D2.1 Data governance, acquisition, sharing and access processes

Security of Critical Infrastructure by Multi-Modal Dynamic Sensing and AI

01 May 2026

ITEA Project No: 22006

sintra-ai.eu

DOCUMENT VERSIONS

Version no	Date	Authors	Changes
1.0	26.11.2025	Farhad Aghili	Initial Draft, Document Structure
2.0	22.2.2026	Jussi Simola	Content for the structure
3.0	18.6.2026	Farhad Aghili	Final version for review
3.1	24.6.2026	Aylin Yorulmaz	Review notes
3.2	26.6.2026	İsmail Uzun	Reviewed
4.0	29.6.2026	Farhad Aghili	Submitted version

Deliverable review procedure:

- **3 weeks before due date**: deliverable leader sends deliverable –approved by WP leader– to Project Manager.
- **Upfront** PM assigns a co-reviewer to cross check the deliverable.
- **1 week before due date**: co-reviewer provides input to deliverable leader.
- **Due date:** deliverable leader sends the final version of the deliverable to PM.

CONTENTS

1 ACRONYMS

Acronym	Expanded
ABAC	Attribute-Based Access Control
AI	Artificial Intelligence
API	Application Programming Interface
AWS	Amazon Web Services
BHS	Baggage Handling System
BLE	Bluetooth Low Energy
BMS	Building Management System
CAB	Change Advisory Board
CASB	Cloud Access Security Broker
CRA	Cyber Resilience Act
DEM	Digital Experience Monitoring
DPIA	Data Protection Impact Assessment
DSAR	Data Subject Access Request
ELK	Elasticsearch, Logstash, Kibana
ERP	Enterprise Resource Planning
ETL	Extract, Transform, Load
EXIF	Exchangeable Image File Format
FIDS	Flight Information Display System
GDPR	General Data Protection Regulation
GIS	Geographic Information System
GLB	GL Transmission Format Binary
glTF	GL Transmission Format
GNSS	Global Navigation Satellite System
HLS	HTTP Live Streaming

HSTS	HTTP Strict Transport Security
HTTPS	Hypertext Transfer Protocol Secure
IAM	Identity and Access Management
IoT	Internet of Things
JSON	JavaScript Object Notation
JWT	JSON Web Token
KMZ	Keyhole Markup Language Zipped
KVKK	Turkish Personal Data Protection Law
LPWAN	Low-Power Wide Area Network
LTE-M	Long Term Evolution for Machines
METAR	Meteorological Aerodrome Report
MFA	Multi-Factor Authentication
MQTT	Message Queuing Telemetry Transport
NATS	Neural Autonomic Transport System (messaging system)
NIS2	Network and Information Security Directive 2
NVR	Network Video Recorder
OIDC	OpenID Connect
OPA	Open Policy Agent
PII	Personally Identifiable Information
PKI	Public Key Infrastructure
RBAC	Role-Based Access Control
REST	Representational State Transfer
ROI	Region of Interest
S3	Simple Storage Service (AWS)
SAML	Security Assertion Markup Language
SASE	Secure Access Service Edge
SDK	Software Development Kit

SIEM	Security Information and Event Management
SMS	Short Message Service
SSO	Single Sign-On
TAF	Terminal Aerodrome Forecast
TCP/IP	Transmission Control Protocol / Internet Protocol
TOTP	Time-Based One-Time Password
UAV	Unmanned Aerial Vehicle
UCON	Usage Control
UDP	User Datagram Protocol
VPN	Virtual Private Network
VPC	Virtual Private Cloud
WebRTC	Web Real-Time Communication
WSS	WebSocket Secure
XML	Extensible Markup Language
ZTNA	Zero Trust Network Access

2 TABLE OF FIGURES

3 INTRODUCTION

EU cybersecurity regulations, especially GDPR, create requirements for all data handling in the SINTRA project. These requirements are based on trust, security, and accountability. Devices, software, information gathering, handling, analysis, and sharing must comply with the applicable requirements.

These requirements define who can access and use the data, and under which conditions. They help ensure that the data is correct. Data governance protocols consist of the policies, procedures, roles, and standards that an organization uses to manage data throughout its lifecycle, ensuring its quality and compliance.

The core element of the governance framework is a risk-driven approach. This guides all procedures in the research activities and in the proposed multimodal sensor platform within the AI-aided system environment. Data governance mechanisms and techniques have been developed with the involvement of SINTRA stakeholders.

This document is a comprehensive report on the SINTRA platform, with a focus on data governance considerations for human and technical actions such as AI-aided and multi-modal sensing systems. It explains the appropriate protocols, what must be considered in data handling, and how information should be shared, including the entire data processing cycle from acquisition to use. It presents the crucial protocols that are used, as well as customer requirements, expectations, and needs related to those protocols.

Work Package 2 focuses on the development and implementation of the platform, ensuring the secure and ethical handling of data, identifying and patching vulnerabilities, and protecting against cyber threats. The platform will be developed using an agile methodology, open-source tools, and custom-built components, with a microservices architecture for flexibility and scalability. The platform is implemented using a modular microservices architecture to ensure scalability, maintainability, and interoperability across participating organizations.

Key Objectives

In alignment with the goals of SINTRA and the activities of Work Package 2, this deliverable defines the following key objectives related to data governance, acquisition, sharing, and access processes:

- Establish a secure, privacy-compliant framework for data governance, acquisition, processing, analysis, and sharing.

- Develop and apply state-of-the-art cybersecurity measures across the platform lifecycle.
- Strengthen interoperability across partner systems through a unified governance model.
- Implement risk-driven methods that guide both technical and organizational decision making.
- Ensure adherence to applicable EU-level and national regulatory requirements, including GDPR, the AI Act, and NIS2.
- Support collaboration between consortium partners through harmonized governance and security practices.

3.1 Purpose of the Deliverable

This deliverable combines the outcomes of Tasks 2.1 and 2.4 into one coherent document. The following tasks influenced the content of the document.

Task 2.1 Data Governance Protocol

Task 2.1 establishes the foundational requirements for the SINTRA platform's data governance protocol. It consolidates regulatory requirements, stakeholder expectations, and risk-driven design principles into a coherent governance model that supports trustworthy, secure, and accountable data management throughout the platform.

Task 2.2 Ethical Data Handling

Task 2.2 defines general data handling principles that support ethical and responsible processing within the project. These principles form the baseline upon which the more comprehensive governance model is built. The outcomes of Task 2.2 are documented in Deliverable D2.2.

Task 2.3 Cybersecurity Measures

Task 2.3 focuses on the identification, analysis, and mitigation of cybersecurity risks. The requirements established under Task 2.1 inform the development of secure architectures, risk mitigation strategies, threat analysis methodologies, and operational cybersecurity controls applied throughout the platform.

Task 2.4 Access Control, Data Acquisition, and Privacy Protection Framework

Task 2.4 expands on the governance requirements by specifying mechanisms for secure access control, trustworthy data acquisition, privacy-preserving data sharing, and inter-organizational policy enforcement. It operationalizes several of the principles established in Task 2.1, ensuring their consistent implementation across technologies and partner environments.

3.2 Relation to Other Work Packages

Work Package 2, covering data governance, data sharing, security, privacy protection, and ethics, provides the foundation for Work Package 3, which focuses on multi-modal trustworthy AI analysis for anomalies, threats, and crime detection, and for Work Package 4, which addresses cross-coordination, visualization, and demonstrators.

3.3 Structure of the Document

In this deliverable, following the introduction, Chapter 4 presents the data governance framework and associated governance principles. Subsequent chapters describe the processes for data acquisition, data sharing, access control, privacy protection, and use-case-specific applications within the SINTRA platform.

4 DATA GOVERNANCE

The SINTRA project adopts a structured data governance framework to ensure secure, ethical, and compliant handling of multimodal data across all participating partners. This framework defines how data is accessed, processed, shared, and retained throughout its lifecycle. It is aligned with applicable European regulations and is implemented through a combination of organisational policies, technical mechanisms, and partner-specific practices.

4.1 Data governance principles

The SINTRA data governance principles define the key aspects that guide how data is managed, accessed, and protected across the platform. These principles ensure consistency, security, and compliance throughout the data lifecycle.

Data governance aspect	Definitions (SINTRA context)
Data access	Controlled and role-based access to data depending on user roles, use-case requirements, and security classification.
Data retention	Data is stored only as long as necessary for project objectives, in compliance with GDPR and project-specific agreements.
Data Integrity	Mechanisms ensuring that data remains accurate, consistent, and protected from unauthorized modification.
Data Security	Protection of data through encryption, access control, and secure infrastructure.
Data Compliance*	Alignment with regulatory frameworks such as GDPR, AI Act, and NIS2.

*The detailed legal and regulatory framework, including GDPR-related requirements and data protection considerations, has been extensively addressed in Deliverable D1.1. To avoid duplication, this section focuses on the governance principles relevant to data handling within the SINTRA platform, while referring to D1.1 for the complete legal and compliance analysis.

4.2 Data Access per Use Case

Use-case	Explanation (Data Governance Contribution)
UC 1 – Airport use case	1) Authentication and Session Management With centralized authentication (corporate directory/SSO integration), user logins are managed from a single point. Session security: timeout, multi-session control, additional device/location based controls (as required). 2) Authorization: Role and Responsibility-Based Access

	RBAC: AOCC Operations Manager, Terminal Operation, Apron/Ramp, Security, Facility, Regulation, etc. module/display/operation authorizations by role. Least privilege and authority hierarchy/segregation. 3) Scope-Based Access: Data Space / Operational Context Location/area coverage (terminal, pier/gate, stand, airside/landside), institution/organization scope (multi-tenant/multi-stakeholder), time coverage (Current / Next 3h / Next 24h views). Authorization is narrowed not only at the role level, but also by what operational context the user is working in. In addition to role information, each user's token carries scope claims (location, organization, time); Services automatically filter these scopes for each query. 4) Data-Level Protection: Area/Area-Group Masking Column/field-based authorization: even within the same screen, some fields (contact name, identifier, sensitive notes) can be closed to certain roles. Masking: some values may be shown partially/anonymously instead of fully displayed (regulatory and personal data scenarios). Access control goes beyond the module and row level to the column (area) level. The same record can be seen by multiple roles, but each role sees only the areas of that record that they are entitled to; the rest are hidden, masked or anonymized. 5) Access Channels: UI, API, and Integration Access Controls UI access: "view/create/update/close" permissions based on modules (Dashboard, Flight, PAX, Apron, CCTV, Incident, Chat, Settings). API access: key/certificate-based access, scope limiting, and rate limits for integrations. Service accounts are authorized by separate principles. The platform supports three different access channels: human users (UI), external integrations (API), and system-to-system service accounts. The authorization principles and control mechanisms of each channel are separated from each other.

	6) Audit and Traceability Who-what-when-accessed: audit records for data viewing and critical operations. Configuration changes (Settings/Alert rules): before/after the change, the person who made it, the timestamp, and the justification field. Incident management: Ownership → transaction history in the "Open in Progress → Resolved" feed. Every meaningful action on the platform — data viewing, data modification, configuration update, event lifecycle — is recorded in an immutable audit trail. Audit is the mandatory basic layer for both security investigation, regulatory compliance (KVKK/GDPR, ICAO/DGCA requirements) and operational responsibility.
UC 1.2 – Advanced F&B, Retail, Shop and Airport Safety and Security Solution (FBRSAS) Scenario	F&B (ARD) In the Food and Beverage (F&B) scenario, data governance focuses on the controlled management of data related to restaurants, cafés, and food court areas. Data access is limited to authorized Food & Beverage (F&B) operators, airport management, operations teams, and platform administrators. Users can only access data related to their assigned restaurant, café, food court, or operational responsibility. Commercially sensitive F&B data are protected through role-based access, masking, encryption, and secure data-sharing mechanisms. Data from sensors, and camera sources is timestamped, validated, and traceable across the processing pipeline. Raw or sensitive data is retained only for the required period, while aggregated indicators may be kept for planning, service improvement, and trend analysis. Access to dashboards, reports, configuration changes, and data-sharing actions is logged to ensure accountability and compliance with General Data Protection Regulation (GDPR), Turkish Personal Data Protection Law (KVKK), and SINTRA data governance rules. Koçtaş's contribution to data governance in the FBRSAS scenario is based on the principles of **privacy-**

	by-design and **data minimization**. In the retail use case, access to video sources is **strictly restricted to authorized components**, and analytics are performed only within predefined **operational regions of interest (ROI)**. As reflected in the implemented system architecture, raw video data is processed locally at the **edge layer within the store environment** and is not transmitted outside the premises. Instead, only **event-based, metadata-driven outputs** are shared with the SINTRA platform through secure and standardized interfaces. This approach ensures **controlled data access**, **traceability**, and compliance with **GDPR and KVKK regulations**, while also supporting interoperability across different use cases. Furthermore, the system does not process customer identities and focuses solely on behavioral analysis, with final decision-making remaining under human supervision, reinforcing ethical data usage principles.
UC 2 – Port use case – Netherlands, Belgium, Finland	Data access is restricted to explicitly authorized operators and AI systems. Multimodal data (UAV, sensors, cameras) is processed under controlled conditions, ensuring anonymization where required and secure sharing between stakeholders. All potentially sensitive data with regard to GDPR is only stored in protected on-premise storage and automatically deleted on a nightly basis.
UC 4 – Construction site use case	Access to the Sensolus asset-tracking subsystem is role-based and tenant-isolated. Tags and trackers carry a unique cryptographic identity; only non-personal asset metadata is shared with the SINTRA consortium for the agreed pilot sites, in line with the Sensolus privacy policy and GDPR.

4.3 Data policies and standards

The SINTRA platform follows a set of established policies and standards to ensure secure and compliant data governance across all components.

A key regulatory driver is the Cyber Resilience Act (CRA), which defines cybersecurity requirements for digital products and services. Within SINTRA, CRA principles are particularly relevant for IoT devices and connected sensor systems, ensuring secure design, vulnerability management, and lifecycle compliance.

Additional standards and frameworks are applied to ensure interoperability, security, and data protection across the platform.

Policy / Standard	Relevance to SINTRA
NIS2 Directive	Strengthens cybersecurity requirements for critical infrastructures.
Cyber Resilience Act (CRA)	Ensures security of connected devices and software components.
ISO 27001	Information security management practices.
GDPR	Ensures lawful processing of personal data and privacy protection.
AI Act	Governs the use of AI systems, especially for risk-sensitive applications.

4.4 Data quality and management

Ensuring high data quality is essential for the reliability of the SINTRA platform, particularly in environments where multimodal sensor data supports operational decision-making and AI-based analysis.

Key Requirement	Definition
Data validation at ingestion	Data is validated at the point of ingestion using schema checks and predefined quality constraints to ensure correctness and consistency before further processing.
Automated anomaly detection	Mechanisms are implemented to automatically detect anomalies in the data, including issues related to completeness, consistency, and outliers.
End-to-end lineage and provenance tracking	All data is traceable throughout its lifecycle, including documentation of data origin, transformation steps, applied algorithms, and involved system or service components.

Versioned datasets and transformations	Data and processing pipelines are versioned to ensure reproducibility of analytics and machine learning results over time.
Quality scoring frameworks	Data quality is continuously assessed using defined metrics such as accuracy, completeness, and timeliness, with results monitored and audited.

These requirements ensure that analytical outputs and AI models trained on platform data maintain reliability and traceability.

5 DATA ACQUISITION PROCESSES

5.1 General Overview of Data Acquisition

Trustworthy data acquisition is a foundational requirement for multi-partner AI-enabled platforms. Data provenance and lineage mechanisms capture information about data origin, transformations, and ownership throughout the data lifecycle, supporting accountability and auditability. Tools such as Apache Atlas and DataHub enable automated lineage tracking across heterogeneous systems. Provenance information is particularly valuable for AI model auditing and DPIA documentation, although integration with local systems and harmonization of metadata standards remain challenges.

Source attestation mechanisms complement provenance tracking by validating the authenticity and reliability of data providers. Trust scores can be adjusted dynamically based on historical behavior, validation outcomes, or detected anomalies. Technologies such as digital certificates, public key infrastructures, and distributed ledger solutions support immutable attestation records. This enables risk-aware ingestion decisions, provided that governance frameworks clearly define trust metrics and thresholds.

Automated quality and integrity checks ensure that ingested data conforms to expected formats, schemas, and completeness requirements. Tools such as Great Expectations and Amazon Deequ support rule-based validation and continuous monitoring. However, advanced threats such as data poisoning or steganography-based malware may bypass basic checks. Complementary safeguards, including anomaly detection and provenance analysis, are therefore required to establish a robust trust layer for downstream analytics.

5.2 Overview of Data Sources

Use case	Data Sources
UC 1 – Airport use case	The platform is continuously fed data by three main types of sources: ● Operational systems: TAMS, FIDS, BHS, and ERP. ● Infrastructure sensors: CCTV, Fire Alarm, and PFM sensors. ● External sources: Weather and Air Quality. The platform's value is generated by combining and exploiting the data from its standalone systems, making data ingestion the most important layer. The three main types of sources continuously feed the platform with the following kinds of data: **Operational Systems (TAMS, FIDS, BHS)**

	These systems feed real-time operational data into the platform. This data is crucial for real-time operations and includes specifics like flight updates, predictions, and pax data (passenger data). Furthermore, the platform utilizes sensitive information such as passenger security records and flight scheduling configurations for access control purposes. **Infrastructure Sensors (CCTV, Fire Alarm, PFM sensors)** These sensors continuously feed data into the platform. This includes CCTV (Closed-Circuit Television) data, Fire Alarm data, and PFM data.[1] External Sources (Weather, Air Quality sensors) These sources continuously feed external data into the platform. Specifically, this involves Weather data and Air Quality sensor data.
UC 1.2 – Advanced F&B, Retail, Shop and Airport Safety and Security Solution (FBRSAS) Scenario	In addition to data from operational systems, such as flight updates, we make use of weather data provided by the main system, as well as camera, and acoustic sensor data. In the FBRSAS scenario, the retail and food & beverage context primarily relies on video data originating from fixed CCTV cameras covering operationally critical indoor areas such as checkout and service zones. These video streams constitute the main data source for situational awareness and anomaly detection use cases. The data sources are conceptually limited to visual inputs required for operational monitoring. No additional personal, transactional, or customer-identifying data sources are considered within this scope. Video data is processed using predefined regions of interest (ROI) to support privacy-aware analytics, and downstream usage focuses on metadata-based event generation rather than raw multimedia distribution. This data source overview reflects a minimal and controlled input set, aligned with SINTRA's modular architecture and data governance principles.
UC 2 – Port use case – Netherlands, Belgium, Finland	**Belgium:** For this use case, data is gathered from UAVs and a UGV. For UAVs we use Citymesh's Safety Drone fleet consisting of 70+ Drone-in-a-Box (DiaBs) spread over Belgium. Only Citymesh itself has direct access, with limited access to the Belgian consortium partners for data collected at/on our data collection days. Citymesh acts as the data provider and processor where applicable, following our privacy policy, which meets and exceeds NIS2, GDPR, ISO27001 standards.

	The UGV used is an Ascento Guard, controlled remotely during the test. In addition, for the port use-case, data in the iSPECT platform is aggregated from UAV-based inspection workflows. Primary sources are high-resolution drone imagery with embedded EXIF metadata (GPS, camera parameters), photogrammetric 3D models (GLB/GLTF with XML camera alignment), georeferenced orthomosaic maps (KMZ), and real-time video streams. This data is a combination of raw data (imagery & EXIF data) and processed data (3D model, orthomosaic, AI annotations, …) All data is provided through our own capture methods and processed internally, never leaving the Skyebase organisation scope. All data is organisation-scoped from the point of ingestion, meaning every acquired object is immediately bound to a tenant context with enforced access boundaries. No data enters the system without prior authentication and authorisation verification. Data sharing consists of both raw data and processed data. In the case of the orthomosaic and 3D, post processing assures anonymous data. We both provide and process the data within our own scope. **Netherlands, Port of Moerdijk:** for this use case data is gathered from sensors and devices operated exclusively by the SINTRA project partners. Primary sources are high-resolution fixed and drone-based cameras with embedded EXIF metadata. All data is produced and consumed within a protected on premise LAN environment and not designed or planned to exit this environment. For multimodal analysis data is also consumed from (semi) public data sources, such as weather and P2000 data. Such data sources are, because they are already public, not considered for compliance and governance. **Finland**: Source attestation mechanisms complement provenance tracking by validating the authenticity and reliability of data providers. Trust scores can be adjusted dynamically based on historical behavior, validation outcomes, or detected anomalies. Technologies such as digital certificates, public key infrastructures, and distributed ledger solutions support immutable attestation records. This enables risk-aware ingestion decisions, provided that governance frameworks clearly define trust metrics and thresholds.

	University of Jyväskylä: The platform integrates data from a wide range of heterogeneous sources, including IP cameras, smart locks, microphones, vibration sensors, network nodes, and environmental monitoring systems. Weather-related data inputs include temperature, pressure, dew point, humidity, wind speed and direction, wind gusts, visibility, cloud cover, and sea water level. By combining and analysing data from these diverse systems, the platform enables comprehensive situational awareness. This makes robust data collection, integration, and analysis key functionalities for supporting reliable monitoring, anomaly detection, and decision-making processes.
UC 4 – Construction site use case	Sensolus BLE asset tags, stationary and vehicle-mounted BLE scanners, and GNSS / sensor telemetry from trackers. Tags emit no personally identifiable data. All data is processed in EU-resident cloud infrastructure.

5.3 Acquisition Methods and Pipelines

Data Ingestion

Data ingestion is the platform's most critical layer, as the main value is generated by combining and exploiting data from standalone systems. The ingestion process involves interfacing with data-generating systems using varying protocols and unifying the collected data for use within the platform. Data ingestion and transformation workflows are implemented using n8n and Bento, which handle essential functions such as extraction, normalization, enrichment (data joins), validation, and routing. For specialized or complex edge cases that lack existing adapters, a customized service called the "DataIntegrationService" is utilized.

Event Backbone (Pipeline)

The NATS Server functions as the high-speed event backbone, serving as the central nervous system for the platform. Its primary role is to trigger processing tasks and notify downstream listeners in real time. Events channeled through the backbone can initiate ETL pipelines, refresh data views, or trigger serverless functions.

Use case	Acquisition
UC 1 – Airport use case	Data ingestion is the most critical layer, as the platform's main value is generated by combining and exploiting data from standalone systems. • Ingestion Process: This involves interfacing with data-generating systems using varying protocols and unifying the data for use within the platform.

	• Workflows and Tools: Data ingestion and transformation workflows are implemented using n8n and Bento. These tools handle extraction, normalization, enrichment (data joins), validation, and routing. • Custom Service: A customized service, "DataIntegrationService," is utilized for edge cases that lack existing adapters. Event Backbone (Pipeline): The NATS Server functions as the high-speed event backbone (or central nervous system), which triggers processing tasks and notifies downstream listeners in real time. Incoming events can initiate ETL pipelines, refresh data views, or trigger serverless functions.
UC 1.2 – Advanced F&B, Retail, Shop and Airport Safety and Security Solution (FBRSAS) Scenario	In the FBRSAS scenario, data acquisition in the retail and food & beverage context is based on ingesting video streams obtained from fixed CCTV cameras deployed in operationally critical indoor areas such as checkout and service zones. Video streams are accessed via standard streaming interfaces and processed within controlled analytics components. The acquisition pipeline follows a modular and decoupled approach. Data ingestion is strictly limited to inputs required for situational awareness and anomaly detection, with processing constrained to predefined regions of interest (ROI). No bulk data collection or continuous storage of raw multimedia content is required within the scope of integration with the SINTRA platform. Downstream ingestion focuses on the generation of normalized, metadata-based events that can be securely transmitted to higher-level platform components. This acquisition approach supports scalability, interoperability, and privacy-by-design principles while remaining fully aligned with the SINTRA common architecture.
UC 2 – Port use case – Netherlands, Belgium, Finland	**Belgium:** Data is transmitted over a secured 5G link over our own 5G network to our own datacenter, where it is processed by our SENSE platform and made available to the necessary parties. This is mainly Citymesh itself, with limited access to the Belgian consortium partners for data collected at/on our data collection days. In addition, data enters the platform via two secure pipelines. The primary method uses time-limited presigned URLs: the authenticated

	client requests a short-lived upload token from the API, which validates the user's session and role-based permissions before issuing a direct-to-S3 signed PUT URL. Files are encrypted at rest with AES-256 server-side encryption. For files requiring server-side processing (e.g. KMZ parsing), data passes through the application layer where authentication is re-verified before storage. EXIF metadata is extracted and stored separately; no raw file metadata is exposed to other tenants. Image serving uses a secure proxy that validates access permissions per request and issues short-lived signed URLs (15-minute expiry) rather than exposing storage paths directly. Data follows an ORM structure and is thus standardized against our own internal system. Data transformation happens between post processing and provisioning within the system. Data sharing has no transformation compared to internal usage. These pipelines are not specific to SINTRA and are generally applicable within the iSPECT ecosystem. **Netherlands, Port of Moerdijk:** Data enters the platform internally via vendor-specific protocols and connections. These exist only within the physical and virtual security perimeters of the Port of Moerdijk harbour office network. Data transformation and continued processing is performed on the on-premise server. Only aggregated result information is securely exposed via the dashboard hosted via HTTPS on the same server. **Finland:** Automated quality and integrity checks ensure that ingested data conforms to expected formats, schemas, and completeness requirements. Tools such as Great Expectations and Amazon Deequ support rule-based validation and continuous monitoring. However, advanced threats such as data poisoning or steganography-based malware may bypass basic checks. Complementary safeguards, including anomaly detection and provenance analysis, are therefore required to establish a robust trust layer for downstream analytics.
UC 4 – Construction site use case	Sensolus's BLE tags broadcast cryptographically protected advertisements; scanners and trackers upload over NB-IoT on authenticated, encrypted channels, with signed firmware and per-device credentials. Cloud-side fusion combines scanner reports into asset locations. The data is accessible through a secured API.

5.4 Feasibility and Constraints

Use case	Feasibility and Constraints
UC 1 – Airport use case	The platform's design principles and tenancy model define its feasibility and constraints. **Feasibility/Guiding Principles (Design Goals):** ● Cost Efficiency ● High Resilience ● Operational Simplicity ● Fast Developer Iteration Cycles ● Flexibility and Integration The architecture is designed to incorporate heterogeneous systems, custom applications, and external AI models without forcing them into a rigid structure. ● Fault Tolerance: The system is built to function without a single point of failure, ensuring continuous operation. ● Scalability: The platform is designed to scale as data volumes and operational complexity grow, adapting to massive amounts of real-time data without performance degradation. **Constraints (Tenancy Model):** ● The SINTRA Platform is strictly defined as an exclusive Single-Tenant System. ● The architecture enforces a policy of one tenant per airport deployment. Each airport is provisioned with entirely Dedicated Infrastructure, meaning there are no shared components between different airports (e.g., separate Nats clusters, PostgreSQL databases, and MinIO storage buckets).
UC 1.2 – Advanced F&B, Retail, Shop and Airport Safety and Security Solution (FBRSAS) Scenario	The FBRSAS retail and food & beverage use case is considered feasible within the SINTRA framework, as it builds upon widely deployed CCTV infrastructures and leverages modular, event-driven analytics. The reliance on video-based inputs combined with metadata-oriented outputs enables system integration without the need for specialized sensors or tightly coupled hardware configurations. The primary constraints are associated with privacy requirements, regulatory compliance, and the operational heterogeneity of retail environments. These challenges are mitigated by restricting data processing to predefined regions of interest and by exchanging only

	event-level information with the SINTRA platform, rather than raw video streams. Although variations in camera positioning, lighting conditions, and store layouts may affect detection accuracy, they do not compromise the overall applicability or scalability of the proposed approach.
UC 2 – Port use case – Netherlands, Belgium, Finland	**Belgium:** This connection is built in proven technologies, aligned with the SINTRA access control model. The architecture enforces a multi-layered security model at every acquisition step: transport-level encryption (HTTPS with HSTS), application-level authentication (session-based with enterprise SSO), and resource-level authorisation (RBAC with EDITOR/INSPECTOR/VIEWER roles). These specifications and restrictions apply across the board, including iSPECT's own platform as well as the SINTRA data sharing setup. All data resides in AWS eu-west-1 for EU data residency compliance. Presigned upload URLs expire after one hour, limiting the window for potential token misuse. Server-mediated uploads are capped at 50 MB. The platform auto-scales horizontally (1-3 instances) with asynchronous AI processing decoupled via message queues, ensuring acquisition pipelines remain available under load without compromising security checks. Data sharing with 3rd parties or SINTRA partners is limited to the RBAC scope of their role, this is managed by Skyebase and guarantees limited access. This is further scoped down on a per-tenant basis. This is linked to the user's profile within iSPECT and drills further down into the permission scope by making sure there is a link between the organisation of the provided token and the organisation of the requested data. **Netherlands, Port of Moerdijk:** All data ingress/egress takes place using established industry standard protocols and practices. **Finland:** AI-enabled platforms introduce additional risks related to data acquisition, preprocessing, and model deployment that can compromise operational reliability and security. Locally collected datasets may suffer from incomplete coverage, missing features, or distributional idiosyncrasies, which can be amplified by AI models and propagate errors through shared outputs. These issues can result in overfitting, poor generalization, or inconsistent performance in heterogeneous environments. Data integrity threats, such as unintentional errors, mislabeling, or malicious manipulation (e.g., data poisoning), may degrade model performance or enable adversarial exploitation. Mitigation strategies include strict access controls for datasets, tamper-evident logging,

	automated dataset validation, anomaly detection, and periodic retraining under controlled conditions. Opaque preprocessing pipelines further increase operational risk by reducing reproducibility and traceability. Workflow management tools such as MLflow and Kubeflow Pipelines can enforce standardized, auditable preprocessing steps, ensuring that data transformations are documented and verifiable across partner organizations. Consistent adoption of such frameworks is essential to maintain model reliability and support regulatory compliance.
UC 4 – Construction site use case	Acquisition uses three complementary Sensolus scanner types: (i) battery-powered self-localising scanners installed ad-hoc by non-technical site personnel; (ii) externally-powered scanners with battery backup for high-cadence cases such as gates or trailers; and (iii) the Sensolus mobile app acting as a nomadic scanner on smartphones already present on the site. The approach builds on the existing Sensolus tracker platform and multi-tenant cloud back-end and requires no bespoke site IT. Data delivery from the Sensolus cloud to the SINTRA platform is built on proven technologies aligned with the SINTRA access control model. Constraints. Cost and battery require multi-year autonomy at hourly scan cadence. BLE signal reliability degrades inside metal containers and dense steel structures; this is mitigated by combining the three scanner types. Connectivity uses LPWAN (NB-IoT / LTE-M) The BLE-scanner is in scope of the EU CRA and EN 18031.

6 DATA SHARING PROCESSES

The SINTRA documentation provides a detailed view of information sharing across organisations, emphasising minimal external sharing, strict controls, and risk-based governance.

6.1 General Principles of Data Sharing

Key Principles

- **Minimisation of external data sharing** unless strictly necessary.
- **Strict exclusion of sensitive data** (e.g., video recordings, captured voices, personal data) from third-party access.
- **Processing of all data-sharing operations through DPIA-based rules**, ensuring full compliance.
- **Immediate reaction to abnormalities**, supported by audit-trail mechanisms.
- **Clear governance and documentation** for any inter-organizational data transfer.

These principles ensure that only governed, controlled, and legally justified data leaves the platform or moves between partners.

6.2 Data Sharing, Data Formats and Interfaces

UC 1 – Airport use case

Data Sharing and Export Controls

Report/output authority: data export (CSV/PDF, etc.) or bulk viewing authority is limited on a role-by-role basis. Fraud prevention: additional approval/limit/warning in cases such as large volume of queries, bulk downloads, access to sensitive screens.

Data export is one of the platform's highest-risk actions: data leaving the controlled environment can no longer be protected by platform policies. Therefore, export and bulk access are defined as a separate permission class from normal read privilege and are surrounded by additional layers of protection.

Report / Output Authorization (Export Authorization)

Important principle: Authorization to see a field does not imply authorization to export that field. Export is a separate permission.

Leave classes:

Permission	Description	Typical Roles
data:view	Viewing on the screen one by one	Operational roles
data:export:row	Single record PDF/print (operational)	Operational roles
data:export:bulk	Multi-record CSV/XLSX export	Admin, planning, reporting
data:export:sensitive	PII/regulatory coverage data included	Only Compliance / Op. Dir.
data:export:scheduled	Define a timed report	Admin, reporting roles
data:report:share	Share the report with another user/external email	Administrators (through the approval process)

Format and content control:

Masking is preserved. The masked area on the screen is also masked in export. Export is not the way to bypass masking.

Separate permission + justification + approval is mandatory for export that includes PII.

Formats are restricted. Some data can only be extracted as PDF (controlled); CSV/XLSX (reusable) requires additional permission.

Automatic username + timestamp + "Confidential / TAV" watermark is added to PDF/Excel printouts.

Classification, actor, exportId, correlationId are written to the metadata of the output → can be traced back to the source in case of leakage.

Operational role 1,000, manager 50,000, compliance unlimited + approval.

Share control:

- Out-of-system sharing (email, link) is limited to internal recipients; Additional approval is required for sharing to external email addresses.

- Link-based sharing is ephemeral and authority-controlled — there is no generic "anyone with the link" sharing.
- "Public dashboard" or anonymous access is only possible with aggregate / non-sensitive content, clearly marked.

Abuse / Insider Threat Prevention:

Even within the authorization, unusual access patterns are detected and restricted. Data export is protected by the trio of authorization control + behavior control + audit.

Speed and volume limits:

Per-user export rate limit is applied. Hourly/daily export number and total line quota.

Window-based quota exists. Sudden jumps are captured with a sliding window.

Concurrent export limits. The number of parallel export jobs for the same user is limited.

Expensive queries such as too large date range / large location set will be rejected or fragmented.

In case of exceeding the limit, a soft block (warning + approval) or hard block (rejection + audit) is applied.

Additional approval triggers (Step-up Approval)

The user is asked for justification + additional consent if:

Trigger	Action
Number of rows above threshold (e.g., 10,000 >)	Justification field required + step-up MFA
Export containing PII / sensitive area	Maker–checker (second authorized approval)
Out-of-hours access (night, weekend, holiday)	Banner warning + audit flag
Unexpected IP / location / device	Step-up MFA + additional approval
New user (first N days)	Quotas are lower, training mode

Export frequency exceeds baseline	Automatic notification to the administrator
Batch CCTV image/recording download	Dual approval + hourly usage restriction
Mass closure without reason for event closure	Blocked

Warnings and visible friction:

Summary screen before export to the user: "12,430 records contain 14 sensitive areas. Do you want to continue?"

Two-person integrity in a high-risk transaction: the user initiates the transaction, the administrator approves, then the output is generated.

Jobs that are pending approval appear in the user dashboard + admin queue.

Anomaly detection:

- A baseline of user-based behavior (weekly average exports, typical hours, frequently used modules) is extracted.
- Deviation signals: export at 03:00 at night, query 10x of normal, batch download from module that it never touched, → automatic alarm + temporary restriction.
- In multi-signal accumulation, the account is temporarily put in read-only mode; The solution is handed over to the security team.
- Anomalies are transmitted to SIEM; It is integrated with enterprise threat-hunting processes.

External leak prevention (DLP integration):

- Integration with enterprise DLP solution — sensitive content is blocked if it tries to leave the organization via email/cloud upload.
- On mobile/remote devices, the screenshot/print restriction (according to BYOD policy) can be checked.
- Separate limits for bulk access via API and separate audit (also applied to service accounts).

Audit and Traceability:

Each export is treated as an independent audit category; stored with richer content than standard audit records:


```
{
  "eventId": "exp-2026-04-27-...",
  "eventType": "data.export",
  "actor": { "id": "user-123", "roles": ["operations-manager"], "mfaVerified": true },
  "resource": "incident-report",
  "scope": { "location": "IST. T1", "dateRange": "2026-04-01..27" },
  "format": "xlsx",
  "rowCount": 8421,
  "fields": ["flightNo", "stand", "delay", "reason", "... maskedFields"],
  "containsPII": true,
  "justification": "Monthly compliance report — KVKK/DGCA",
  "approver": "user-456 (compliance-lead)",
  "fileHash": "sha256:abcd...",
  "watermarkId": "wm-9f3c...",
  "deliveredTo": "secure-share://report-...",
  "expiresAt": "2026-05-04T14:00:00Z"
}
```


- Which fields are exported are recorded (not just "report received", but which columns).
- File hash + watermark ID: A forensic trace traced back to the source in case of a leak.
- Access life: The generated report file itself is also timed — the share link becomes unavailable after a certain period of time.
- From the Compliance interface, the query "who received which report in the last 30 days?" is answered with a single click.

Operational Process Control:

- Data classification: Each dataset/field is marked with the tag public / internal / confidential / restricted.
- KVKK/GDPR compliance: Data subject request flow (DSAR) is supported as a separate process for personal data exports.
- Retention period: Generated reports are stored for a period of time, automatically deleted; retention is also within the scope of audit.
- Education and awareness: The user sees a short "data usage responsibility" reminder when starting export.

Architecture Flow:

```
User → Export Request ↓ [Permission control] data:export:* Is there any? ↓
  [Scope/Filter] row + area + apply masking ↓ [Volume/Anomaly control] limit / baseline
exceedance? ├── Generate directly → below threshold └── Justification →
above threshold + (if necessary) maker-checker approval ↓ [Production] watermark + file
```

```
hash + tagging ↓ [Delivery] controlled channel (signed URL, ephemeral) ↓ [Audit] enriched record + SIEM ↓ [Monitoring] anomaly algorithm, retrospective review
```

UC 1.2 – Advanced F&B, Retail, Shop and Airport Safety and Security Solution (FBRSAS) Scenario

The FBRSAS scenario is designed to address safety, security, and situational awareness requirements in semi-public commercial environments, such as retail stores and food & beverage areas. Within the retail domain, it emphasizes the monitoring of operationally important indoor areas by using video analytics and event-driven system integration. Within the F&B area, people-occupiable areas and anomalies in these areas are monitored and detected.

This use case follows a modular approach within the SINTRA framework, showing how specialized analytics tailored to a specific domain (e.g., retail) can still support broader safety and security goals across different sectors. Importantly, it achieves this without requiring tightly integrated or complex system dependencies, making the solution more flexible and scalable.

UC 2 – Port use case

Belgium Port use-case

In order to keep data sharing efficient, transparent, and scalable across multiple parties, a middleware layer is needed between each proprietary system and the central platform. This middleware has two main purposes. First, it decouples the shared data contract from each party's proprietary data structures, allowing data to be translated into a standardized format that supports consistency, validation, and interoperability without forcing changes to the underlying systems. Second, it provides a common authorisation boundary that is agnostic to any single first- or third-party implementation, enabling all participating systems to validate identity, permissions, and access scope in a consistent way without requiring deep custom integrations on each side. The central platform then acts as the common integration point: it can use validated party-issued tokens to visualise first-party data while also facilitating controlled access to third-party data. In addition, by acting as the middle layer between parties, the central platform reduces integration complexity to a single connection point and can use mechanisms such as caching and horizontal scaling to handle demand more efficiently.

<Proprietary structure -> per-integrator standardised middleware layer -> central platform standardised contract>

Data Type	Typical Proprietary / Native Format	Standardized Mapping in Central Model	Carrying Interface	Notes
Sensors (BLE / telemetry)	Vendor-specific device payloads, tag packets, polling responses, local DB records	Canonical JSON sensor object or `sensor.updated` event with fields such as `sensorId`, `position`, `batteryPct`, `status`, `tenantId`, `occurredAt`	REST API for query/history, WebSocket for live updates	Best suited for strong schema normalization because values are structured and low-bandwidth
Positional / operational metadata	Product-specific asset models, drone telemetry, camera state, GIS records	Canonical metadata/event envelope such as `metadata.position.updated` with `entityId`, `entityType`, coordinates, heading, timestamp	REST API and WebSocket	Good fit for shared event schema versioning and tenant/source attribution
Static reference data	Proprietary asset registries, zone definitions, configuration tables	Shared JSON resource shapes for assets, zones, source descriptors, integrator capabilities	REST API	Usually cached most safely because it changes less often than live telemetry
Video feed descriptors	Camera registry records, NVR metadata, stream	Shared video-source descriptor containing source ID, integrator ID, stream type, labels,	REST API	The descriptor is standardized even if the

	configuration objects	access mode, and capability metadata		actual stream technology varies
Recorded video / playback	Vendor playback endpoints, HLS/MP4 links, NVR session URLs	Standardized video resource metadata with URL/reference, source info, time bounds, and access policy	REST API returning signed URL or playback descriptor	The payload can be standardized without normalizing the underlying media container itself
Live video stream (IVS Real-Time)	Integrator-owned stage/session setup and AWS token request parameters	Standardized stream access contract: source descriptor plus token handoff payload for viewer subscription	Token-issuing API, then direct IVS SDK/WebRTC path	Important distinction: the access contract is standardized, but the media path is carried directly after token issuance
AI-annotated stream overlays / detections	AI platform-specific inference outputs, detection events, track metadata	Shared annotation/event schema with object IDs, labels, confidence, region, timestamp, source reference	REST API for history, WebSocket/ events for live detections	Usually modeled as metadata linked to a stream, not as a replacement for the stream itself

Alerts / rule outputs	Proprietary alarm/event systems, rule engine outputs	Canonical alert event envelope such as `alert.rule.triggered` or notification payloads	Event bus, queue, webhook, REST API	Better treated as a shared event contract than as raw subsystem output
Identity / access context	Product-specific user/session/role models	Standard OIDC/JWT claim contract: `iss`, `sub`, `aud`, `tenantId`, roles/scopes, expiry	OIDC flows, bearer token over API/WebSocket bootstrap	Not business data, but critical interface metadata for all sharing flows

Finland Port use-case

Data sharing across organizational boundaries requires mechanisms that go beyond static access control. Usage Control (UCON) models regulate not only initial access but also ongoing data usage, incorporating obligations and conditions during processing. Solutions such as NextLabs Control Center and Axiomatics Usage Control enable dynamic enforcement and are relevant where operational contexts or legal requirements evolve. However, their complexity and monitoring overhead must be carefully managed.

In the SINTRA project, dividing external and internal information sharing has been done in a way that minimizes external information sharing.

An audit trail method has been used to exclude unintentional sharing of information. Accurate records of the different stages of information processing minimize risks and reduce vulnerability. Abnormalities are reacted to immediately. Sensitive information is not shared to the 3rd parties, including video recording materials, captured voices, or personal information. Additionally, all data handling is processed based on the rules, which operations will be processed through DPIA analysis.

There is no need to gather the identified personal data for the external actors, despite the multimodal sensor network enabling it.

National data ecosystems and data spaces provide an overarching governance framework for structured data sharing among trusted actors. Inspired by initiatives such as GAIA-X, they emphasize interoperability, trust, and compliance. Supporting tools include metadata catalogs and anonymization services. Governance maturity is critical to ensure sustainability and scalability.

Machine-readable data-sharing policies further support automated compliance by encoding legal and contractual rules into enforceable logic. Standards such as XACML combined with Open Policy Agent reduce ambiguity and human error in policy interpretation. While national standardization remains challenging, such approaches improve interoperability and auditability.

Secure data sharing in port of Kemi use case is implemented by University of Jyväskylä (JYU) through the design and deployment of a dedicated layer between each proprietary system—such as IP cameras, smart locks, microphones, vibration sensors, network nodes, and environmental monitoring systems—and the AI platform. This fulfills two key functions. First, JYU has enabled the decoupling of system-specific data structures from the shared data environment, allowing heterogeneous data sources, including weather data (temperature, pressure, humidity, wind, and sea level), to be translated into standardized and harmonized formats when needed. This ensures data consistency, quality, validation, and interoperability without requiring changes to existing systems. Second, JYU has implemented a unified security and authorization boundary within the middleware, enabling consistent identity verification and access control. The AI platform, as designed by JYU, serves as the common integration point where validated and standardized data is aggregated, processed, and made available for analytics and AI-driven applications. It facilitates controlled access to both first-party and third-party data while ensuring data integrity and protection against unauthorized modification.

UC 3 – Railway use case

The same as the Belgium Port use-case.

UC 4 – Construction site use case

Sensolus and C-site follow the same proprietary-to-canonical mapping approach described for the Belgium Port use-case; the Sensolus cloud exposes asset and event data via a secured API aligned with the SINTRA access control model.

6.3 Subsystem-Specific Sharing Processes

UC 1 – Airport use case

There are preferred access channels in the platform design such as UI, API, and Integration Access Controls.

UI access: "view/create/update/close" permissions based on modules (Dashboard, Flight, PAX, Apron, CCTV, Incident, Chat, Settings). API access: key/certificate-based access, scope limiting, and rate limits for integrations. Service accounts are authorized by separate principles.

The platform supports three different access channels: human users (UI), external integrations (API), and system-to-system service accounts. The authorization principles and control mechanisms of each channel are separated from each other.

UI Access (Human Users)

Operational modules are dynamically rendered based on the user's role; Authorization control is applied at both the UI and backend layers (defense in depth).

Module × Transaction Authorization Matrix:

Module	View	Create	Update	Close/Delete	Typical Roles
Dashboard	✓	–	Widget editing	–	All operational roles
Flight	✓	–	Status/notes	–	Operation, Apron, Terminal
PAX (Passenger)	✓	–	Flow/density notes	–	Terminal, Operation Direction.
Apron / Ramp	✓	Assign a stand	Turnaround	–	Apron, Operation Direction.
CCTV	✓ (Authority-based)	–	–	–	Security, Operations Direction.
Incident	✓	✓	✓	Only Op. Dr. Dir.	All roles (own domain)

Chat	✓	✓ (message)	✓ (own)	–	All operational roles
Settings / Config	✓ (read-only)	✓	✓	✓	Admin Only
Threshold/Alert	✓	Module manager	Module manager	Op. Dr. Dir. + confirmation	Module owners
User & Role Mgmt	✓	✓	✓	✓	Admin Only
Audit Log	✓ (read-only)	–	–	– (immutable)	Compliance, Admin

- Each module × action pair is defined as a separate permission (flight:view, incident:close, settings:update...); role = collection of these permissions.
- The backend validates every request from the UI with the same permission definition; Hiding the button in the UI is not considered a security check.
- If there is no action authority, the UI component will not render at all or will be shown passive ("disabled + tooltip").
- Critical actions (incident close, threshold update, config change) are preceded by a confirmation dialog + justification field, some trigger step-up MFA.

API Access (System Integrations):

External system integrations (A-SMGCS, METAR/TAF, BMS/IoT, CCTV, FIDS/BIDS, source ERPs, etc.) It accesses through a completely separate channel from the UI, with the service account ID.

Identity & Key Management:

- A separate service account (machine identity) is defined in Keycloak for each integration.
- **OAuth 2.0 Client Credentials Flow** (default) — client_id + client_secret or mTLS certificate to receive tokens.

- **mTLS (mutual TLS):** Certificate-based mutual authentication in critical integrations (CCTV, A-SMGCS); certificates are signed from the corporate PKI.
- **Signed JWT / Private-Key JWT:** Ephemeral token generation that does not require key sharing.
- **API Key:** Only in low-risk, read-only integrations; always with scope and IP allowlist.
- No long-lived secrets are embedded in code or repos — they are injected through secret management (Vault / AWS Secrets Manager / Kubernetes secrets).
- Secret rotation: Service account credentials are automatically rotated periodically (e.g., 90 days) and at the time of the event.

Scope Limitation:

- Each service account can only access the subject/topic/endpoint set it needs (least privilege).
- **Module scope:** METAR service only weather:write, A-SMGCS only surveillance:write.
- **Direction scope:** producer / consumer role (cannot write read-only integration).
- **Source scope:** specific stream / topic / queue list.
- **Location scope:** the airport/terminal to which the integration belongs.
- NATS authorization: Each service account has its own NATS user; pub/sub permissions are limited by subject pattern (aocc.weather.>, aocc.surveillance.metar.>).
- Network control: IP allowlist; If possible, it connects via private network / VPN / VPC peering.

Speed Limits and Protection:

- API Gateway level rate limiting: request/second and request/minute limits on a service account basis.
- Burst allowance + sustained rate is defined separately.
- When the limit is exceeded, 429 Too Many Requests + Retry-After header.
- Quota: Daily/monthly call quotas (critical in the external provider chain such as METAR/TAF).
- Concurrency limit, payload size limit, timeout, circuit breaker, DDoS protection and anomaly detection are applied.

Service Account vs. Human User Separation:

System integration with the human user is authorized separately in principle; This distinction is critical for both security and control.

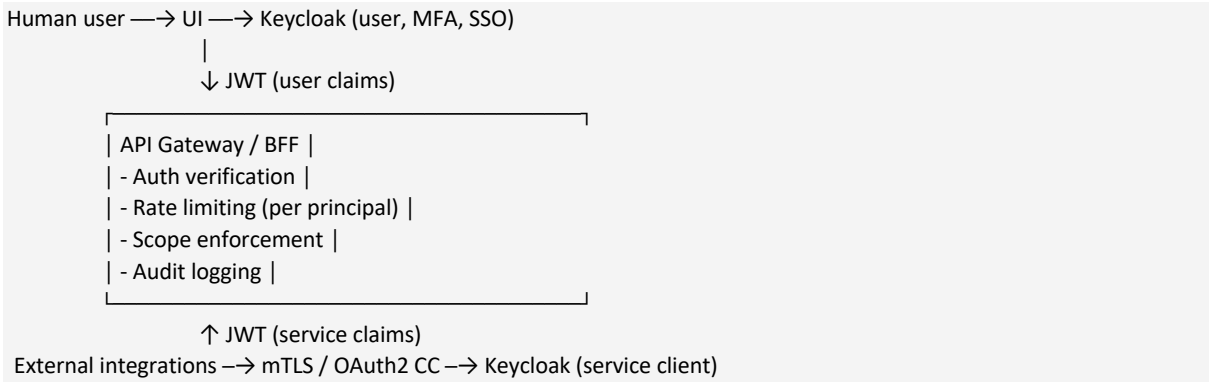
Size	Human User	Service Account
Identity source	Azure AD / Keycloak (federated)	Local client (machine identity) in Keycloak
Auth method	Username + password + MFA	Client credentials / mTLS / signed JWT
Token lifetime	Short access (5–15 min) + refresh	Very short access (5 min), no refresh
Ownership	Individual (depending on the person)	System/application based (owner team is assigned)
Scope of authority	Broad, role-based, UI actions	Narrow, scope-based, single-purpose
MFA / step-up	There is	No; instead of mTLS + IP allowlist
UI login	Yes	No — disabled
Audit tag	actor.type=user	actor.type=service
Life cycle	With HR/AD process	With integration onboarding process
Rotation / Review	Password policy, periodic review	Automatic secret rotation + scope review

Important principles:

- A service account never has a human role, and vice versa: human users are prevented from connecting to the API with client credentials.
- Service account sharing is prohibited — each integration uses its own account; The compromise blast radius remains narrow.
- "Shadow" integration barrier: Unauthorized/unregistered service account cannot be created; All Machine Identities are kept in the central inventory.

- Service account actions are always logged with the resource integration name + owner team; It is never confused with the human user.

Unified Architecture:



Belgium Port use-case

Sharing processes remain subsystem-specific at the proprietary layer, but are normalized through each integrator's communication layer. This means the central platform does not require every vendor or subsystem to adopt the same internal data model, storage pattern, or protocol. Instead, each party keeps its own subsystem unchanged and uses the middleware layer to translate proprietary structures into agreed external contracts. In practice, the exact format depends on the subsystem and data type: sensors may expose structured telemetry, metadata sources may provide positional or status updates, and video systems may expose stream references, descriptors, or token-based access flows. The important principle is that internal formats stay vendor-specific, while externally shared data is reshaped into standardized APIs, realtime event formats, and authorization-aware access patterns. This approach reduces coupling between parties and avoids deep point-to-point integrations. It also creates a controlled boundary where tenant mapping, permission mapping, validation, and filtering can take place before data reaches the central platform or another party. As a result, standardisation happens at the integration boundary rather than inside the subsystem itself. That makes it possible to support multiple product types and vendors without forcing redesign of their internal systems, while still ensuring that shared data is consistent enough to aggregate, visualize, and govern centrally.

Long-Term Interoperability

Long-term, products work together not because they share one internal implementation, but because they become compatible at the contract level. The communication layer provides that compatibility by aligning each subsystem to a stable external interface built around

shared identity, authorization, and data contracts. In this model, OIDC establishes trust between parties, claims and scopes determine what may be accessed, and standardized request/response and event schemas define how data is exchanged. Over time, this allows new products and integrators to be added with limited additional effort, since each new participant only needs to integrate against the shared contract rather than against every other system individually. This also gives the ecosystem a more sustainable path for growth. As products evolve, their internal schemas, APIs, and technologies may change, but the shared interface can remain stable and versioned. That means interoperability is preserved through controlled adaptation rather than rigid uniformity. In practical terms, the long-term goal is a federated model in which proprietary systems remain independent, but can still participate in a common operational picture through standardized middleware, common trust rules, and well-defined external data formats and interfaces.

Finland Port use-case

Security metrics provide evidence-based evaluation of access control effectiveness, data integrity, availability, and privacy safeguards. Metrics such as access violations, integrity incidents, availability indicators, and anomaly detection rates support continuous improvement and informed governance.

Validation is conducted in laboratory environments to enable controlled evaluation of access control, data sharing, privacy mechanisms, and monitoring under simulated threat scenarios. Validation focuses on feasibility, integration, and governance effectiveness rather than full-scale deployment.

Regulatory alignment is addressed through design-driven compliance with GDPR and DPIA requirements, embedding data minimization, purpose limitation, and accountability into platform architecture. The EU Artificial Intelligence Act further informs governance of AI-enabled components through risk management, traceability, robustness, and human oversight.

Network Security in Kemi Seaport

Modern port operations are critically dependent on integrated information systems that support operational continuity and security. Traditional security architectures, however, are increasingly inadequate due to legacy components, heterogeneous ICT environments, and evolving threat landscapes. Contemporary architectural paradigms—particularly Secure Access Service Edge (SASE) and its core component, Zero Trust Network Access (ZTNA)—offer a more robust and adaptive approach.

Basics of the SASE Architecture

Secure Access Service Edge architecture (SASE), introduced by Gartner in 2018, presents a technology-agnostic, identity-centric security framework composed of modular cybersecurity controls. In its foundational form, SASE integrates:

Internet security controls and Zero Trust Network Access (ZTNA). ZTNA is a modern architectural model that combines network-level security principles at the L3 level (IP) and DNS addresses in terms of masking the real IP address of the destination service, where the user can resolve the address of the destination service if the user has the right to do so according to the policy.

ZTNA is supported by additional functions such as Digital Experience Monitoring (DEM), Cloud/Remote Browser Isolation, Sandboxing, and Cloud Access Security Broker (CASB) capabilities. These components can be deployed with minimal disruption to existing infrastructures and allow granular, user-specific policy enforcement via continuous identity verification and AI-driven risk assessment.

Deployment of ZTNA in the RCGE environment

Within the RGCE environment, JAMK will deploy a ZTNA solution as part of the 2026 demonstration. The risk-driven solution reduces the risk to the port's critical business and IT services compared to traditional VPN services, enabling more secure use of services. One of the targets to be protected could also be the SINTRA platform. ZTNA enhances security relative to legacy VPN systems by abstracting real destination IP addresses, enforcing authentication through an Identity Provider, and applying individualized policy sets that strictly define permitted services. This architecture materially reduces the exposure surface of critical business systems, including potential targets such as the SINTRA platform.

Limitations of Traditional VPN Architectures

Conventional VPN solutions suffer from several structural and operational deficiencies:

- High susceptibility to denial of service attacks against VPN gateways,
- Architectural complexity accumulated over extended ICT lifecycles, acquisitions, and mergers,
- Performance and reliability issues caused by routing all traffic—including cloud services—through VPN tunnels,
- DNS inconsistencies resulting in degraded user experience, especially for latency-sensitive services (e.g., real-time collaboration tools).

These issues can produce systemic misalignment where VPN services are incorrectly perceived as the root cause of failures generated elsewhere in the architecture.

Architectural Remediation Through ZTNA

The transition towards a ZTNA-based architecture enables clear segmentation of business-critical traffic via a dedicated ZTNA path, typically implemented using application connectors deployed near protected services. This approach:

- Supports staged migration from legacy systems,
- Simplifies governance for Change Advisory Boards (CAB) through identity-level risk scoping,
- Allows controlled testing cycles (smoke tests, limited pilot groups),
- Reduces dependency on broad, high-risk VPN changes affecting thousands of users.

Proposal for the architecture

The SASE/ZNTA model yields two fully governed operational domains:

1. ZTNA protected business application traffic, with precise per-identity access control, and
2. Residual traffic, handled either through direct internet egress or via the SASE cybersecurity controls.

This architecture effectively mitigates the long-term structural problems introduced by traditional VPN-based remote access solutions and restores a clean, scalable security posture aligned with modern operational requirements.

ZTNA solutions address long-standing architectural challenges that emerge when service delivery depends on multiple interlinked ICT domains—such as network, backbone, workstation, remote-access, and identity services—each with its own complex change-management processes. Unlike traditional VPN infrastructures, which often become unintended bottlenecks, identity-based ZTNA isolates access to only the authorised services and therefore avoids the cascading issues typically imposed on VPN services.

By enforcing identity at the individual level, ZTNA enables controlled and low-risk technology transitions. This significantly improves change-management practices: the CAB can evaluate impacts at a per-user level rather than approving modifications that might affect thousands of VPN users simultaneously. Because ZTNA can be deployed alongside existing remote-access solutions, testing becomes straightforward, beginning with smoke tests for a single user and expanding to small pilot groups without extensive architectural disruption.

Rollout can proceed once testing issues are resolved, with identity-based logging providing clear visibility during intensified monitoring. The resulting architecture restores the original intent of secure remote access by restricting business-critical services to defined user groups through application connectors, while non-critical traffic follows alternative paths.

Ultimately, this produces two well-controlled domains—the ZTNA-protected service layer and all remaining traffic—effectively mitigating lifecycle-related weaknesses inherent in traditional VPN-centric models.

Figure below illustrates a use case scenario with a business application usage described with a ZTNA-based solution.

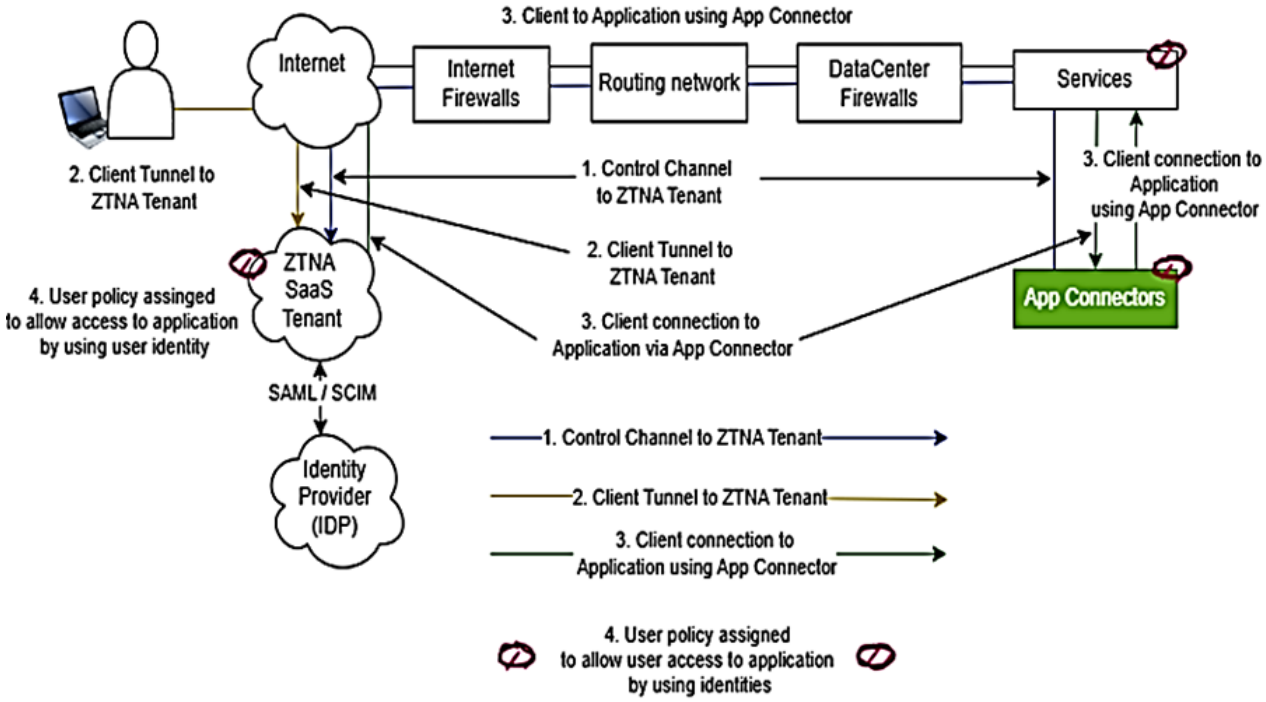


Figure 1. Basic service-model with Zero Trust Network Access

7 DATA ACCESS CONTROL PROCESSES

7.1 General Principles of Access Control

In a national, multi-partner platform, access control must balance organizational autonomy with the need for consistent governance and accountability. Distributed and federated models are therefore considered state-of-the-art for such environments.

Access control is a foundational security mechanism in SINTRA, enabling controlled access to sensitive sensor data, logs, and derived datasets.

Main principles:

- Least-Privilege Principle: Users receive only the permissions needed for their duties.
- Need-to-Know Basis: Access is strictly tied to predefined purposes.
- RBAC/ABAC Hybrid Model: Roles reflect organizational responsibilities
- Attributes capture dynamic conditions (purpose, time, location, sensitivity)
- Federated Identity Management: Enables cross-organizational authentication while preserving local authority.
- Centralized IAM: Integrates SAML/OIDC-based identity providers.
- Strong Authentication: MFA for elevated privileges; no anonymous accounts.
- Continuous Logging: Immutable records to support compliance and forensic investigation.

Access-control mechanisms are tightly integrated with the governance protocol and reflect customer expectations.

7.2 Access Control Models and Mechanisms

Solution Type	Description	Relevance in SINTRA	Example Technologies / Tools
Role-Based Access Control (RBAC)	Access rights are assigned based on predefined user roles reflecting organisational responsibilities.	Provides a clear and manageable structure for assigning permissions across partners and use cases.	Keycloak, WSO2 Identity Server

Attribute-Based Access Control (ABAC)	Access decisions are based on dynamic attributes such as user context, time, location, and data sensitivity.	Enables fine-grained and context-aware access control in complex, multi-partner environments.	Axiomatics, NextLabs
Hybrid RBAC/ABAC Model	Combines role-based and attribute-based approaches to balance simplicity and flexibility.	Supports SINTRA's need for both structured governance and dynamic access conditions.	Keycloak Authorization Services, NextLabs
Federated Identity Management	Allows users to authenticate using credentials from their home organisation, avoiding duplication of identities.	Essential for cross-organisational collaboration while preserving autonomy of each partner.	Shibboleth, Keycloak (SAML / OIDC)
Centralised Identity and Access Management (IAM)	A central system manages authentication and authorisation policies across the platform.	Ensures consistency, unified governance, and simplified compliance auditing.	Keycloak, WSO2 Identity Server
Federated Policy Enforcement	Access control policies are defined under shared governance but enforced locally by each organisation.	Supports decentralised environments where partners retain control over their own systems.	Open Policy Agent (OPA), Keycloak
Centralised Policy Enforcement	Access decisions are made by a central authority to ensure uniformity across the platform.	Simplifies compliance and auditing but introduces dependency on central systems.	Axiomatics Policy Server, OPA
Multi-Factor Authentication (MFA)	Requires multiple authentication factors (e.g. password + token) to verify user identity.	Enhances security for sensitive data access and administrative roles.	OIDC providers, Identity platforms
Zero Trust Architecture (ZTA)	Access is granted based on continuous verification of identity and context, not network location.	Strengthens security in distributed environments and supports secure cross-partner access.	ZTNA solutions, SASE frameworks

Audit Logging and Monitoring	Records all access and actions in immutable logs for traceability and compliance.	Enables forensic analysis, accountability, and regulatory compliance (e.g. GDPR).	ELK Stack, SIEM systems

7.3 Roles and Permissions

Federated identity management enables users from different organizations to authenticate using their home credentials, avoiding identity duplication and reducing administrative overhead. This approach supports collaboration under shared legal and regulatory frameworks while preserving organizational independence. In a single-country context, trust relationships are easier to establish; nevertheless, harmonized assurance levels, identity proofing practices, and governance structures remain essential. Technologies such as Shibboleth, widely used in research federations, and Keycloak, supporting SAML and OpenID Connect, exemplify this approach. Federated identity also strengthens accountability by allowing authentication events to be traced back to the originating organization.

7.4 Authentication and Authorisation

Authorization requirements in such platforms are increasingly addressed through hybrid Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) models. These models combine relatively stable role assignments with dynamic attributes such as purpose of use, data sensitivity, time, or contextual risk. This hybrid approach enables fine-grained authorization while maintaining manageable policy structures. Its main challenge lies in policy complexity and lifecycle management as organizational roles and attributes evolve. Platforms such as WSO2 Identity Server, Keycloak Authorization Services, and NextLabs support hybrid policy enforcement and are particularly suited to environments where regulatory compliance and operational flexibility must coexist.

Policy federation further extends this model by allowing organizations to align access control rules while enforcing them locally. This avoids centralization of control while promoting consistency. In a single-country setting, shared legal interpretations reduce ambiguity, but semantic alignment of policies remains a key challenge. Policy federation can be combined with Secure Access Service Edge (SASE) architectures to enforce trust-based access at the network edge. While this approach improves scalability and resilience, it requires robust governance mechanisms to manage policy evolution and resolve conflicts.

AOCC platform generated in *UC 1 – Airport use case* has a comprehensive roles & responsibilities management.

Authorization: Role and Responsibility-Based Access

RBAC: AOCC Operations Manager, Terminal Operation, Apron/Ramp, Security, Facility, Regulation, etc. module/display/operation authorizations by role. Least privilege and authority hierarchy/segregation.

Role-Based Access Control (RBAC)

Authorization is centrally managed through Keycloak, as is authentication; Synchronized groups from Azure AD are mapped to Keycloak roles, and application services make access decisions based on role/permission claims within the token.

Defined operational roles (example):

Role	Area of Responsibility	Typical Authorizations
AOCC Operations Manager	General operations coordination	All modules (read), event management, threshold/warning rule management, configuration
Terminal Operation	Passenger and terminal flow	Terminal modules, FIDS/BIDS, queuing/density indicators
Apron / Ramp	Aircraft ground handling services	Apron status, turnaround, gate/stand assignment, ramp events
Security	Airport security	Security cameras, event logs, access control data
Facility	Building and infrastructure	BMS metrics, maintenance requests, IoT device statuses
Regulation / Compliance	Compliance monitoring	Audit logs, reports, KPI/SLA views (read-only)
Administrator / Admin	System management	User/role management, integration configuration, system settings
Viewer	Reader-only stakeholders	View only the dashboard

Roles:

It defines permission for module, display, process (CRUD) and data source/location breakdown when necessary. Each API call is validated against the role/permission claim in the user's token (at the NestJS guard / policy enforcement point).

Least Privilege:

- Users have no authority by default; Access is granted only by explicit assignment.
- Roles are narrowly defined according to the area of responsibility; A user in the Apron role cannot see security camera recordings or regulation reports.
- Data-level filtering: The user's terminal/location/unit is read through the token claim or user profile and applied to queries as an automatic filter (row-level access).
- Module visibility: On the UI side, menus and screens are dynamically rendered according to the user's role; on the backend side, each endpoint also passes an authorization check (defense in depth).
- The authorization definition for newly added modules/screens is opt-in — they cannot be accessed unless explicitly assigned.

Segregation of Duties:

Critical operations require a high level of authorization, separate from ordinary operational authorizations:

- Incident closure / permanent resolution approval → Operations Manager or authorized supervisor.
- Configuration change (integration settings, source system connections) → Admin role.
- Threshold / warning rule update → Relevant module maintainer + approval (maker-checker principle: the changer cannot approve their own change).
- User/role management → Admin only; upgrading your own role is blocked.
- Audit log access → Compliance / Admin only; logs cannot be changed (append-only).

These principles are applied through the following mechanisms:

- Hierarchical roles: Parent roles include the authority of lower roles (composite roles), but critical operations require additional explicit permission.

- Segregation of duties rules: Prevents the same user from simultaneously moving two conflicting roles (e.g., who made the change + approver).

- Step-up authentication: MFA is requested again before high-risk transactions.

- All authorization decisions are audit logged: who, when, to which resource, what action they took/attempted, → are transferred to SIEM.

Architecture Summary:

Azure	AD	groups	→	Keycloak	roles	→	JWT	claims
		↓						
	API		Gateway		/		Service	Guards
		↓						
	Module/Display		authority	+	Row/Data		level	filter
		↓						
	Audit Log (append-only)							

As a result, authorization; It is managed on a single policy plane that works synchronously with the corporate directory, built on the principles of central definition, least privilege, hierarchical approval, and segregation of duties.

Finland: Centralised and Federated Policy Enforcement

Centralized policy decision-making with local enforcement provides uniform authorization outcomes across participating organizations and simplifies compliance auditing. In national platforms, consistent legal interpretation supports centralized governance and oversight. Solutions such as Axiomatics Policy Server or centralized Open Policy Agent (OPA) deployments enable this model. However, centralization introduces potential single points of failure and increases trust dependencies, making high availability, redundancy, and strong governance essential.

Alternatively, federated policy decision-making distributes authorization logic across organizations while operating under shared governance constraints. This improves resilience and supports heterogeneous operational requirements. Tools such as OPA and Keycloak enable decentralized evaluation while maintaining interoperability. Federated models require strong coordination, comprehensive logging, and audit mechanisms to ensure

compliance, but they are well suited to dynamic environments where autonomy and adaptability are prioritized.

Belgium: We have implemented a Zero Trust architecture for the Belgium C-SITE and Skyebase platform by enforcing strict identity verification, granular access control, and continuous validation across all layers of the system. At the identity level, AWS Cognito is configured with mandatory multi-factor authentication (SMS/TOTP), strong password policies, and short-lived tokens with backend validation to ensure secure session management. Access control is refined using fine-grained IAM policies aligned with the principle of least privilege, complemented by attribute-based access control (ABAC) in the application layer through dynamic role assignment (e.g., custom:role) and enforcement in Django services. On the network side, the platform is segmented using VPCs, subnets, security groups, and ACLs to isolate front-end, back-end, and database components, minimizing lateral movement. Data protection is ensured through encryption at rest and in transit using AWS KMS with regular key rotation. Together, these measures eliminate implicit trust, requiring continuous authentication and authorization for every request, thereby aligning the platform with core Zero Trust principles.

A Zero Trust architecture has been implemented by combining federated identity, strict trust policies, and continuous contextual verification across both infrastructure and application layers. At the identity and access layer, OpenID Connect (OIDC) is used to federate GitHub Actions with AWS, eliminating long-lived credentials and enabling short-lived, role-based access via *sts:AssumeRoleWithWebIdentity*, with tightly scoped trust policies that restrict access to specific repositories, audiences, and environments . Fine-grained IAM permission policies further enforce least privilege by limiting actions (e.g., S3 operations) to explicitly defined resources. At the application layer, Zero Trust principles are reinforced through granular role- and permission-based access control, mandatory multi-factor authentication (e.g., email verification), and enhanced token handling with secure storage, encryption, and strict validation. Continuous verification is achieved by incorporating contextual signals such as geolocation, time-of-access checks, and dynamic claims, combined with comprehensive logging of both authorized and unauthorized access attempts to support auditability and anomaly detection. Additionally, database access is hardened through layered controls including MFA, authorization policies, IP restrictions, and encryption. Together, these mechanisms ensure that every access request—whether from CI/CD pipelines or end users—is explicitly authenticated, tightly authorized, and continuously validated, fully aligning the system with Zero Trust principles.

7.5 Retention & Deletion Policies

Data collected for research is retained only for the duration necessary for scientific analysis.

- Data Retention: Data should be retained only as long as is necessary for the specific purpose for which it was collected.
- Data Deletion: Data should be securely deleted when it is no longer needed.
- Security policies: Security policies should be implemented to guide the processing of sensitive data and ensure compliance with data protection regulations
- Operational datasets (logs, events) follow partner-specific retention rules but cannot exceed GDPR compliant timeframes.
- Backup retention must be limited and secured; SINTRA follows a two-week backup deletion policy unless legally mandated otherwise.

Data will be retained only as long as is necessary for the specific purpose for which it was collected. Data will be securely deleted when it is no longer needed. Security policies will be implemented to guide the processing of sensitive data and ensure compliance with data protection regulations.

AOCC platform generated in *UC 1 – Airport use case uses* simple retention and deletion policies

Audit Log Retention Policies:

The platform maintains strict retention periods for audit trails to ensure security investigation and regulatory compliance:

- Operational Audit Logs: These are retained for a minimum of 1 year.
- Regulatory/Compliance Logs: Logs falling under regulatory scope are retained for 5+ years.
- Access Credentials: Service account secrets (machine identity) are subject to a 90-day periodic rotation or immediate rotation upon a security event.

Deletion and Immutability Standards:

The document emphasizes data integrity and controlled deletion through several mechanisms:

- Append-Only Principle: Audit logs are strictly append-only; they cannot be modified or deleted once written. This is enforced via technical measures such as WORM (Write Once Read Many) storage or S3 Object Lock.

- Timeline Integrity: In incident management, users are prohibited from deleting comments or attachments. If a correction is needed, a new entry must be added while the original remains part of the immutable timeline.
- Exported Data: Generated report files are assigned an expiration period (access life), after which sharing links become invalid and the files are automatically deleted.
- User Lifecycle: The "joiner/mover/leaver" lifecycle for users is managed centrally via Azure Active Directory (Entra ID), ensuring that access is revoked immediately when a user leaves the organization.

Session and Token Lifetimes:

- Retention of active sessions is managed through time-to-live (TTL) settings:
- Access Tokens: Short-lived, typically 5–15 minutes.
- Refresh Tokens: Medium-lived, typically 8–24 hours.
- Session Termination: Single Logout (SLO) is used to terminate all sessions across the platform simultaneously from a single point.

Data Minimization (Privacy by Design):

Following KVKK/GDPR principles, the platform defaults to data minimization:

- Unauthorized fields are omitted from responses entirely (never sent to the UI) rather than just being hidden.
- Sensitive data is often masked or anonymized at the backend level, ensuring the original sensitive value is not retained in the client's memory or network logs.

8 PRIVACY & GDPR CONSIDERATIONS

The Platform outlines several architectural and procedural measures designed to ensure compliance with KVKK (Personal Data Protection Law) and GDPR standards. The core philosophy centers on "Privacy by Design," ensuring that data protection is an automated part of the system rather than a manual afterthought.

Data Minimization and Access Control:

The platform strictly follows the principle of data minimization, ensuring that users are only presented with the specific information required to perform their duties. This is achieved through multi-layered filtering:

- Default Deny Policy: Access is not granted by default; it requires explicit assignment based on a user's role and operational necessity.
- Backend Exclusion: Sensitive fields that a user is not authorized to see are omitted from the system response at the API level. This prevents unauthorized data from ever reaching the user's device or the browser's memory.
- Spatial and Temporal Scope: Access is limited geographically (e.g., specific terminals or gates) and temporally (e.g., only the next 3 hours of operational data), preventing broad exposure of the entire airport's passenger or personnel data.

Masking and Anonymization Strategies:

For instances where data must be visible but remains sensitive, the platform employs various masking techniques to protect individual identities.

- Field-Level Masking: Personal identifiers like names, emails, and phone numbers are partially masked (e.g., "A*** Y***") so that the context is maintained without revealing the full identity.
- Tokenization and Hashing: Highly sensitive identifiers such as passport numbers or Turkish ID numbers (TCKN) are tokenized or hashed at the database level.
- K-Anonymity for Reporting: When generating analytical reports, the platform suppresses or rounds data for small groups (e.g., fewer than 5 people) to prevent the "re-identification" of individuals within a dataset.

Accountability and Transparency

A critical component of GDPR is the ability to audit who accessed what data and why.

- Immutable Audit Trails: Every access to sensitive PII (Personally Identifiable Information) is logged in an append-only format that cannot be altered or deleted.

- Justification Requirements: For critical actions, such as unmasking a passenger's passport number for an emergency or exporting a report, the system requires the user to enter a formal justification.
- Adli İz (Forensic Marking): Any data exported from the system is tagged with a digital watermark and the identity of the person who exported it, creating a traceable path in the event of a data leak.

Secure Data Lifecycle:

The platform manages the entire lifecycle of sensitive data to prevent "zombie" access or forgotten records.

- Automated Expiration: Exported files and reports are given a short "access life" and are automatically deleted by the system after the expiration period.
- Centralized Identity Management: By integrating with Azure AD, the platform ensures that as soon as an employee's contract ends, their access to all sensitive airport and passenger data is immediately revoked across all modules.
- Maker-Checker Controls: For high-risk privacy decisions, such as changing alert thresholds or accessing incident reports, the platform requires a second authorized person to approve the request, preventing a single point of failure or insider threat.

8.1 GDPR-Sensitive Data

According to the design of the platform, sensitive data is classified into specific groups based on its impact on privacy, security, and commercial value. The platform distinguishes between operational information and data that requires higher levels of protection or masking.

Personal and Identity Information

The most restricted category is Personal (PII) data, which includes identifiers such as passenger names, passport numbers, and personnel identification details. This also extends to contact information like email addresses and phone numbers, which are subject to masking strategies to prevent unauthorized disclosure. National ID numbers (e.g., TCKN) are also considered sensitive and are typically masked or tokenized at the backend level.

Sensitive Operational and Security Data:

Operational data becomes sensitive when it involves security-related incidents or detailed descriptions of events. This includes Sensitive Operational data such as incident descriptions, operational notes, and security camera (CCTV) recordings. Access to this information is restricted to specific roles like the Operations Manager or Security personnel and often requires "break-glass" procedures or double-approval (maker-checker) to be unmasked or exported.

Regulatory and Legal Data:

Data categorized as Regulatory / Legal Hold includes incident reports and any information falling under the scope of KVKK (Personal Data Protection Law) or GDPR. This data is used for compliance tracking and is often limited to read-only access for auditors or compliance officers.

Commercial and Restricted Data:

Commercial sensitivity is another key classification, covering Commercial / Restricted data such as contract details, SLA (Service Level Agreement) penalties, and commercial KPIs. Furthermore, forward-looking Temporal Scope data, such as machine learning-generated predictions and capacity projections, is treated as sensitive because it carries significant commercial value and could be misleading if misinterpreted.

8.2 Enhanced Anonymisation & Privacy-Preserving Methods

In SINTRA we have enhanced techniques and methods used to establish anonymization/privacy.

Visual privacy: To support the development of the privacy-preserving solutions all collected sample and test data is compliant with GDPR. Where personally identifiable information is collected, processed and persisted for later use, the subjects are informed and consent provided. This is particularly the case with the test-day data collection required for evaluation of the privacy-preserving methods that have been developed.

Enhanced BLE security and privacy for secure tracking: We use Bluetooth Low Energy (BLE) beacons to conduct the logistical tracking of assets. A BLE beacon periodically broadcasts an identifier which can be linked to an individual asset, in accordance with the iBeacon protocol. However, this identifier was not secured, meaning it was open to various attacks such as spoofing and replay as well as eavesdropping. To counteract this, we propose a security extension, based on Google's Ephemeral Identifiers (EID), in which an identifier is encrypted before transmission. For this purpose, we used the on-tag hardware accelerated AES-GCM authenticated encryption with associated data (AEAD) cipher, with a 128-bit key. Additionally, we implement periodic key rotation to facilitate forward security. Using the cipher, we encrypt an internal counter value that is synchronized with resolving server. Only if you know both the start time and frequency of the counter, as well as corresponding key, you can link the encrypted identifier to a specific beacon. This means that a beacon's identifier can no longer be traced back to a specific physical beacon, meaning it is no longer susceptible to eavesdropping. Additionally, this defense does not allow for any practical replaying.

Visual privacy for UAS video footage streaming: Video footage produced by UASs during surveillance or inspection missions around critical infrastructure may contain sensitive visual information. According to GDPR, this information should be processes appropriately. For example, a UAS pilot may not need access to faces of passengers or license plates of parked cars. However, simple irreversible anonymization, such as blacking out detected information, may not sustainable. If for example law enforcement needs access to the footage, then the anonymization needs to be reversible. For this purpose, we have devised a processing pipeline in which we first perform machine learning-based detection of sensitive areas in a frame. Then, we embed a watermark inside the frame, which encodes recovery information for that image, alongside authentication bits. Afterwards, the sensitive areas are anonymized. However, using the watermark, we can automatically detect which areas have been anonymized downstream, and recover those areas of the image in grayscale solely based on the watermark. Additionally, the watermark also protects against malicious tampering, as we can also use it to detect any other alterations to the image. Both this tamper detection and the recovery are based on randomization enabled by a cryptographically secure pseudorandom number generator (CSPRGN), which uses a shared secret key as an input. That way, only authorized personnel (e.g. law enforcement, a system administrator, …) are able to undo anonymization and view the original image.

9 CONCLUSION

This deliverable provides a consolidated overview of the data governance, data acquisition, data sharing, and data access control processes within the SINTRA project. The document establishes the principles, requirements, and mechanisms that support secure, trustworthy, and compliant handling of data throughout its lifecycle.

The presented governance framework is based on risk-driven principles and aligns with applicable regulatory requirements, including GDPR and relevant cybersecurity regulations. It defines how data is acquired, processed, shared, accessed, retained, and protected across the different SINTRA use cases and partner environments. Particular attention has been given to privacy protection, access control, traceability, and interoperability, ensuring that data can be used effectively while respecting legal, ethical, and operational constraints.

The deliverable also consolidates partner-specific contributions and use-case requirements related to airport, port, railway, construction site, and retail/F&B environments. These contributions demonstrate the diversity of data sources, acquisition methods, sharing mechanisms, and access control approaches that must be supported within the SINTRA platform. At the same time, they highlight the importance of applying common governance principles across heterogeneous systems and organisational boundaries.

The framework presented in this document provides the foundation for the subsequent development, integration, validation, and demonstration activities within the project. As the SINTRA platform evolves, the described processes and governance mechanisms will continue to be refined based on technical developments, use-case feedback, and regulatory considerations. Additional partner contributions and implementation details may be incorporated as the project progresses and the platform moves towards large-scale validation and demonstration activities.

Overall, the work described in this deliverable contributes to the establishment of a secure, privacy-aware, and interoperable environment for data-driven situational awareness, supporting the broader objectives of the SINTRA project.