Journal of Space Safety Engineering 11 (2024) 80–86

Contents lists available at ScienceDirect

Journal of Space Safety Engineering

journal homepage: www.elsevier.com/locate/jsse

Improving air and space safety through enhanced coordination with the SpaceTracks Suite microservice architecture

Jens Hampe*, Anouk Stahnke

German Aerospace Center DLR, Lilienthalplatz 7, Braunschweig 38108, Germany

ARTICLE INFO

Article history:
Received 22 June 2023
Received in revised form 9 October 2023
Accepted 26 January 2024
Available online 9 February 2024

Keywords:
Launch und re-entry safety
Launch Coordination Center
Spaceflight integration
Software architecture
Air and space traffic management

ABSTRACT

Within the SpaceTracks project, a Launch Coordination Center (LCC) prototype is currently being developed by the German Aerospace Center (DLR). The aim of the LCC is to support the coordination among stakeholders of launches and re-entries before, during, and after the operation. Thereby, interests and needs of all stakeholders should be balanced and the situational awareness should be increased. At the core of the LCC is the SpaceTracks Suite (STS) microservice architecture.

When developing software solutions to integrate spacecraft into European airspace, various aspects must be considered: different space flight characteristics, the complex air traffic system and other concept requirements like security, scalability, flexibility, resilience and arbitrarily expandability, hence an agile procedure model and a loosely coupled and flexible software design is favored. This paper describes the DLR current approach, design considerations and solution characteristics of the STS.

© 2024 International Association for the Advancement of Space Safety. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Introduction

Spaceflight activities have increased over the past few years and even more frequent operations are expected for the future. Thereby, Launch and Re-entry Operators (LROs) have to coordinate with a multitude of affected stakeholders like the Network Manager (NM), Air Navigation Service Providers (ANSPs), or Maritime Authorities (MAs) to implement adequate safety measures. For example, Notice to Airmen (NOTAM) and Notice to Mariners (NOTMAR) are published in advance of a mission to prevent aircraft and vessels from entering potentially dangerous areas during the operation. Concerning this matter, two aspects must be mentioned. At first, the coordination between all involved actors can take a lot of effort and time due to the large number of actors involved as well as non-automated ways of working. Secondly, airlines and ANSPs can experience significant financial losses due to additional routes and therefore fuel costs as well as omitted route charges resulting from airspace closures of large spatial and temporal extent.

Therefore, when it comes to high frequented spaceflight activities on a regular basis in the future, advanced services and procedures to improve multi-stakeholder coordination and to better balance the interests and needs of space, aviation, and shipping would be desirable. For this reason, the German Aerospace Center (DLR) is currently developing a Launch Coordination Center (LCC) as part of the SpaceTracks project. The aim of the LCC is to ensure safe, efficient, and economic operations for all stakeholders involved in and affected by launch and re-entry activities by providing specific services and implementing adequate procedures for the pre-mission, execution, and post-mission phase of a mission.

A comparable system, called Space Data Integrator (SDI), is already being used as a prototype in the USA by the Federal Aviation Administration (FAA) [1]. The idea of the LCC is therefore to develop a system specifically in line with and against the background of the unique European challenges, that arise primarily from the union of a multitude of individual states.

Against this background, the objective of the SpaceTracks project, which started in June 2021, is to realize an LCC prototype and to demonstrate it within a pilot mission by the end of 2024. For this purpose, the prototype is established in DLR's Airport Control Center Simulator (ACCES) [2]. So far, a Concept of Operations (ConOps) and a Minimum Viable Product (MVP) have been defined considering requirements determined by stakeholder discussions. Further, the system is currently being implemented in software and hardware. In this regard, building an appropriate software architecture is essential. Therefore, the SpaceTracks Suite (STS) microservice architecture is developed to enable scalable, resilient, and flexible applications.

* Corresponding author.
E-mail address: jens.hampe@dlr.de (J. Hampe).

https://doi.org/10.1016/j.jsse.2024.01.005
2468-8967/© 2024 International Association for the Advancement of Space Safety. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

2. Launch Coordination Center Concept

To better understand the context and purpose of the STS architecture, the LCC concept is outlined first. This Launch Coordination Center Concept has first been published in [3].

Following the LCC concept, a spaceflight mission is divided into three phases: pre-mission, execution, and post-mission. For each phase, dedicated services are provided and procedures are established to support the seamless integration of air, space, and maritime traffic. Thereby, data from all domains is merged and processed within the LCC and results are distributed to affected stakeholders. In doing so, the aim is to achieve a high degree of automation and standardization. Further, the proposed system should be applicable for all kinds of spacecraft and missions.

Within this context, the current development in the project aims to create a prototype that can support launches and planned re-entries during nominal and non-nominal operations. Thereby, the latter refers to an on-trajectory explosion, for instance. Furthermore, it is also intended to use the system for unplanned re-entries in the long term. However, the current challenge is that the corresponding safety zones cannot be predicted with sufficient accuracy to provide useful recommendations for airspace closures.

2.1. Pre-mission phase

During the pre-mission phase, the LCC provides licensing support (1) to LROs by providing contact details and information about local requirements, (2) to LROs to conduct risk calculations to identify Hazard Areas (HAs), and (3) to Licensing Authorities (LAs) to verify specific criteria, like HAs. Thereby, to be able to provide the service to determine risks and HAs, a Risk Calculation (RC) tool is developed in-house within the SpaceTracks project. The calculations require, amongst others, spacecraft trajectory data, historical traffic data, and information about risk thresholds.

For specific missions, the planning statuses of the LRO and the Launch and Re-entry Site Operator (LRSO) are monitored by the LCC and shared with NM, ANSPs, and MAs to increase the situational awareness already during preparation. Further, data from various sources are used (1) for RC to determine HAs and identify affected ANSPs, (2) to predict the impact on the air traffic system, and (3) to derive recommendations, e.g. with regard to launch time windows. The LCC analysis results are then provided to and revised together with all involved parties to achieve a satisfactory outcome. Further, NOTAM and NOTMAR are published by the responsible authorities based on the HAs accordingly. This information can be considered by pilots for flight planning, for example. At the end of the planning process, relevant details for the mission are put into a briefing package shared with all stakeholders.

2.2. Execution phase

During the execution phase, the LCC provides the service to monitor the mission focusing on the interactions between the stakeholders. Thereby, during nominal and non-nominal operations, the aim is to enable real-time data and information exchange and to increase the situational awareness for all involved parties. For that purpose, (1) the mission status is monitored and shared, (2) the spacecraft trajectory and the surrounding air and maritime traffic are monitored, and (3) notifications are given, e.g. for cleared airspaces and for aircraft within HAs. Further, the LCC supports non-nominal procedures by determining and distributing a Refined Hazard Area (RHA) based on the last spacecraft state vector. The proposed procedure is as follows:

1. LCC receives last spacecraft state vector (from launch provider or mission control)
2. RHA is calculated by the RC module of the LCC
3. ATC is informed about RHA
4. ATC informs and reroutes affected air traffic

Thereby, the time available to inform ATC and clear RHAs depends on where the spacecraft is on its trajectory when the non-nominal event occurs.

In the future, a connection to inform pilots directly about hazardous events is also conceivable. However, this requires a precise definition of what actions a pilot may take in response to such an event on his own initiative, in coordination with the controller, or only after instruction by the controller. The clarification of such issues is the planned subject of future DLR projects, but the technical interface is already being considered in this context (cf. Section 4.2. EFB Client and SWIM Service).

2.3. Post-mission phase

During the post-mission phase, the LCC provides services (1) to determine the actual impact on the air traffic system and (2) to evaluate and compare planned and actual data, e.g. with regard to trajectory, HAs, and time schedule. Further, standardized stakeholder feedback is collected and analyzed to derive best practices and lessons learned for improving and optimizing the established procedures. The LCC findings are then shared with the stakeholders.

3. Methods

To reach the SpaceTracks project goals a six-folded method has been used. It consists of the following steps:

1. Analyses of the current system landscape
2. Designing a first concept of operation (ConOps)
3. Defining basic requirements to fulfill the ConOps needs
4. Selection of an appropriate flexible software engineering process
5. Defining a Minimum Viable Product (MVP)
6. Conduct workshops with future users to refine the user needs and requirements for the system design

3.1. Analysis, design, requirements

The results of the first three steps of the six-folded approach are expressed in chapter 2. It gives a very short overview over the main system requirements and the ConOps related topics. More detailed information can be found in [3]. Analyzing the current operational background and procedures of a system and generating a vision of a new operational concept are only the first steps of system development. When developing software solutions to integrate spacecraft into European airspace, various aspects must be considered, especially different space flight characteristics and the existing complex air traffic framework together with other functional and non-functional concept requirements.

As described in [3], many stakeholders in different positions are involved in and influenced by launch and re-entry operations and related aerospace activities. The solution context with the involved and influenced stakeholders is shown in the system scope and context diagram in Fig. 1.

The stakeholder involvement will be realized directly over the STS or indirectly via appropriate Application Programming Interfaces (APIs) to external systems. From the perspective of the LCC, the following relevant stakeholders and generic interfaces to external systems are identified:

- Core stakeholders
    - Airspace Users

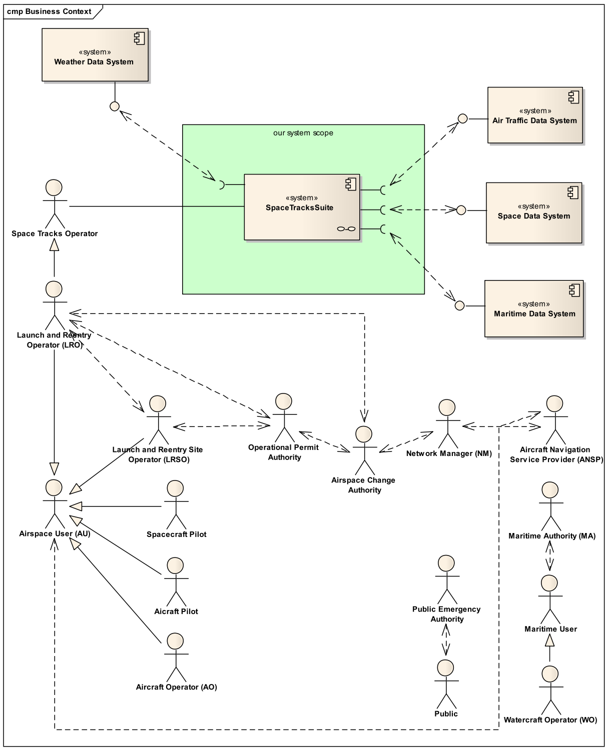

Fig. 1. SpaceTracks Suite context.

 - Air Navigation Service Providers
 - Network Manager
 - Launch and Re-entry Operator
 - Launch and Re-entry Site Operator (incl. Launch and Range Safety)
- Complementary stakeholders
 - Maritime Authorities (e.g. Marine Coastguard Agencies, Hydrographic Offices)
- Regulation Entities
 - Space Traffic Management Organizations
 - Airspace Change Authorities
 - Operational Permit Authorities
- External systems and interfaces
 - Air Traffic Data System (for air traffic related data)
 - Space Data System (for space vehicle related data)
 - Maritime Data System (for maritime traffic related data)
 - Weather Data System (for weather related data
 - Additional interface for pan European data exchange, as in [4]

ConOps, requirements and context are one side of the medal, designing a flexible and arbitrarily expandable software architecture is the other side of system design and realization of a vision, hence an agile procedure model combination of the Scrum and Twin Peaks approaches was chosen in the fourth step. Together with user conducted and future user workshops this approach facilitates a flexible process, open to further system changes and future project needs.

3.2. Scrum and twin peaks

Combining Scrum and the Twin Peaks model can provide a powerful framework for software development. Scrum is a popular agile methodology that emphasizes teamwork, collaboration, and the ability to respond quickly to changing requirements. The Twin Peaks model, on the other hand, focuses on managing risk and uncertainty regarding requirements and architecture.

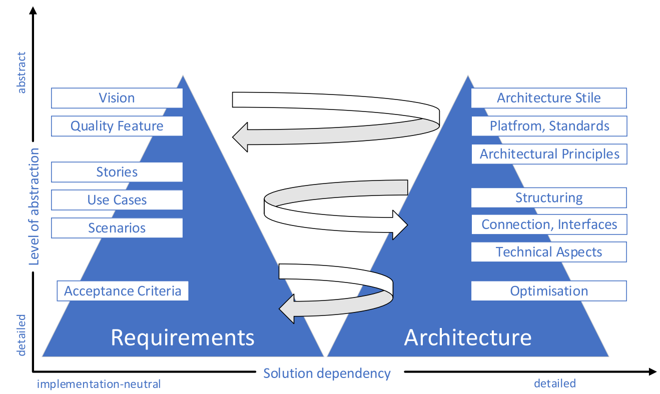

Fig. 2. Twin Peaks model.

In the Scrum methodology, the development process is broken down into sprints, which are short iterations that typically last two to four weeks. During each sprint, the team focuses on delivering a set of features that have been prioritized by the product owner. The team meets daily to discuss progress and identify any obstacles that may be hindering their progress [5].

The Twin Peaks model, on the other hand, focuses on managing the risks associated with software development [6]. The model is based on two peaks: the first peak represents the requirements associated with the development process, while the second peak represents the architecture associated with the product being developed. The requirements include issues such as vision, quality features, business context and acceptance criteria. The architecture development risks associated with the product being developed include issues such as architecture style and principles and additional technical aspects as depicted in Fig. 2 [7].

By combining Scrum and the Twin Peaks model, teams can identify and manage both the technical site and business requirements associated with software development.

3.3. Minimum Viable Product (MVP)

The Minimum Viable Product (MVP) [8] is a concept that is widely used in the world of entrepreneurship and product development. The idea behind an MVP is to create a product with the minimum features necessary to satisfy early customers and gather feedback for future product iterations. The concept can also be applied to scientific research, where the MVP is the minimum experiment or prototype necessary to test a hypothesis or idea.

The MVP approach in scientific research emphasizes the importance of iteration and adaptation. The first version of an experiment or prototype is not expected to be perfect, but rather to provide a foundation for future iterations. By gathering data and feedback from the MVP, researchers can refine and improve their hypotheses and prototypes, leading to more robust and validated results. The MVP approach can also be useful in fields such as engineering and technology, where the development of complex products can take years and require significant resources. By focusing on the minimum features necessary to satisfy early customers, companies can reduce the time and cost of product development and quickly bring products to market.

However, it is important to note that the MVP concept should not be used as an excuse for poor quality or lack of effort. While the MVP approach emphasizes efficiency and speed, it is still important to ensure that the minimum viable product is of sufficient quality to meet the needs of early customers and provide accurate data for future iterations.

4. Architecture solution

The LCC system is currently being implemented in software and hardware. In this regard, building an appropriate software architecture is essential. Therefore, the STS microservice architecture is developed to enable scalable, resilient, and flexible applications. This chapter describes the architecture solution. Therefore, a short overview of the used concept of microservices is initially provided. Following that, the STS microservice architecture solution is depicted within the context of the LCC concept, and lastly, an overview of the implementation of the architecture solution is given.

4.1. Microservice architecture

A microservice architecture is a software design pattern that structures an application as a collection of small, autonomous, and loosely coupled services. Each service focuses on a specific task or business capability, communicates with other services through APIs, and can be developed, deployed, and scaled independently.

The concept of microservices has gained popularity in recent years due to the benefits it offers over traditional monolithic architectures. Some of the advantages of microservice architecture are, as in [9] chapter 1, [10] chapter 4:

4.1.1. Scalability

With microservices, each service can be scaled independently, allowing the system to handle increased traffic and load more efficiently. This means that resources can be allocated to specific services that need them, improving performance and reducing costs.

4.1.2. Flexibility

Microservices allow for greater flexibility in the development process. Services can be developed and deployed independently, allowing developers to work on different parts of the system simultaneously, reducing the time to market, and enabling faster iterations and updates.

4.1.3. Resilience

By breaking the application down into autonomous services of the same type, microservice architecture improves the resilience of the system. If one service fails, the others can continue to function, preventing the entire system from crashing.

4.1.4. Technology heterogeneity

Microservices can be developed using different programming languages, frameworks, and tools, allowing teams to choose the best technology for each service. This approach reduces the risk of technology lock-in, increases innovation, and fosters a culture of experimentation.

4.1.5. Improved maintainability

With microservices, each service is independent, which means that developers can update and maintain it without affecting the rest of the system. This makes the codebase more modular and easier to manage, reducing the risk of bugs and errors.

4.1.6. Increased agility

Microservices allow for greater agility in the development process. Services can be deployed independently, reducing the time to market, and enabling faster iterations and updates. This approach helps organizations respond to changing customer needs and market trends more quickly and efficiently.

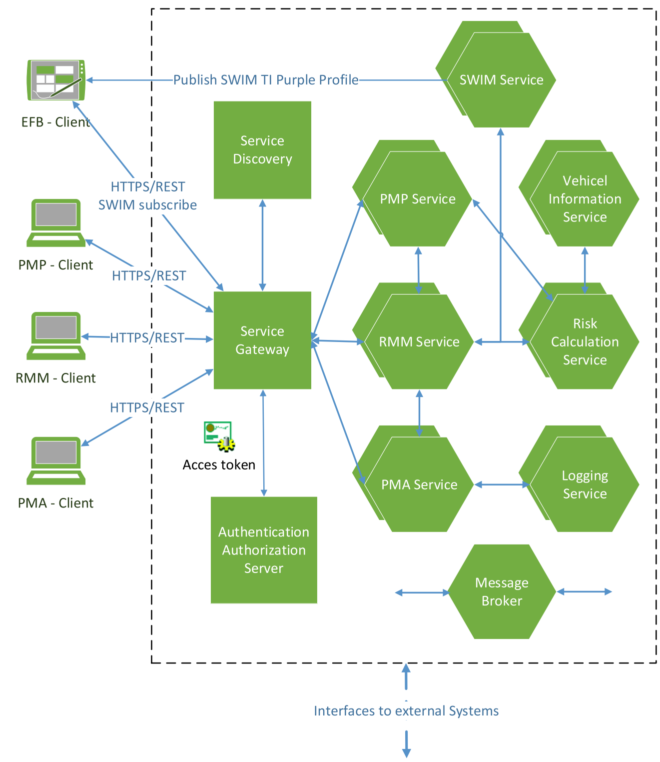

Fig. 3. SpaceTracks Suite architecture.

4.2. The SpaceTracks Suite architecture

This chapter describes the software architecture solution designed to address the challenges faced by developing an LCC prototype. The solution focuses on scalability, maintainability, and flexibility, enabling to quickly adapt to changing demands and technological advancements. The architecture solution is based on a microservices architecture, where applications are broken down into smaller, modular services that can be independently developed, tested, and deployed. Each service is designed to perform a specific function, and communication between services is achieved through lightweight protocols such as Representational State Transfer (REST) [11] or messaging over a message broker.

The overall STS architecture solution is depicted in Fig. 3. Not all data connections are shown for greater clarity. It consists of the following services:

4.2.1. Service gateway

The service gateway respectively the API gateway is the entry point of all microservices. It decouples the interface that clients see from the microservice implementation and add additional features like protection against threats, analysis and supervision, monitoring incoming and outgoing traffic, authentication and authorization and service discovery together with the corresponding services.

4.2.2. Authentication and authorization service

This service will offer means that only qualified users are able to access the STS. This service will use a token-based security framework like OAuth2 [12] for authorization. Authentication will be done with a third-party authentication service as identity provider. If the user successfully authenticates, they are presented with a token that must be sent with every request. Hence, providing single sign-on (SSO) identity and access management, ensuring that only authenticated users with the proper authorization can access.

4.2.3. Service discovery

This service will be responsible to find the appropriate service, respectively host in a distributed architecture. It is critical to microservices to enable horizontal scalability and resilience by adding more instances of a single service inside an environment. The service consumers are abstracted away from the physical location of the service. Hence, it enables application resiliency and horizontal scalability.

4.2.4. PMP service and PMP Client

The Pre-Mission Phase (PMP) Service will offer methods and coordination means to fulfill the specific planning tasks of a mission. The PMP Client is the dedicated human machine interface (HMI) that offers a graphical user interface (GUI) to conduct the appropriate tasks. The PMP services will be responsible for the following tasks:

- Import mission and vehicle specific data
- Provides licensing support
- Create and update missions as registered user
- Enable risk calculations to identify (HAs) by using the Risk Calculation Service and Vehicle Information Service
- Coordination with future licensing authorities
- Offer and share information about planning statuses to increase the situational awareness
- Offer methods to predict the impact on the air traffic system and offer recommendation for launch windows
- Generate a detailed briefing package for all stakeholders

4.2.5. RMM service and RMM Client

The Real-time Mission Monitor (RMM) Service will offer methods and coordination means to fulfill the specific tasks of the execution phase. The RMM Client is the dedicated HMI that offers a GUI to conduct the appropriate tasks of the real-time mission. The services are using the created, calculated and persisted data of the PMP services, the Risk Calculation Service and the Vehicle Information Service. The RMM services will be responsible for the following tasks:

- Enable general real-time monitoring and information exchange to evolved stakeholders
- Monitoring and sharing mission status over appropriate interfaces
- Monitoring of the spacecraft trajectory and the surrounding air and maritime traffic situation
- Display notifications for e.g. cleared airspace and for aircraft within HAs
- Support non-nominal procedures by determining and distributing RHA based on the last spacecraft state vector using the Risk Calculation Service and Vehicle Information Service

4.2.6. PMA service and PMA Client

The Post-mission Analysis (PMA) Service will offer methods to fulfill the specific data analysis tasks. The PMA Client is the dedicated HMI that offers a GUI to conduct the appropriate analysis tasks. The services are using the recorded and persisted data of the PMP services, the RMA services, the Risk Calculation Service, the Vehicle Information Service and the Logging Service. The services will be responsible for the following tasks:

- Fetching all recorded and persisted data of the relevant services
- Provide descriptive statistics of the missions
- Analyses of the actual impact on the air traffic system
- Evaluation and comparison of planned and actual data, e.g. with regard to trajectory, HAs, and time schedule.
- Collection of standardized stakeholder feedback
- Analyses to derive best practices and lessons learned and to improve and optimize the established procedures
- Provide means to share the findings with the stakeholders

4.2.7. Risk Calculation Service

The Risk Calculation Service is the main component to conduct all risk and impact related calculation of the pre-mission and real-time mission phases. Together with the internal core risk calculation application the service is able to:

- Conduct risk calculations
- Calculate all needed variants of HAs with respect to nominal and off-nominal events
- Calculate Instantaneous Impact Points (IIPs) respective areas with and without trajectory variations for stage/fairing separation (launch specific)
- Evaluate launch area and launch trajectory regarding hazard areas (safety) through STS risk model calculations
- Evaluate launch area and hazard areas regarding potential, general air traffic impact (optional maritime traffic) through STS risk model calculations
- Persisting all mission specific calculations and mission data

4.2.8. Vehicle Information Service

The Vehicle Information Service will be the source for all relevant space vehicle flight characteristics and needed spaceport characteristics. The information service offers methods to register and fetch the different space vehicle and spaceport data from an appropriate database.

4.2.9. EFB Client and SWIM Service

The EFB Client will be used to demonstrate how pilots could soon access a wide range of aeronautical, flight and hazard information in the cockpit via the System Wide Information Management (SWIM) infrastructure [13]. This SWIM Service will use the capabilities of existing electronic flight bags (EFB) to process information and provide it to the pilot independently of specific avionics system solutions on board the respective aircraft. The information that will be transmitted in the event of an acute hazard is displayed in the form of a cockpit alert and will contain the identified hazard area as well as all relevant information for coordinated hazard avoidance or minimization. The data transmission will be based on existing technologies and protocols, in particular the planned implementation of the SWIM Air/Ground purple profile [14].

4.2.10. Logging Service

The Logging Service will be the central service to log all relevant events of the STS. It will be the central service for monitoring and debugging by establishing distributed tracing following some core logging and tracing patterns like log correlation, log aggregation and microservice tracing as explained in [9] page 26. All data will be stored in a central database and can be used for the post-mission phase and future replay and simulation capabilities.

4.2.11. Interfaces to external systems

Dedicated interfaces will be provided for all involved stakeholders to gather and disseminate all relevant data, hence offer means for multi stakeholder coordination. The data exchange will be executed via a Message Oriented Middleware (MOM). The following conceptual and technical interfaces will be envisaged:

- Launch and Re-entry Operator (LRO)
- Launch and Re-entry Site Operator (LRSO)
- Operational Permit Authority
- EUROCONTROL (NM)
- ANSPs
- Maritime Authority (MA)
- Weather information service

- Air traffic information service
- Maritime traffic information service
- SWIM infrastructure

As the future Single European Sky ATM Research (SESAR) SWIM "Intranet for ATM" concept requests all the air traffic participants to act as the communicating sub-system, interfaces have to fulfill this ATM integration need as described in [15,16].

4.2.12. Message Broker

The Message Broker will be used to send and receive all messages between distributed microservices and the external interfaces. Therefore, the service will be able to realize different message patterns like the publish subscribe or the queuing message pattern. All services, interfaces and message brokers are establishing an LCC as an interface for the aggregation, processing, and distribution of relevant data for different stakeholders.

4.3. Implementation approach

Microservices can be developed using different programming languages, frameworks, and tools, allowing to choose the best technology for each service. This approach reduces the risk of technology lock-in, increases innovation, and fosters a culture of experimentation. The first SpaceTracks Suite services has been implemented with the Java Spring Boot Framework.

Java Spring Boot is a powerful and flexible framework for building Java-based web applications. It provides developers with a comprehensive set of tools and features that enable them to quickly and easily create robust, scalable, and highly maintainable applications. With its easy configuration, embedded servers, auto-configuration, and developer-focused features, Spring Boot is a great choice for developers who want to improve their productivity and create high-quality applications [17].

5. Discussion

The main goals of the SpaceTracks project are to make future space operations in European airspace flexible, dynamic and economical, and to increase the ability to respond to possible changes during operations, all while maintaining or improving the safety of all stakeholders involved in the mission.

Safety can be achieved through organization, rules and standards, coordination and information exchange, monitoring, best practices, and technology. The STS architecture solution serves the realization of the LCC concept and has the potential to maintaining or improving future space travel by licensing support, risk calculations, information sharing, multi-stakeholder coordination and cooperation in pre-mission, execution, and post-mission phase. It offers functionality to determine the flight risk for all different spacecraft and mission variants (e.g. air launch, vertical launch, suborbital flight) in real-time or a priori to a mission. Crucial for this are functions for fail-safe real-time data exchange in nominal and non-nominal operational situations in order to be able to react quickly and effectively to disruptive events.

The STS will contribute to achieve risk mitigation to an acceptable safety level by design and operations, partly covering human on-board and public safety, environmental protection and Space Traffic Management (STM) regarding the fields of space mission's safety as defined by the International Association for the Advancement of Space Safety (IAASS) Fig. 4, as in [18] page 267.

As stated in [18] page 266 safety refers to threats that are nonvoluntary in nature, while security refers to threats which are voluntary. Securing a microservices architecture is a complex and laborious task and involves multiple layers of protection. To achieve

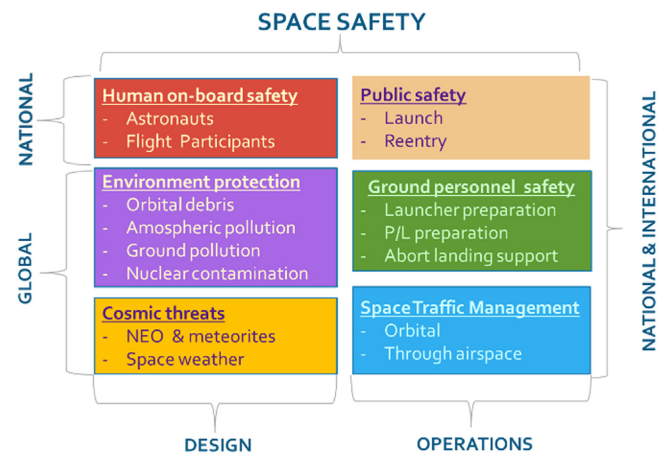

Fig. 4. Space safety fields – Credit: IAASS (International Association for the Advancement of Space Safety) https://commons.wikimedia.org/wiki/File:Space_Safety.png [Accessed 03 04 2023].

the necessary application layer security an authentication and authorization service will be a crucial component of the STS. In addition to that the STS will give respect down to the network layer by using an API gateway for all service calls and HTTPS/Secure Sockets Layer (SSL) for all service communications. Provide zones for services (for example, a public API and private API) and limiting the attack surface of the microservices by locking down unneeded network ports.

Analog to the FAA's SDI prototype the LCC respective STS will be an emerging technology element for future space operations in the European airspace. However, the actual use and operational implementation have not yet been determined in detail. Rather, the current project is about technology demonstration. Accordingly, no financing model has been defined for the time being.

Nevertheless, the services of the LCC are intended to be used by various stakeholders in the future. Thereby, the first application results from a cooperation between DLR and EUROCONTROL as part of the ECHO 2 project of the SESAR 3 research and innovation program for the digitalization of European skies. This collaboration is also based on DLR's contribution to the SESAR European Concept for Higher Airspace Operations ECHO ConOps [19]. The technical solutions incorporate conceptual and technical work by DLR in the area of real-time data exchange for launch and re-entry operations, as developed in the SpaceTracks project. The project therefore foresees the development and validation of a Network Real-time Mission Monitoring (N-RMM) module for Launch and Re-entry Operation within the European air traffic network.

6. Conclusions and outlook

The software architecture solution outlined in this paper provides a scalable, maintainable, and flexible approach for the LCC development by breaking down the LCC components into smaller, autonomous services, to achieve greater flexibility, scalability, resilience, maintainability, technology heterogeneity, and agility. By adopting a microservices architecture and incorporating future containerization, the LCC can quickly adapt to changing demands and technological advancements while maintaining the quality and reliability of their applications.

The combined use of Scrum and the Twin Peaks model provides a powerful framework for software development, by focusing on teamwork, collaboration, business requirements, and architecture. The MVP concept is a valuable approach for scientific research and product development, allowing researchers and entrepreneurs to quickly and efficiently test their ideas and gather feedback for future iterations. By focusing on the minimum amount of work necessary to test a hypothesis or develop a product, researchers and entrepreneurs can reduce the risk of investing too much time and resources into an idea that may not be viable.

The next steps as part of the development process of the LCC prototype are the implementation and integration into the DLR's ACCES for verification and validation purposes. Thereby, the flexible STS architecture solution makes it possible to validate different coordination scenarios. For example, with stakeholder agents in a local control room or with local distributed stakeholder agents, or a combination of both.

In addition, the LCC's real-time components will be further developed and modified within the ECHO 2 project to meet the needs of EUROCONTROL as Europe's Network Manager. This cooperation is crucial in order to lay the foundation for the operational use of a standardized system based on the LCC concept and architecture for the planning and monitoring of rocket launches in Europe.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Jens Hampe: Conceptualization, Methodology, Software, Writing – original draft, Visualization, Investigation, Writing – review & editing. **Anouk Stahnke:** Conceptualization, Methodology, Software, Writing – original draft, Visualization, Investigation, Writing – review & editing.

References

[1] R. Frodge, D. Murray, Space data integration, J. Space Saf. Eng. 9 (2) (2022) 182–188 no. VolumePages.

[2] R. Suikat, S. Kaltenhaeuser, J. Hampe, F. Timmermann, B. Weber, ACCES - A gaming and simulation platform for advanced airport operation concepts, in: Proceedings of the AIAA Modeling and Simulation Technologie Conference, Toronto, Ontario Canada, 2010.

[3] A. Stahnke, T. Rabus, S. Kaltenhaeuser, Supporting the safety and efficiency of airspace transition for launch and re-entry operations in Europe, in: Proceedings of the 2nd International Conference on Flight Vehicles, Aerothermodynamics and Re-entry Missions & Engineering (FAR), Heilbronn, Germany, 2022.

[4] S. Kaltenhaeuser, C. Kluenker, D.R. Schmitt, Interoperable data exchange for safe and efficient launch and re-entry operations in an international environment, in: Proceedings of the International Astronautical Congress, IAC, 2022 73rd International Astronautical Congress (IAC).

[5] t2informatik, "Scrum whitepaper. Alles Wichtige über Scrum auf einen Blick," [Online]. Available: https://t2informatik.de/downloads/scrum-whitepaper/. [Accessed 10 November 2022].

[6] t2informatik, "Twin Peaks model guide. Alles Wichtige über das Twin Peaks model auf einen Blick," [Online]. Available: https://t2informatik.de/downloads/twin-peaks-model-guide/. [Accessed 10 November 2022].

[7] S. Zoerner, Softwarearchitektur für alle!?, 2017. [Online]. Available: https://www.youtube.com/watch?v=4QAFlEPdcJI. [Accessed 10 November 2022].

[8] t2informatik, "MVP Guide. Alles wichtige über Minimal Viable Products auf einen Blick," [Online]. Available: https://t2informatik.de/downloads/mvp-guide/. [Accessed 10 November 2022].

[9] J. Carnell, I.H. Sánchez, Spring Microservices in Action, 2nd ed., Manning Publications Co., 2021.

[10] M. Richards, Software Architecture Patterns, 3rd ed., O'Reilly Media, Inc., Sebastopol, CA, 2017.

[11] R.T. Fielding, "Architectural styles and the design of network-based software architectures – Dissertation," 2000. [Online]. Available: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation_2up.pdf. [Accessed 31 March 2023].

[12] OAuth working group, "OAuth 2.0," 2023. [Online]. available: https://oauth.net/2/. [Accessed 23 March 2023].

[13] F. Morlang, J. Ferrand, R. Seker, Why a future commercial spacecraft must be able to SWIM, J. Space Saf. Eng. 4 (1) (2017) 5–8 Vols. Volume.

[14] Eurocontrol, "Solution PJ.17-01 — SWIM TI purple profile for air/ground advisory information sharing," 26 March 2023. [Online]. Available: https://www.atmmasterplan.eu/data/sesar_solutions/20431837. [Accessed 31 March 2023].

[15] F. Morlang, J. Hampe, S. Kaltenhaeuser, J. Jakobi, D.R. Schmitt, Commercial space transportation and air traffic insertion - SESAR requirements and the European perspective, in: Proceedings of the 1st Annual Space Traffic Management Conference, Daytona Beach, Embry-Riddle Aeronautical University, 2014.

[16] S. Kaltenhäuser, F. Morlang, T. Luchkova, J. Hampe and M. Sippel, "Facilitating sustainable commercial space transportation through an efficient integration into air traffic management," 1 December 2017. [Online]. Available: https://www.liebertpub.com/doi/full/10.1089/space.2017.0010. [Accessed 4 April 2023].

[17] The Spring team an community, „Spring," VMware, [Online]. available: https://spring.io/. [Zugriff am 5 April 2023].

[18] J. Pelton, T. Sgobba, M. Trujillo, Space safety, Handbook of Space Security: Policies, Applications and Programs, 2nd ed., Springer Nature Switzerland AG, 2020 S. Cham, Ed..

[19] ECHO project consortium, "ECHO project - European concept of higher airspace operations," [Online]. Available: https://higherairspace.eu/echo-project/. [Accessed 3 April 2023].