EagleArca

Access And Platform Overview

https://www.youtube.com/embed/VYuoxGxOgN8?si=vw5XZd7hAPd4Eoy5

> Click here to download the PDF

Welcome. In this video, we will explore EagleArca, a digital platform designed to bring together, within a single operational environment, territorial observation, geospatial data management, and the execution of missions with robotic systems. The objective is not only to describe what the platform allows users to do, but to understand how it fits within a broader ecosystem of digital technologies dedicated to environmental monitoring, infrastructure inspection, and the management of processes distributed across large areas.

In recent years, territorial monitoring and asset management activities have become progressively more complex. The amount of available data has grown significantly, driven by the satellites of the Copernicus programme, by sensors installed in the field, by drones, and by mobile robots operating directly in real-world contexts. This abundance of information, however, represents value only to the extent that it can be organized, queried, and translated into knowledge useful for decision-making. This is precisely the space in which EagleArca operates.

In this video we will see what EagleArca is, what it is used for, in which contexts it is applied, which problems it is designed to address, and what its main functions are. We will begin with a general view of the platform and gradually move toward the details of its operational components.

What EagleArca is

EagleArca is a digital platform for territorial monitoring and spatial data management, designed to bring into a single environment the set of activities that, in traditional workflows, are often fragmented across different tools. Data acquisition in the field, the organization of information into structured classes, two- and three-dimensional cartographic representation, analysis through charts and reports, and the planning of missions with drones and quadruped robots: all these activities find in EagleArca a single point of access.

The platform is articulated around two main components. On one side, the Backoffice, dedicated to system configuration and administration. On the other, the App, the operational environment in which users work directly with projects, data, and missions. This separation between configuration and operation allows for precise control over how the system is structured, while at the same time providing field operators with a clear interface for data collection and consultation.

EagleArca is organized around the concept of organization. Each organization represents an isolated environment within the platform, with its own users, projects, object classes, and configurations. A single user can belong to multiple organizations, but their access to projects always depends on the roles and groups to which they have been assigned. This architecture, as we will see, is what allows the platform to be used by heterogeneous teams without compromising the separation between different operational contexts.

What it is used for and which needs it adesses

To understand what EagleArca is used for, it is helpful to start from the problems the platform is called upon to solve. In many operational contexts, organizations involved in territorial monitoring, infrastructure management, or survey activities face a series of recurring difficulties.

Data come from very different sources. Satellite imagery from the Copernicus programme offers a systematic and large-scale view, but it must be integrated with closer-range observations. Drones make it possible to obtain high-resolution images over targeted areas, while mobile robots enable direct inspections at ground level. To these sources are added permanently installed sensors, which produce continuous time series. Each source has its own format, its own update frequency, and its own dedicated tools. Without a common environment in which to bring them together, the risk is that of working with parallel streams of information that are never truly connected to each other.

To this is added the collaborative dimension. Many projects involve a wide range of professional figures: field technicians, analysts, project managers, system administrators. Each of these roles has different needs in terms of data access and level of action. Managing these differences manually, across disconnected tools, is costly and easily introduces errors.

EagleArca responds to these needs by offering a unified environment in which heterogeneous sources can converge, information can be structured according to coherent schemas, and the visibility and editability of every element can be precisely regulated. It is a digital infrastructure designed to accompany the entire data cycle: from planning, to collection, to archiving, all the way to analysis and the generation of documentary outputs.

The application contexts are numerous. In environmental monitoring, the platform supports the observation of phenomena such as the evolution of vegetation, water conditions, and changes in the soil. In agriculture, EagleArca makes it possible to integrate satellite observations, drone missions, and ground inspections to support crop management, the planning of rotations, and the verification of plant health. In infrastructure management, the platform is used for the census, inspection, and long-term control of distributed assets, from electrical networks to water networks, all the way to buildings and industrial areas. Furthermore, EagleArca finds application in three-dimensional surveys, in the post-event monitoring of areas affected by natural phenomena, and in surveillance activities in environments where human intervention is difficult or risky.

Platform Architecture

Let us now look more precisely at how EagleArca is organized from a logical point of view. Understanding this structure is essential in order to orient oneself within the functions that follow. At the highest level we find organizations. An organization is an overall container, within which users, projects, object classes, and configurations live. Organizations are separated from each other, and this separation guarantees that the data and operations of one context do not interfere with those of another.

Within the organization, users are placed. Each user is a person who accesses the platform, and can be classified as internal, if created directly within the organization, or external, if coming from another organization and added as a collaborator. The access of users, however, is not managed at the individual level, but through two intermediate structures: roles and groups.

A role defines the permissions associated with a given configuration: what is visible, what is editable, what remains hidden. Permissions apply to object classes and reports, and are articulated across three levels: hidden, visible, and editable. A group, in turn, is a set of users who share the same role and access the same projects. By combining roles and groups, the platform makes it possible to build scalable access configurations, avoiding the need to repeat the same settings for every individual user.

Projects represent the actual operational containers. Each project is a dedicated workspace, within which the objects collected in the field live, together with the classes to which they belong, the cartographic resources, the user groups that can operate within them, and the associated reports. A project can be dedicated, for example, to the monitoring of an urban area, to the three-dimensional survey of an infrastructure, to the mapping of the state of a road network, or to the analysis of the conditions of a river area.

Completing this architecture, we find the Global Lists, object classes that are independent from individual projects and available at the level of the entire organization. Global Lists collect shared data, such as records, inventories, or verification reports, which must remain accessible to multiple teams simultaneously. They are centralized repositories, useful for hosting reference information that can be reused across different projects.

Object Classes

One of the central concepts in EagleArca is that of the object class. An object class defines the data structure of a given element to be represented or surveyed within the project. Each class represents a type of observable element, such as a pole, a tree, a pipeline, a sensor, or an operational area, and specifies which information must be collected for each instance of that class and how that instance must appear on the map.

The platform distinguishes two main categories of object classes. Vector classes allow the creation of georeferenced objects, displayable on the map. The basic geometry of these classes can be a point, for individual elements such as single sensors or trees; a line, for extended elements such as pipelines or paths; or a polygon, for areas such as buildings, operational zones, or perimeters. Non-vector classes, known as Form classes, instead define the attributes of objects that do not have a geographic representation, and that are used to manage information such as inspection reports, technical checks, or reference lists.

Each object class is composed of a set of attributes, which are the data fields necessary to describe each instance. Attributes can take different forms: text fields, numerical values, predefined value lists, dates, attachments such as images or videos, measurable dimensions such as length or area, Boolean fields, or lists of sub-attributes for composite data. Each attribute is further configurable in terms of being required or optional, having a default value, being locked in read-only mode, or being marked as the main attribute, the one that quickly identifies the instance during consultation.

Vector classes can be characterized by a base style, defined through shape, color, thickness, and transparency. To this are added conditional styles, a function that makes it possible to automatically modify the appearance of objects on the map based on the value of one or more attributes. It is thus possible, for example, to visually distinguish an operational status from a fault status, or to represent through a color gradient the progression of a continuous quantity such as height, temperature, or an environmental index.

The careful design of object classes is a fundamental step, because it establishes in advance how data will be collected, organized, and then queried. It is in this phase that an information requirement is translated into a coherent, replicable structure that can be used at the project scale.

Project views and operational tools

Once inside a project, EagleArca provides a set of views between which it is possible to switch in order to interact with the data. The main ones are the two-dimensional map, the three-dimensional map, the Inventory, the Reports, and the Streaming view. Each of these views responds to a specific operational need.

The 2D view displays the georeferenced map of the project and the objects it contains. It is the main tool for territorial exploration and for verifying the position of assets. The 3D view, in turn, places objects in a three-dimensional space, integrating representations such as meshes and point clouds that allow the territory to be observed in its depth and spatial complexity. Vector objects are georeferenced and maintain the same geographic position in both views, so that switching between 2D and 3D does not result in any loss of coherence.

The Inventory represents a tabular and structured view. It shows the complete list of objects of a specific class, allowing users to consult their attributes, filter them, sort them, and compare them. It is particularly useful when the user is not working on a single element, but on a set of instances that share common characteristics. The Inventory is also the only way to access Form classes, which have no geometry on the map.

The Layers panel allows users to manage the display of views and of individual object classes. From here it is possible to show or hide elements, lock their interactions, activate identification labels, and adjust the snap behavior during the drawing of geometries. It is an operational tool that allows the user to build, session by session, the working environment most suited to the task at hand.

When an object is selected, the platform opens an attribute detail panel, which reports all the values associated with the selected instance. From this panel it is possible to edit the data, export them in CSV format, delete the object, and locate it in the other views. The ability to work simultaneously on the map and on the detail panel, without having to close one to consult the other, is one of the aspects that characterize the operational logic of the platform.

Among the operational tools is also the ability to create new objects directly within the project. For Form classes, insertion takes place through an attribute compilation panel. For vector classes, the user selects the drawing tool from the toolbar, chooses the object class of interest, and traces the geometry on the map, whether it is a point, a line, or a polygon. Once the drawing is complete, the attribute panel opens for the compilation of the associated information.

To this is added the shapefile import function, which makes it possible to upload geographic data in a standard format and automatically transform them into objects of the selected class. During the import, the geometries present in the file are converted into instances, and the attributes are populated by associating the fields of the file with those of the object class. This is a function that facilitates the integration between EagleArca and existing GIS workflows.

Sensor data, Dashboard and Reports

One of the most relevant areas of the platform concerns the management of continuous data coming from sensors. When an object class is configured with one or more data sources, known as Data Sources, the objects of that class can be linked to streams of information organized over time. It is thus possible, for example, to associate an environmental sensor with a Data Source that systematically records temperature, humidity, or other monitored values.

Data Sources are complemented by charts, which are also configured at the object class level. EagleArca supports several types of charts, including line, counter, bar, scatter, three-dimensional scatter, and heatmap. Each chart can be customized in its source data, its visualization, and its styles. Conditional styles are also available for charts, making it possible to automatically highlight anomalous values when predefined thresholds are exceeded.

The consultation of this information takes place through the Dashboard Panel. This panel displays the charts associated with the selected objects, grouping them by instance. When the user selects multiple objects simultaneously, the panel makes it possible to compare their time series, supporting comparative analysis activities. It is a particularly useful function when monitoring extends across multiple measurement points and requires observing in a coordinated way the evolution of different quantities.

Alongside the Dashboard, the platform offers the Report Panel, dedicated to the consultation and management of reports. Reports are documents generated from templates configured in the Backoffice, which define structure, parameters, and available formats. Within the project, users with the appropriate permissions can generate new reports, download them, and view their details. Reports represent the documentary output of the system, and are the tool through which collected data are returned in a synthetic form, shareable outside the platform.

The Mission Planner and integration with drones and robots

One of the distinctive aspects of EagleArca is its integration with robotic systems. The platform includes a dedicated module, the Mission Planner, through which it is possible to plan, execute, and archive operational missions carried out by drones and quadruped robots.

Mission configuration passes through the Global Lists. In particular, three fundamental lists are used. Mission Units represent the robotic units available in the project. Add-ons represent sensors, cameras, and accessories, native or additional, that can be installed to extend the capabilities of the mission. Profiles represent technical configurations composed of one or more Add-ons, and allow complex setups to be quickly recalled. The platform also integrates the hangar, which serves as the home base and recharging station for the automated missions of compatible drones.

Once the units have been registered, the user can proceed with the actual mission planning. In the case of a quadruped robot, the process involves defining a Mission Home, that is, the robot's starting and return point, and a sequence of Waypoints, which represent the stages of the route. To each Waypoint it is possible to associate specific operational actions, such as the acquisition of photographs or the recording of videos. On the map it is also possible to draw Geofences, that is, areas of safe operation, and Danger Zones, that is, areas from which the robot must be excluded.

The logic for the drone is analogous, with some adaptations specific to flight. The sequence of Drone Waypoints defines the flight trajectory, and the actions that can be associated include the acquisition of images. Before starting the mission, the system performs a Mission Pre-Check, a preliminary verification of the connection status of the drone and the hangar, and of the consistency of the configured Waypoint sequence.

The Mission Pre-Check is an important step. The system distinguishes between blocking errors, which prevent the mission from starting until they are resolved, and non-blocking errors, which allow the user to proceed but require an explicit acceptance of responsibility from the operator. Once the pre-check has been passed, the mission can be started.

During execution, the platform offers a real-time monitoring mode. Through the Streaming view it is possible to follow the video feeds coming from the Mission Units and the connected add-ons, record videos, and take photographs. As for the quadruped robots, the operator can suspend the automatic execution at any time and take manual control of the robot directly from the platform interface. All media acquired during the execution are saved automatically. For the quadruped robot, media associated with Waypoints remain consultable in the archived Waypoints at the end of the mission, while media acquired through standalone actions are saved in the archived missions section. For the drone, instead, media acquired during the execution of a Waypoint are saved directly in the original Drone Waypoint.

Once completed, the missions remain available for consultation. It is possible to review the configuration used, the route actually followed, and the media acquired. In the case of quadruped robots, it is also possible to restore the configuration in order to generate a new mission, avoiding the need to reconfigure it manually from scratch.

Integration of satellite, aerial and ground observation

The overall value of EagleArca emerges more clearly when one considers the possibility of integrating, within the same operational environment, data sources that act on different scales. Satellite observations from the Copernicus programme offer a systematic view over large areas. Drone missions provide high-resolution imagery over targeted zones. Ground inspections carried out with the quadruped robot allow close-range observation of individual elements at specific points in the territory.

In the application contexts documented for the platform, this integration takes concrete form. In the agricultural domain, for example, Sentinel data make it possible to observe extensive cultivated areas and to analyze environmental indicators that influence crop growth. Drones, managed through the Mission Planner, perform automated flights to acquire detailed aerial imagery. The quadruped robot, finally, is employed for ground-level inspections, down to the individual group of plants. The three scales of observation converge in the same project within the platform, where they can be consulted, compared, and translated into operational decisions.

This logic of integration extends to other domains as well. In the monitoring of water resources, in the mapping of cultivated areas, in the management of agricultural infrastructure, and more generally in all contexts in which knowledge of the territory requires moving between a comprehensive view and a punctual analysis, EagleArca offers a coherent environment in which these different levels of information can coexist.

The overall workflow

To summarize, it is useful to observe the workflow that the platform supports as a whole. It begins with the Backoffice, where the administrator configures organizations, users, roles, groups, object classes, report templates, and projects. It is in this phase that the structure of the system is defined, and that the rules according to which the data will be collected and organized are established.

The workflow then moves to the App, where operators access projects and carry out the actual activities. Here, data are collected in the field, manually or through automated missions; the attributes of objects are filled in; external shapefiles are imported; charts and dashboards are consulted; reports are generated; and missions with drones and quadruped robots are planned and started.

The collected data remain within the platform, organized according to the configured classes and structures, and become the starting point for further analyses, for the generation of documentary outputs, and for comparison over time. The platform in this way supports the entire data cycle, from planning to delivery, maintaining coherence across its different phases.

Conclusion

In this video we have seen a general overview of EagleArca. We introduced what the platform is, what it is used for, in which contexts it is applied, and which needs it is designed to address. We then examined its architecture, based on organizations, users, roles, groups, and projects, and we described the central role of object classes as a tool for structuring data. We reviewed the views and operational tools available within the project, from the 2D and 3D maps to the Inventory, from the Dashboard panel to the reports. Finally, we explored the Mission Planner module and its integration with drones and quadruped robots, and we considered the overall value of integrating satellite, aerial, and ground observation.

EagleArca presents itself as a unified operational environment for the digital monitoring of the territory, capable of accompanying the user throughout the entire data cycle and of bringing into coherent dialogue different technologies and sources of information. See you in the next video.