Release guide

LuciadRIA 2025.0
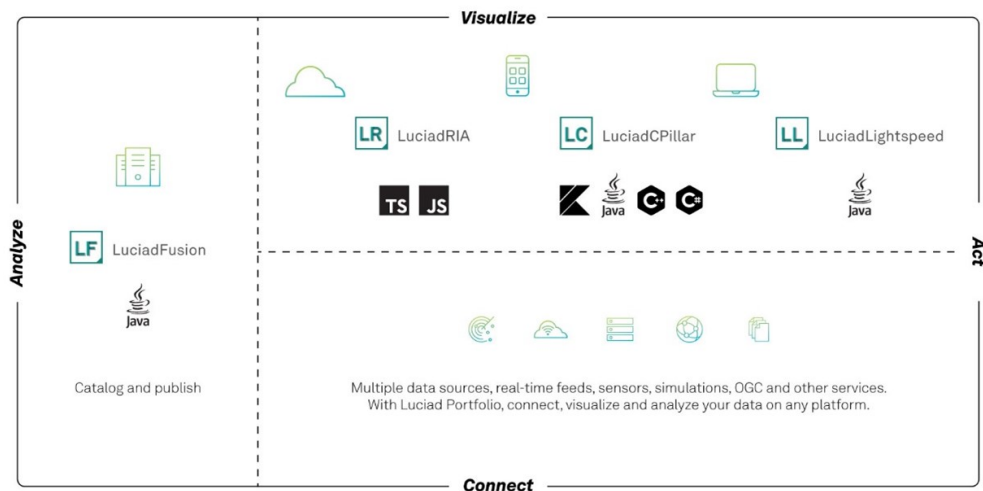
24 June 2025

Contents

About this release

The 2025.0 release of LuciadRIA adds support for Azure Maps, introduces vertical view labels on axis sub-ticks, and adds lines on the *BorderGeoCanvas*. It also includes support for *RasterImageModel* and *RasterImageLayer* on WebGLMap, as well as support for single-image WMS requests.

The Luciad portfolio

2D non-WebGL map end of life announcement

LuciadRIA was released in 2012 with a 2D map based on an HTML5 Canvas. In 2016, the WebGL-powered hybrid 2D/3D map was introduced, leveraging WebGL support in most browsers. Moving forward, we will focus solely on WebGLMap. WebGLMap offers all the capabilities of a non-WebGL map and brings much more to the table.

We previously announced that 2024.1 was to be the last release with the Canvas map support. However, we decided to keep the Canvas map for one more release. This means that the 2025.0 release now marks the end of the non-WebGL Canvas Map in favor of the hardware-accelerated WebGL Map only. The 2025.0 release, and possible future 2025.x minor upgrades, will be the last one to contain the 2D Canvas Map.

The 2025.0 version of LuciadRIA will be actively maintained until the release of version 2028. Contact the Luciad Product Management team at product.management.luciad.gsp@hexagon.com if you plan to extend your maintenance on LuciadRIA 2025 beyond 2028 so we are aware of your project.

You can find more information here.

Benefits of the new features

Integrate Microsoft Azure Maps

LuciadRIA 2025.0 introduces support for Azure Maps. Bing Maps is [being deprecated by Microsoft](#), which offers Azure Maps as its new mapping service. We will continue to support Bing Maps for *Bing Maps for Enterprise services* users, for which Microsoft will continue support until June 30th, 2028.

Example of Azure Maps traffic and aerial imagery in LuciadRIA

The Azure Maps support includes the following features:

- Integrate Azure Maps as a map layer using *AzureMapsTileSetModel*, similar to how Bing Maps was used

- Use an Azure Maps Shared Key for authentication
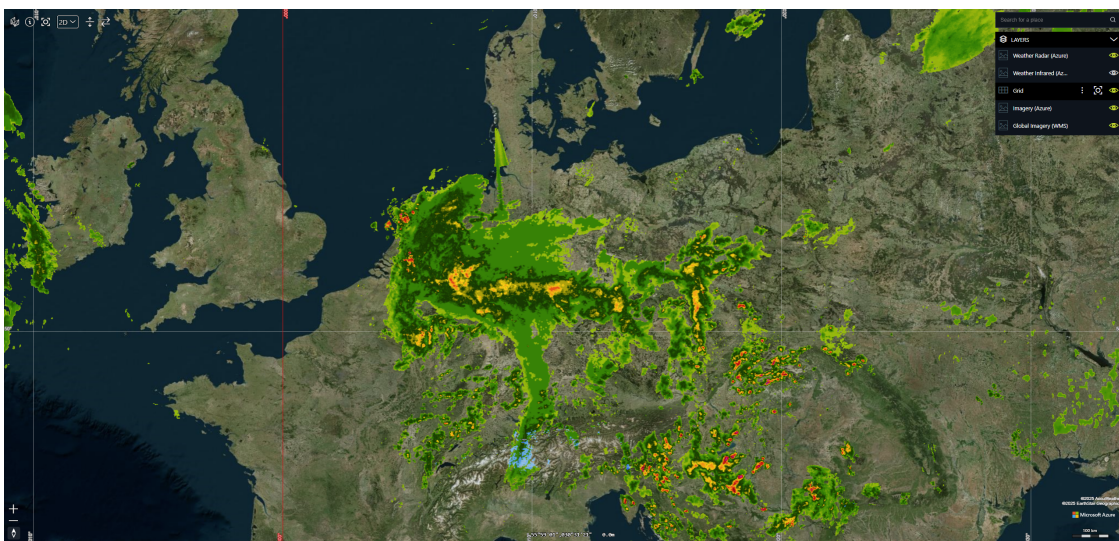
Example of Azure Maps weather data in LuciadRIA

Sample code/documentation to get you started

For more information on how to use Azure Maps in your application, see the [Azure Maps guide](#).

The Data Formats sample has been updated to include Azure Maps. As before, the LuciadRIA release includes the sample code.

Style and label axis subticks on a vertical view

You can now style and label not only the main ticks on the axes of a vertical view, but also the more fine-grained sub-ticks. The number of sub-ticks will automatically adapt, depending on the zoom level, to make sure that all labels remain readable.
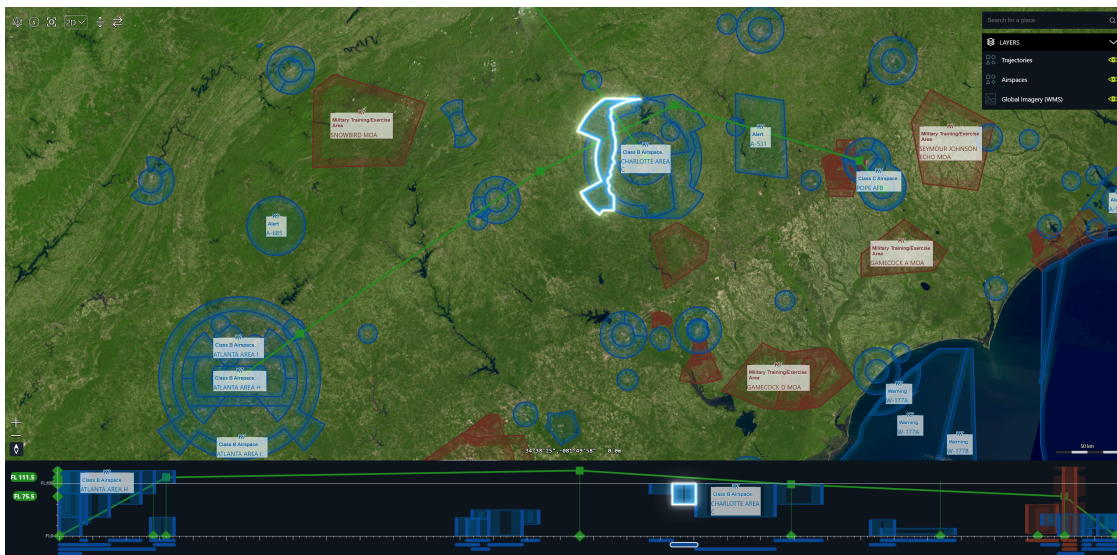
Sample code/documentation to get you started

The Timeline sample was adapted to use a different style for the sub-ticks. The sample code is available in the LuciadRIA release.

In the API, *AxisConfiguration.subTicks* now allows you to specify a function that dynamically sets the number of sub-ticks depending on the map and the pixel spacing.

Draw lines on the BorderGeoCanvas

The *BorderGeoCanvas* now supports drawing lines. This is useful for drawing the bounds of an airspace in the border of a vertical view, for example.

Using the newly supported lines in the Vertical View sample to highlight airspaces and their bounds

Sample code/documentation to get you started

For more information on decorating the axes of a cartesian view, see [Visualizing data in a non-geospatial view](#).

The Vertical View sample has been updated and now displays lines in the border instead of icons.

Use more Canvas map features on WebGL maps

Some map features were still available on Canvas maps only, and not yet on WebGL maps. In light of the future Canvas map deprecation, we are striving to bring those last few features to the WebGL map. In this section, we highlight the most important additions. For a full overview, we refer to the [release notes](#).

RasterImageModel support on WebGLMap

LuciadRIA 2025.0 makes *RasterImageModel* and *RasterImageLayer* available on the *WebGLMap*, with exactly the same functions they have in a Canvas map.

Support for single-image WMS requests

LuciadRIA 2025.0 now supports single-image WMS requests on *WebGLMap*. This means that you can use WMS services that support single-image requests only, such as the ones from the European Space Agency (ESA) Copernicus Open Access Hub.

Support for additional canvas labeling features

Starting from this release, the WebGL map also supports *InPathLabelStyle.inView* and *InPathLabelStyle.restrictToBounds*, working exactly as they always did on a Canvas map. They are supported on both 2D and 3D maps.

Other improvements

This LuciadRIA release also features a number of other improvements and bug fixes.

Convert from bounds to OrientedBox
[ShapeFactory.createOrientedBoxFromBounds](#) has been added. It converts bounds to its corresponding *OrientedBox*.

Cartesian references in WFS
WFSFeatureStore now supports local cartesian references. A typical use case are intelligence reports with vector overlays that are defined relative to an image.

World-sized styling in cartesian maps
You can now use world-sized styling in cartesian maps.

Support for LOCAL_CS in WKT
You can now use the *LOCAL_CS* tag while WKT parsing to create local cartesian references.

Support for minimum and maximum layer scale on cartesian maps
Cartesian maps now also support the *minScale* and *maxScale* properties of layers.

Support for adding label to a feature in OGC SLD
You can now add a label to a feature in OGC SLD using the *Mark* element. For more information, see [Use the vendor-specific WKN extension to style Mark elements](#).

Transformation chaining

A convenience method called *createChainedTransformation* has been added to *TransformationFactory* to allow you to chain transformations.

Point cloud picking

LuciadRIA 2025.0 introduces picking support for point clouds. This means that you can now programmatically select individual points in a point cloud and retrieve their properties. You can use the pre-existing *Map.pickAt* method, which now also works for point clouds. A new article, Point cloud picking, covers this functionality in more detail.

About Hexagon

Hexagon is the global leader in digital reality solutions, combining sensor, software and autonomous technologies. We are putting data to work to boost efficiency, productivity, quality and safety across industrial, manufacturing, infrastructure, public sector and mobility applications.

Our technologies are shaping production and people-related ecosystems to become increasingly connected and autonomous — ensuring a scalable, sustainable future.

Hexagon's Safety, Infrastructure & Geospatial division improves the resilience and sustainability of the world's critical services and infrastructure. Our solutions turn complex data about people, places and assets into meaningful information and capabilities for better, faster decision-making in public safety, utilities, defense, transportation and government.

Hexagon (Nasdaq Stockholm: HEXA B) has approximately 24,500 employees in 50 countries and net sales of approximately 5.4bn EUR. Learn more at hexagon.com and follow us @HexagonAB.

Links to third-party websites

This document may provide links to third-party websites for your convenience and information. Third-party websites will be governed by their own terms and conditions. Hexagon does not endorse companies or products to which it links.

Third-party websites are owned and operated by independent parties over which Hexagon has no control. Hexagon shall not have any liability resulting from your use of the third-party website. Any link you make to or from the third-party website will be at your own risk and any information you share with the third-party website will be subject to the terms of the third-party website, including those relating to confidentiality, data privacy and security.

Hexagon provides access to Hexagon international data and, therefore, may contain references or cross references to Hexagon products, programs and services that are not announced in your country. These references do not imply that Hexagon intends to announce such products, programs or services in your country.

Revisions

Hexagon reserves the right to revise these terms at any time. You are responsible for regularly reviewing these terms. Your continued use of this document after the effective date of such changes constitutes your acceptance of and agreement to such changes.

Questions

Contact us with any questions regarding these terms.