into any directions. Therefore, the specified protection area corresponds to a safe distance at the front and rear of each vehicle.

Operational Database (SECAIR Data)

For simplicity, the core data handled by the Business Rules software component are represented in Figure 4 by a single database. However, the physical implementation includes one relational database to store dynamic informational entities such as vehicles, operators, flights and aircrafts, and another to store the static airport cartographic layout, i.e., thematic map layers.

Both databases are managed by the Business Logic using the Microsoft SQL Server 2008, a database management system capable of dealing with business data and map features, describing the airport layout, within the same database.

In the SECAIR system, all thematic layers use the World Geodetic System 1984 (WGS84) as the spatial reference system in conformity to the specifications of the A-SMGCS manual [13] and comply with the ED-119 standard. The ED-119 standard defines the physical dataset requirements to develop the airport mapping. These include: geometry accuracy requirements, feature rules and descriptive (metadata) attributes. Since each layer is spatially referenced, they overlay one another and can be combined in a common map display.

The resulting geo-database consists of vector and attribute features. The vector features represent geometric feature instances that are classified as points, lines or polygons. As outlined in figure 4, each observed occurrence reported by the Middleware to the Business Logic is stored in the geodatabase as new point features. This means that a new record with the object "Position" data structure is created within the object Position layer for each new position being reported.

The critical operational areas within the airport must have a polygon layer with specific metadata. Figure 5 presents an example of the type of metadata for the Stand layer (AM_PARSTANDAREA) and the set of metadata for a specific feature (i.e., stand named S14) within the Stand layer. Examples of critical polygon layers with metadata to validate safety infringements are: Runway area and Runway thresholds, Taxiways, Apron, Stand, Service road and holding lines just to mention some. In the SECAIR project there are fifteen critical polygon layers from a total of twenty nine layers used to characterize the airport layout.

D. Presentation tier

At the Presentation tier, the surveillance capability of the SECAIR system is presented to end-users in three different ways.

The Map viewer corresponds to a graphical layout managed by a GIS engine specifically designed to cope with two main requirements, namely to be operated by non-skilled airport stakeholders and to cope with airport stakeholder data processing needs for each spatio-temporal event. The Map Viewer represents the moving objects as colour coded point features with a timestamp and a set of descriptive data about the resources causing, for instance, a safety infringement; this might include data about the aircraft, vehicle, driver, flight data or layout of the area where the event occurred. The Map Viewer GIS engine is responsible to compute and represent in real-time (i.e., up to one second) the movements of all observed objects, computing simultaneously dynamic changes to the spatial context derived from daily airport business activities. Additional metadata (e.g., speed, logged driver, vehicle category) about each surveyed object are also provided.

The user can interact with the features of each layer by selecting, for instance, a specific stand and manually change its status, or obtain information about flights and tasks assigned to a specific stand area. It is also possible to verify which road segment is operational and check for traffic circulation rules that apply to the selected road segment (e.g., speed limit for different visibility conditions and directions of traffic flow) or analyse how many speed limit infringements occurred.

The Alert Viewer shows the corresponding textual description of alert messages in terms understandable by the end-user. This means that for each moving object causing an event, the Alert Viewer at each client application will present the alert messages contextualized with business semantic and ordered by severity level. All alert messages have a start and an end time, plus a set of additional descriptive data related to each event.

The KPI Viewer presents, in a spatial dashboard, the values of key performance indicators describing how the business is performing. The correlations between KPIs are mapped in a dendrogram structure. Each node of the dendogram carries some information needed for graphical visualisation of the data using size and colour coding.

IV. SYSTEM IMPLEMENTATION

A. *Communication protocol*

We developed a protocol for communication between the hardware devices and the middleware (see Table 1). This protocol has the following characteristics:

- Independent of operating systems and hardware architectures
- Small packet size
- Clearly defined operations, even in complex use cases
- Full coverage of value range
- Easy to implement in various languages (complete implementation available in C# and C++).

Table 1. Communication protocol packet contents.

Bytes	Description
2	Magic "LP" = 0x4C50
1	ProtocolVersion = 0x02
8	Timestamp
16	Source ID
16	Message type qualifier (ID)
4	DataLength: Length of upcoming data field
variable	Data Area: Actual data field with payload

1) Timestamp

The timestamp is an unsigned 64 bit integer value storing the time when the position of the device is measured with nanosecond-precision.

2) Source ID

The source ID identifies the sensor or data system that originally created this message. Together with the timestamp, this can be used to generate an UUID compatible with RFC 4122.

3) Message type qualifier

The type qualifier is used to specify the type of content in a message. Only if the qualifier is known to a system, the message can be understood successfully. Messages with unknown qualifier should be ignored.

The message types include positional or environmental information, device health checking, firmware updates and several others.

B. *Fusion of location data*

Fusion of location data faces two main challenges. The first is associating objects tracked by the radio based sensors with those tracked by video technologies. The second is the fusion of those associated objects regardless of the technology it stems from.

To satisfy both requirements there are two interacting cycles. One is of high frequency and predicts positions. The other is of lower frequency and manages the association of objects.

The engine featuring these cycles merges data of widely varying quality into a single, continuous and seamless position track. Since the association algorithm knows which objects were tagged (e.g., staff), any observed objects without association can be considered non collaborative and originating potential issues. The discussion of the data fusion algorithm is outside the scope of this paper.

C. *Control Service Implementation*

Within the SECAIR project, control services are the building blocks to implement the business logic, enabling end-users to dynamically manage and interact with the target environment, changing the status of the business context as well as obtaining detailed information about moving objects and receiving automatic alert notifications about any safety infringements or incursions into restricted access areas. All location data are delivered by the platform to the presentation tier following an event-driven approach. End-user applications can subscribe to different events.

Some control services need to integrate with existing airport systems. Within the scope of the field tests at Faro airport, the external airport system used for demonstration is the Flight Information Data System (FIDS). The goal of such integration was to receive airport operational data in order to obtain, in real time, flight data.

We are assuming that location data corresponding to each object is provided by the underlying system. The quality of the data should be good and there should be no gaps (missing data). However, the services should be able to cope even with poor data.

Since we are working in a high security environment, delays or even crashes are to be avoided or at least handled properly. Logging relevant service-internal events and states is one of the basic ways to provide a reliable system. This can be extended by informing the system administrator on any encountered problems like failing services, starving or full buffers, network issues and configuration mistakes.

First of all, the very large system is broken down into various small components, each handling only a specific task. There is a basic differentiation between low level services that work with “raw data” and high level services that are based on those low level services.

Low level services

- Location: Current and historic positions of tracked objects. Event on new positions.
- Entity: List of all tracked objects along with some properties. Event on changed, added or removed objects.
- Map: List of all available areas along with some properties. Event on changed, added or removed areas.
- Storage: List of various data that are used and synchronized globally. Event on changed, added or removed data.
- Integration: A set of various separate services that are used to integrate the system with existing infrastructure. This includes, for instance, a service connecting to the airport flight management system and providing its data. Another included service will use these data to dynamically update the configuration of the alert services, in this case, allowing or disabling certain areas to be passed through by certain kinds of vehicles depending on whether or not an aircraft is moving through that area.

High level services

- Alerting: alerts are stored as a list of critical events along with some properties. Push notifications are triggered on added or changed alerts. Alerts are created by multiple separate services that use the above lower level services to create or change alerts based on certain business rules. For instance, by querying Location (position of an object), Entity (type of object), Map (areas) and Storage (stores permission table), an object of not-allowed type moving into a restricted area could create an alert.
- Multi-Tracking: based on the object position it is possible to activate in a new window the surveillance of the movements of the selected object. This is useful when there is a need to closely follow the movement pattern of one specific object.
- Collision Avoidance: the system can act preventively by determining if two objects (e.g., vehicle-vehicle or vehicle-aircraft) have a collision trajectory. This service is also relevant to assure clearance levels between aircrafts and to alert when a vehicle is not within a safe distance from moving aircrafts.
- Identify Resource: this service provides additional information about the selected object or map feature. In the first case, providing business data collected from the airport system, for instance, to describe the vehicle characteristics, obtain information about the logged driver or to correlate the current position of the vehicle with assigned tasks or flight schedules.
- Path Analysis: this service is used to draw a line as the object moves. The output enables the end-user to get a visual perception about movement patterns, understand patterns in specific time periods like rush hours or detect which vehicles are frequently out of predefined trajectories.

These control services, in general, correspond to actions performed by end-users individually; therefore, it is not feasible to introduce an overhead computation by implementing them at the server side. Some control services relate to business logic and need, therefore, to get access to business rules and business data to be able to actuate over predefined mobile devices or objects within the airport. Concerning the control service implementation let us describe the service usage, scaling and redundancy, connections via WCF and generic collection library.

1) *Service usage*

All services have to offer two basic things: a list of all data and notification of changes. Notification refers to the service actively informing the client of events. For instance, the location service will let all its clients know when a new position arrives. An alerting service could provide notice of new or changed infringements.

An event-based architecture is a pragmatic and reliable way to ensure fast response times, meaning that any occurrences are shown immediately (less than one second) and, as far as possible, dealt with automatically.

2) *Scaling and redundancy*

Unlike traditional monolithic systems, the services run as independent components and, as a matter of principle, are accessed via a network. In fact, it is no problem to run some of the services in a different part of the world if that would be advantageous.

One obvious advantage of this strategy is scaling. Since services run in parallel, just adding a few more machines will directly increase computation capability.

Also, losing a machine due to, for instance, network issues or hard disk failure, does not pose a problem. Another service will simply take over the work of the lost service for the time being.

To enable this architecture, data must not be stored locally on a machine. Instead, we are using distributed high performance databases connected via gigabit Ethernet.

3) *Connections via WCF*

The Application Programming Interface (API) is accessible via Windows Communication Foundation (WCF), one of the most common interchange formats. All connections are encrypted. Similar to the web service model much in fashion nowadays, instead of traditional large and complex interfaces, a simple and clean interface is used. However, in addition to traditional web services, we make use of events to decrease network overhead and provide real time updates.

Service interfaces basically feature only two kinds of data: data lists and events. Noticeably, they do not offer configuration to a client and are thus stateless. This is a feature common to web services and means clients have to specify a complete query at all times. The advantage is that the services do not need to keep track of their clients, simplifying their code a lot and increasing performance generally.

The reasons for choosing WCF over competing technologies lie in its simplicity, reliability, interoperability and high performance coupled with many years of knowledge of using WCF "in the field". Remarkably, WCF is flexible enough to power both simple REST web services and complex state-aware multi-interface services. WCF can handle connections with clients written in different languages (e.g., Java) too, so interoperability with other system, for instance Linux, is a non-issue.

4) Generic connection library

Since all components use WCF, a generic library was established. It allows easily setting up servers and clients for any WCF service, no matter if a simplex or duplex connection is available. It also includes automatic endpoint creation for TCP/IP, HTTP (port 80) and metadata exchange.

Connection to the server is always fault aware. Any failure will transparently be encapsulated and provided to the client process, while avoiding any critical consequences. This means that all client services need to be aware of the fact that any connection may fail to work at any time, and cache their requests and data accordingly.

D. Monitoring

For easier deployment, we make use of WS Discovery. This allows services to publish themselves so that discovery-enabled clients can find them without any configuration. Since Discovery is a standard protocol, it is interoperable with other clients, services and proxies..

A supervision program is used to watch and control the state of all services. It ensures that all services are up and running, possibly restarting those that fail. It also displays the health state of the complete system to an administrator, additionally informing of important issues (e.g., a service failed permanently) by email or other means.

E. Graphical User Interface

The interaction between the software components at the Graphical User Interface (GUI) and the Business Logic are performed mainly through the lower level services to receive domain events. Based on that information, the end-user has the possibility to explore the information in more detail by executing the high level services.

V. USE CASE

In order to validate the SECAIR system, a system prototype for a pilot test is being installed at the Airport of Faro. For field tests, airport vehicles are provided and cameras from the VSTS system are installed. The existing Wi-Fi network is strengthened at specific operational areas in the airport.

Some vehicles will be equipped with an onboard unit, a touch screen display and a radiofrequency reader to automate the driver authentication procedure. All personnel participating in the field tests will receive a radiofrequency card. The association of the driver with the vehicle is performed automatically each time the card is read by the radiofrequency reader installed in the vehicle. At least two client applications are deployed, one situation room to the Apron Control Centre for airport operators and a second situation room to support other airport stakeholders (e.g., Ground Handler).

Figure 5 outlines the airport air side areas selected for the site tests. Three operational areas were considered: the passenger terminal (Boarding Gates 01 and 02) for indoor test scenarios, adjacent indoor-outdoor transition area (to demonstrate ability to track targets moving from indoor to outdoor and back) and the Apron area adjacent to stands 14 and 16 for outdoor test scenarios.

Indoor scenarios (see Table 2) include: zone intrusion detection, target tracking and left behind luggage. Indoor-outdoor transition areas include the surveillance of people at the boarding gates for (dis)embarking procedures. Indoor scenarios reflect operational procedures related mainly with the observation of people and baggage in restricted access areas within the passenger terminal:

- Traceability of a person at the boarding gate area;
- Localization capability of the SECAIR system in the transition area from passenger terminal into restricted access areas (outdoor);
- Location obtained by fusion of data obtained from the following technologies: VSTS and RF.

Outdoor scenarios (see Table 2) cover an area defined by 130x130m of the apron comprising Stands 14 and 16. The outdoor scenarios reflect operational procedures related mostly with the observation of vehicle movements and operational procedures related to parked aircraft assisting tasks:

- Traceability of vehicle and driver at the Apron area;
- Automatic detection of drivers without driving permission / not logged;
- Location obtained by fusion of data obtained from the following technologies: VSTS, GNSS and UWB.

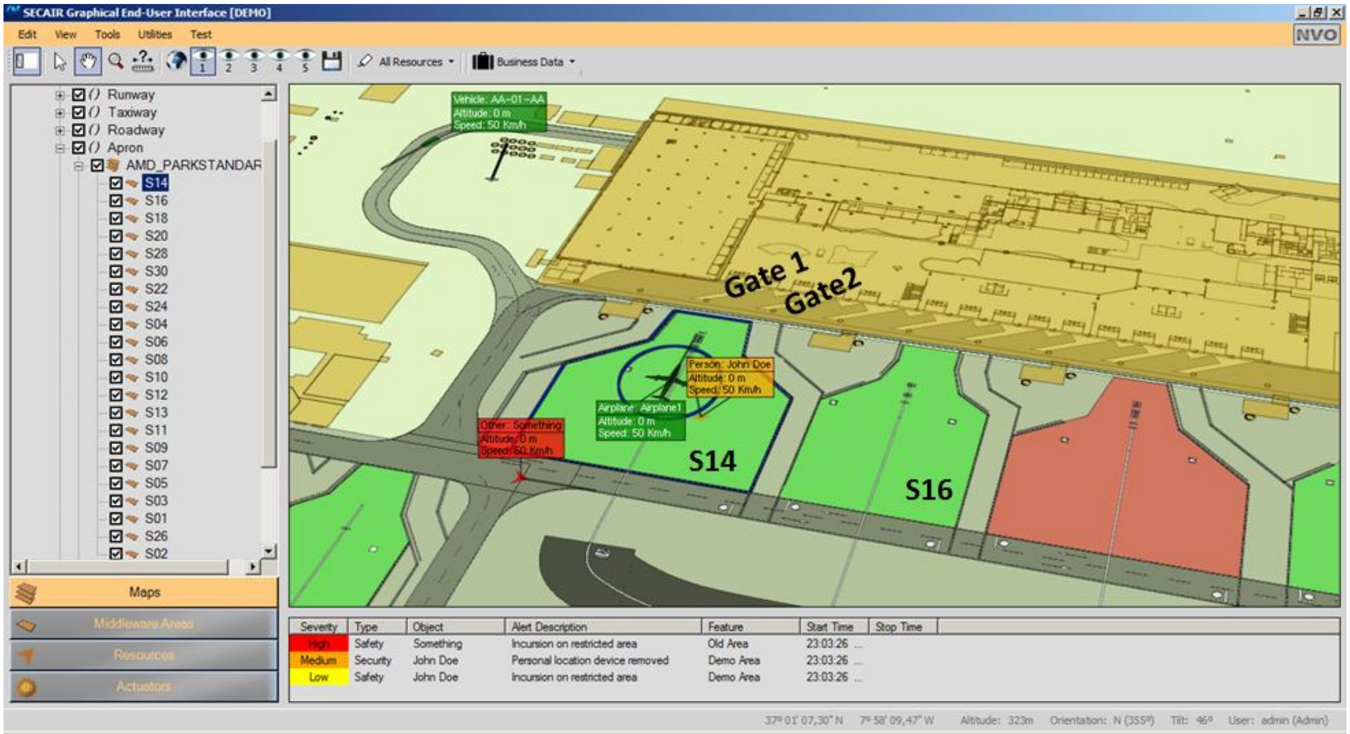
As presented in Figure 5, after a successful login to the SECAIR system, each connected client visualizes the current status of the airport layout as a collection of overlapped themes, each one representing a specific operational area within the airport environment. These themes are managed by the Map viewer, forming the background context over which the observed objects are represented as point features. All thematic layers were provided by the airport authority in a standard format as shape files.

The selection of the "Maps" option at the sidebar, enables the end-user to dynamically access each feature within a specific layer. For instance, to select a feature (e.g., Stand 14) from the AMD_PARKSTANDAREA layer, which is one of the layers used to accurately represent the apron layout, the user just has to select the corresponding polygon at the Map viewer (or navigate to the corresponding layer at the sidebar and select the feature he is interested in). As presented in Figure 5, the apron folder is a logical folder used to semantically group layers; therefore, it is possible to group layers with different geometries (e.g., points, lines and polygons) in the same folder.

Some layers are used just to help end-users to get contextualized with the airport area they are visualizing. These layers are not used by the Business Logic to compute domain events. At this phase and during the field tests only five layers were used to compute domain events:

- AMD_TERMINAL, layer representing the passenger terminal indoor areas,
- AMD_PARKSTANDAREA, layer representing the geometry of all stand areas. Figure 6 presents a print screen of the metadata defined for the layer and the metadata defined for stand 14;
- AMD_ APRONELEMENT, layer representing the geometry of the Apron area;
- AMD_ TWYELEMENT, layer representing the geometry of all taxiways;
- AMD_ SERVICEROAD, layer representing the geometry of all roadways.

When a feature is selected, besides visualizing the feature geometry at the Map viewer, the end-user can also access the metadata defined for the selected feature. Metadata common to all features are specified at the layer level, for instance, to indicate which categories of vehicles are allowed to circulate. These metadata might be duplicated if there is a need to include values for normal and low visibility operations. Metadata specific to a feature are specified individually to each feature within a layer, for instance, to cope with temporary changes to one feature or to cover business rules, which apply only to a specific feature or set of features.



Severity	Type	Object	Alert Description	Feature	Start Time	Stop Time
High	Safety	Something	Incursion on restricted area	Old Area	23:03:26 ...	
Medium	Security	John Doe	Personal location device removed	Demo Area	23:03:26 ...	
Low	Safety	John Doe	Incursion on restricted area	Demo Area	23:03:26 ...	

Fig. 5. Airport layout of the areas selected for the specified scenarios.

In SECAIR, most of the business rules related to airport operational areas have a strong spatial dependency with the metadata defined for polygon features, meaning that in a certain instant the position of the observed object (represented as a point feature) can trigger multiple events. When the GIS engine performs a topological point-in-polygon overlay operation, the current point position of the object is analysed against all intersected layers.

For each intersected layer, the Business Rule component will check if any of the specified business rules is infringed. Since the system operates with a collection of overlapped themes, classified accordingly to their relevance, when monitoring for business rule infringements the algorithm orders the resulting events according to their severity level. This means that, at the Presentation tier, domain events occurring in critical areas are visualized with a higher priority in relation to those occurring in less critical areas (e.g., a runway incursion is more severe than a stand area incursion). When multiple events are detected, the Map viewer only represents the most severe event. The other events are listed at the Alert viewer, enabling the end-user to have a good perception about the sequence of events caused by a specific person/vehicle.

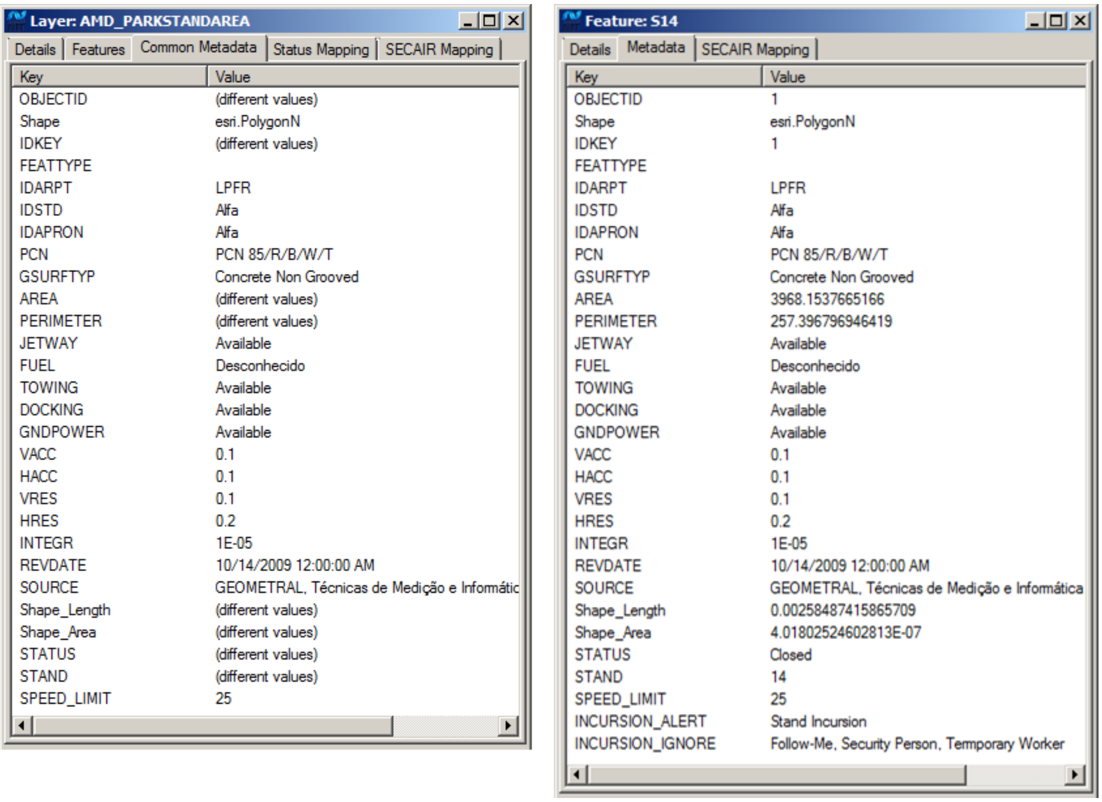
The validation of spatio-temporal events gets even more complicated because for each object movement the system has to perform a set of topological point-in-polygon computations to determine whether a given point in the plane lies inside, outside, or on the boundary of each intersected polygon features. If we consider that at each instance it is possible to have multiple objects and that a new position is triggered, even when a vehicle is not moving, it starts to be very demanding to comply with all the requirements.

For each event, the Business Logic uses the metadata provided by each layer for location-awareness purposes and formulation of the alert message to be visualized by the end-user at the Presentation tier.

The design approach was to provide, in a single screen, the user with all the data relevant for him to take informed decisions. Therefore, the Map viewer corresponds to a spatial dashboard where all observed objects are represented as colour coded point features and the airport layout as the background spatial context, which is used to semantically transmit implicit information about the status of some operational areas.

For instance, an occupied stand is represented with a green polygon, but if it is under maintenance, the same polygon is visualized in grey. In both cases, vehicles are allowed to circulate inside the stand area. In the first situation, vehicles are allowed to enter the stand area to assist a parked aircraft and, in the latter, because the stand area is deactivated. But when a stand is operational with a parked aircraft, as soon as the information about blocks-off is reported to the SECAIR system, the operational status of the stand area automatically changes to cleared, causing the colour of the polygon to change to red to signal a restricted access area where vehicles are not allowed to circulate or park.

Such visual clues transmit valuable insights to the end-user at the control centre. To simplify the identification of events, moving objects can be visualized with a colour coded label to easily transmit which objects are causing infringements from those operating as expected, i.e., labelled with a green colour. Once again, a colour code is used to differentiate less severe (labelled in yellow) from severe events (labelled in red).



Key	Value
OBJECTID	(different values)
Shape	esri.PolygonN
IDKEY	(different values)
FEATTYPE	
IDARPT	LPFR
IDSTD	Alfa
IDAPRON	Alfa
PCN	PCN 85/R/B/W/T
GSURFTYP	Concrete Non Grooved
AREA	(different values)
PERIMETER	(different values)
JETWAY	Available
FUEL	Desconhecido
TOWING	Available
DOCKING	Available
GNDPOWER	Available
VACC	0.1
HACC	0.1
VRES	0.1
HRES	0.2
INTEGR	1E-05
REVDATE	10/14/2009 12:00:00 AM
SOURCE	GEOMETRAL, Técnicas de Medição e Informátic
Shape_Length	(different values)
Shape_Area	(different values)
STATUS	(different values)
STAND	(different values)
SPEED_LIMIT	25

Key	Value
OBJECTID	1
Shape	esri.PolygonN
IDKEY	1
FEATTYPE	
IDARPT	LPFR
IDSTD	Alfa
IDAPRON	Alfa
PCN	PCN 85/R/B/W/T
GSURFTYP	Concrete Non Grooved
AREA	3968.1537665166
PERIMETER	257.396796946419
JETWAY	Available
FUEL	Desconhecido
TOWING	Available
DOCKING	Available
GNDPOWER	Available
VACC	0.1
HACC	0.1
VRES	0.1
HRES	0.2
INTEGR	1E-05
REVDATE	10/14/2009 12:00:00 AM
SOURCE	GEOMETRAL, Técnicas de Medição e Informática
Shape_Length	0.00258487415865709
Shape_Area	4.01802524602813E-07
STATUS	Closed
STAND	14
SPEED_LIMIT	25
INCURSION_ALERT	Stand Incursion
INCURSION_IGNORE	Follow-Me, Security Person, Temporary Worker

Fig. 6. Metadata assigned to the Layer level and metadata assigned to a specific feature within the layer.

All events are logged and represented at the Alert viewer with information about the event (e.g., severity, type, operational area, instant, etc.). This feature is useful to support decision makers in case there is a need to check for a specific event reported during the end-user shift. The reported information is automatically filtered based on the end-user profile, meaning that the list of events reported in the Alert viewer relate to resources from the airport stakeholder. Only the airport authority (e.g., Safety Manager) has access to all reported events.

The KPI viewer presents a set of predefined indicators specified for each of the operational scenarios defined in Table 2.

Table 2. Operational scenarios specified for the field test.

Num.	**Scenario Name**
Outdoor Safety (OSA)	
OSA.01	Surveillance of vehicle movements within a stand area
OSA.02	Collision avoidance support service
OSA.03	Aircraft ground movement tracking
OSA.04	Obstruction of an operational stand area
Outdoor Security (OSE)	
OSE.01	Detection of zone intrusion by unauthorized vehicle
OSE.02	Personnel tracking at the apron area
Indoor Safety (ISA)	
ISA.01	Working zone intrusion by unauthorized person
Indoor Security (ISE)	
ISE.01	Left luggage detection
ISE.02	Indoor-outdoor personnel tracking

The description of each operational scenario follows a template emphasizing relevant issues from the airport stakeholders' point of view. Besides a unique identifier with a semantic meaning for each scenario, the template also covers the following items:

- Name of the scenario, pointing out concerns from the perspective of airport stakeholders;
- Classification of the scenario, addressing environmental influences (indoor/outdoor) and type of events (Safety/Security);
- Technical constraints and a list of key indicators captured by the scenario to measure its impact or relevance;
- List of actions to be performed by each intervenient actor to test the specified scenario;
- Identification of the expected results for each scenario, defining the behaviour of the SECAIR system.

The net outcome of the SECAIR system will be the creation of improved or new services to increase security and safety of airport (or other critical infrastructures). In order to deliver such services, the SECAIR system needs to interoperate with the existing airport system to acquire business information relevant to properly manage domain events, namely data related to:

- Flight schedules, to align the system actions with the flight information scheduled for each stand, tasks allocated to assist the aircraft and reflect any last minute change to what was planned;
- Tasks assigned to staff, including ground handlers ;
- Staff, data relevant to support the system operational intelligence, including data about the airport operator he works to, role performed within the airport and specific data related to their airport driving licence;
- List of all resources (e.g., vehicles and equipment) operating at the airside area;
- Access to information about the main characteristics of the aircrafts operating at the airport.

VI. RELATEDWORK

The surveillance of surface movement events using ground surveillance is challenging for several reasons. First, because of the huge amount of demanding requirements that have to be validated to protect passengers, staff and critical infrastructure from serious security or safety infringements. Second, the quantity and quality of available surveillance is often poor. In an airport environment, unless the surveyed objects are equipped with a transponder, surveillance is based on surface movement radar (SMR) returns only. However, these solutions are extremely expensive to purchase and operate, and are subject to masking and distortion in the vicinity of airport buildings, terrain or plants [14].

The extensive deployment of satellite system and air-to-ground data links results in the emergence of complementary means and techniques. Among these, ADS-B (Automatic Dependent Surveillance-Broadcast) and MLAT (Multilateration) techniques are the most representative [15]. However, current radar based systems have many problems to track surface targets, especially in very dense traffic areas, such as the apron area. On the other hand, most aircrafts turn-off their transponders after landing, compromising their identification possibilities. This means that such systems will not detect non-cooperative vehicles or aircrafts that are not equipped with such a transponder. Therefore, there is a strong demand for a new sensing technology, in particular for smaller/medium airports. Such sensors include near-range radar networks, Mode 3/A, S or VHF multilateration, CCTV systems with video analytics [16], magnetic flux sensors or D-GPS installed in vehicles [17]. Although none of these sensors is individually able to meet all user requirements for airport surveillance, the fusion of the information they give can lead to an acceptable solution.

The Airport Surface Detection Equipment, Model X (ASDE-X) system [18], adopted in the United States, provides precise time-stamped position as required for its primary mission of improving situation awareness. ASDE-X surveillance is based on plot-level fusion of multiple complementary sensors, providing air traffic controllers with highly accurate, real-time position and identification information of all aircraft and vehicles on the airport surface. The system accepts and fuses primary SMR returns from objects on the airport surface. Any available report from ADS-B or secondary surveillance radar is also considered in estimating and reporting the fused track position. But the cost of this solution is high.

But, for the airport to function, it needs to have the right mix of these independent systems, and these systems need to cooperate with each other. At such level of interdependencies the airport system pursues a common goal for a set of stakeholders according to their concerns, which cannot be achieved by these entities individually. This means that an airport is a system-of-systems bounded by and understood through the identification and analysis of existing systems, the knowledge of operational procedures and regulations and organizational issues influencing the system of interest. In recent work [19], it is already possible to find algorithms addressing the very stringent integrity requirements to support complex event processing (CEP).

The CEP paradigm arose as a solution for critical environments, where there is a need to fuse a huge amount of information in order to detect events of interest in the company workflow [20]. Several proposals to fusion this company information have followed a CEP approach in many fields like financial transactions [21], business decisions [22] or RFID-based services [23]. The CEP paradigm has been used in the present work as the mechanism to orchestrate the contextual information inferred by other parts of the system. This means a new approach to aggregate and fusion of situation-awareness information about the observed objects within the organizational context in which they occur. Event-driven organizations are expected to react quickly to changes in their environment. Thus, their decision support systems should be driven by the same transactional events that keep the business operating. Such approach focuses on developing on-board applications that allow a vehicle to infer its role in a scene by taking as input, data from different sensors of the vehicle and other external data sources [24]. Such sensing can provide comprehensive information about the vehicle context, extending the surveillance capabilities of the system not only to perceive the situation context of a vehicle, but also the whole context related to its role within the environment where it operates.

In the literature, it is possible to find research projects (e.g., Airnet at www.airnet-project.com, 2004; ISMAEL at www.ismael-project.net, 2006; EMMA at www.dlr.de/emma, 2008; AAS at www.aas-project.eu, 2010, and LOCON at www.locon-eu.com, 2011), with different technologies that have been developed and successfully tested for ground movement detection, providing actionable data with a high degree of certainty or as a cost-effective solution.

Besides the identified technological related issues, the interactions between system components and between humans and software applications are also subject of interest within the academic community, in particular research on discrete event monitoring [25], [26]. These references consider safety requirements as control structures that restrict system behaviour at meta-model level. They propose a framework for interface control systems. In this framework, functional requirements and safety requirements are separately formalized as interface automata and controlling automata respectively.

According to their approach, requirements include two subtypes: functional requirements and safety requirements, which are requirements about the safe operation of the target system. There are two common causes of the changes to safety requirements. First, safety requirements may change at design-time. Second, safety requirements may change post-implementation. Some safety requirements are unknown before the system is developed and used in real environment. The requirement specification concerns that guided the design of SECAIR system in compliance with end-user functional requirements is in line with the methodology adopted in [27], to fill in the gap between the evolution of safety requirements and traditional verification process.

A similar approach is also conducted in [28], the motivation for this work emerges from multi-agent systems as a paradigm for developing complex, software intensive systems in real world scenarios. According to the multi-agent paradigm, agents are able to interact and to reply to events triggered by their environment. In this work, a detailed presentation of the Ptolemy II tool is presented to support the design and verification of resource constrained in embedded systems. The proposed solution allows the modelling of functional and dependability requirements separately. The functional model is described in terms of labelled interface automata, an action-oriented approach that considers not only the control flow, but also the information flow (input/output actions). Safety and security constraints are specified using controlling automata. It also applies model checking techniques in order to automatically generate a compliant model that will satisfy the dependability requirements.

Within the SECAIR project, one of the technical requirements is that all location–based technologies are coherently integrated using advanced data-fusion techniques in order to reduce installation costs and to address multipath effects reduction. The project fusion techniques operating with high-performance GNSS systems and improved radio based tracking, combined with video based technology will enable the accomplishment of an automatic and reliable prediction of events related to safety infringements. This broad level of integration extends the state-of-the-art for the surveillance of airport surface traffic, enabling unique automated decision support capabilities, with new context aware services that have not, thus far, been available.

Very concretely, the results of the SECAIR project might contribute to some other initiatives and product developments [29] [30] [31] [32] [33].

VII. CONCLUSION

The SECAIR system improves context awareness and controllability via the integration of localization technologies. By integrating multiple sensor technologies, the system delivers a comprehensive picture of ground operations, increasing the controller situational awareness and improving the airport safety.

In the SECAIR, multiple occurrences can be combined from different localization technologies to provide insight into business operations, enabling in-time decision making. Within the SECAIR project, one of the technical requirements is that all localization technologies are coherently integrated using data-fusion techniques in order to reduce installation costs and to address multipath effects reduction. The main objective is to develop new context aware services based on an innovative solution integrating high-performance RF tracking combined with video recognition technologies and mobility management in a middleware platform.

SECAIR is designed as heterogeneous sensor data fusion system architecture, covering the surveillance of non-cooperative resources and functionalities for continuous control of all ground movements within the apron area. A special attention is given to the environment of the system, in particular to information flowing from and to the system. The properties of the system components, as well as the relationships between them, are core elements for the analysis and design of the SECAIR architecture. The project takes lessons learned from previous projects in relation to techniques for multi-target, multi-sensor tracking, responding to very important security and safety issues in airport environments. The software components of the SECAIR system are being tested and a field trial is planned to take place at Faro airport during the last quarter of 2013, with a full evaluation of the results to be done during the first half of 2014.

ACKNOWLEDGEMENTS

SECAIR (ref. E6030) is an R&D project, partially funded under the EUROSTARS program. The authors acknowledge the collaboration of the remaining partners of the project. It is also acknowledged the funding of FCT through the PIDDAC program and of Agência de Inovação.

REFERENCES

[1] G. Pestana, A. Casaca, P. Reis, S. Heuchler, and J. Metter, "Management of Mobile Objects in an Airport Environment", Proceedings of the Second International Conference on Mobile Services, Resources, and Users – Mobility 2012, ISBN:978-1-61208-229-5, pp. 55-59, Venice, Italy, October 2012.

[2] Eurocontrol, "Operational Concept and Requirements for A-SMGCS Implementation Level 2. Ed. 2.1", 2010.

[3] G. Pestana, N. Duarte, I. Rebelo, and S. Couronné, "Adressing stakeholders coordination for airport efficiency and decision-support requirements". Journal of Aerospace Operations (JAO11), 2011.

[4] SECAIR Deliverable 1.1, "Definition of requirements and operational scenarios". Technical deliverable from SECAIR project, Eurostars Program, Brussels, December 2011.

[5] Luckham, David C., "Event Processing for Business: Organizing the Real-Time Enterprise". Hoboken, New Jersey: John Wiley & Sons, Inc., 2012.

[6] M. Ayres Jr. et al, "Safety Management Systems for Airports: Guidebook. Volume 2, ACRP REPORT 1, ISBN 978-0-309-11798-2, 2009.

[7] ISO/IEC/IEEE 42010:2011(E), "Systems and software engineering — Architecture description". 1st ed., Institute of Electrical and Electronics Engineers, Inc., 2011.

[8] W. Schuster, J. Bai, S. Feng, and W. Ochieng, "Integrity monitoring algorithms for airport surface movement", SpringerLink, vol. 16, N. 1, pp. 65-75, 2012.

[9] EUROCAE, "ED-119B: Interchange Standards For Terrain, Obstacle, And Aerodrome Mapping Data". EUROCAE ED-119B / RTCA DO-291B, 2011.

[10] M. Agaram and C. Liu, "An Engine-independent Framework for Business Rules Development". In Proc. of 15th IEEE International Enterprise Distributed Object Computing Conference. 2011.

[11] N. Subbotin, "Development of an Airport Ground Vehicle Runway Incursion Warning System". DOT/FAA/AR-11/26, 2011.

[12] IATA, "Airport Handling Manual (doc. AHM 630)", 30th ed., 20130.

[13] Eurocontrol, "A-SMGCS, Advanced - Surface Movement Guidance and Control System Manual", International Civil Aviation Organization, Approved by the Secretary General and published under his authority, Montreal, 2004.

[14] T. T. W. Hu, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors", IEEE Transacions on Systems Man, and Cybernetics-part C: Applications and Reviews, pages 334 - 351, 2004.

[15] A. Soto, P. Merino, and J. Valle, "ADS-B integration in the SESAR surface surveillance architecture", Proc. of Enhanced Surveillance of Aircraft and Vehicles (ESAV), pp. 13 -18, 2011.

[16] P. A. Vela, et al., "Visual Monitirong of Airport Ground Operactions", proc.of IEEE/AIAA 28th Digital Avionics Systems Conference, 2009.

[17] H. Gao, et al., "Safe airport operation based on innovative magnetic detector system", Institution of Engineering and Technology, Transp. Syst., Vol. 3, Iss. 2, pp. 236-244, 2009.

[18] E. D. Yuan and M. Watton, "An event-driven service oriented architecture for Space Command and Control decision making," 2012 IEEE Aerospace Conference, pp. 1–9, Mar. 2012.

[19] F. Terroso-Sáenz, M. Valdés-Vela, F. Campuzano, J. A. Botia, and A. F. Skarmeta-Gómez, "A complex event processing approach to perceive the vehicular context," Information Fusion, Sep. 2012.

[20] J. Dunkel, A. Fernández, R. Ortiz, and S. Ossowski, "Event-driven architecture for decision support in traffic management systems," Expert Systems with Applications, vol. 38, no. 6, pp. 6530–6539, Jun. 2011.

[21] K. Mani Chandy, "Event-Driven Applications: Costs, Benefits and Design Approaches". California Institute of Technology, 2006

[22] V. Pillac, C. Guéret, and A. L. Medaglia, "An event-driven optimization framework for dynamic vehicle routing". Decision Support Systems, vol. 54, no. 1, pp. 414–423, Dec. 2012.

[23] G. Cugola and A. Margara, "Complex event processing with T-REX," Journal of Systems and Software, vol. 85, no. 8, pp. 1709–1728, Aug. 2012.

[24] T. Waldron, S. Corporation, and E. Syracuse, "Detecting Airtport Surface Movements using Ground Surveillance", Digital Avionics Systems Conference, 2009. DASC '09. IEEE/AIAA 28th 2009.

[25] Zhe Chen and Gilles Motet, "Towards Better Support for the Evolution of Safety Requirements via the Model Monitoring Approach". In Proceedings of the ACM/IEEE 32nd International Conference on Software Engineering (ICSE 2010), pp. 219-222

[26] Zhe Chen and Gilles Motet, "System Safety Requirements as Control Structures". In Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference (COMPSAC), pp. 324-331, 2009.

[27] Gianina Homoceanu and Michaela Huhn, "Tool support for agent-based systems in Ptolemy". In Proceedings of the International Workshop on Security and Dependability for Resource Constrained Embedded Systems (S&D4RCES), 2011.

[28] Y. Liu, I. Gorton, and V. K. Lee, "The architecture of an event correlation service for adaptive middleware-based applications", Journal of Systems and Software no. 81, pp. 2134–2145, 2008.

[29] PPS-Locate and PPS-Airports, Brand of PPS GmbH, Nordostpark 93, 90144 Nuremberg, Germany, ® 2013.

[30] EUROCONTROL SWIM, part of the European ATM Master Plan, Contract between BIJO-DATA and European Organisation for the Safety of Air Navigation, Brusselles-Sesslach, June 2012.

[31] EUROCONTROL SESAR, Single European Sky ATM Research, Council Regulation (EC) No 1361/2008, on the establishment of a Joint Undertaking ("SJU") to develop the new generation European air traffic management system (SESAR), 2008.

[32] V. Mehta et al., "Decision Support Tools for the Tower Flight Data Mananger System", Integrated Communications, Navigation and Surveillance Conference (ICNS), pp. I4-1 to I4-12, 2011.

[33] J. Metter, "EFSUES - Energy efficiency increasing system for controlling and monitoring of airport apron", TNA XI-1/2012, SMI GmbH, Untere Burgbergstrasse 382, D-98663 Heldburg, Germany, Heldburg, 2012.