Release guide

LuciadCPillar 2024.0

28 June 2024

Contents

About this release

The 2024.0 release of LuciadCPillar adds support for vector tiles, a new data type that boosts the performance of contextual vector layers. Moreover, with technology updates and the completion of multilanguage documentation, we continue to improve the developer experience. This release also includes some format-specific improvements.
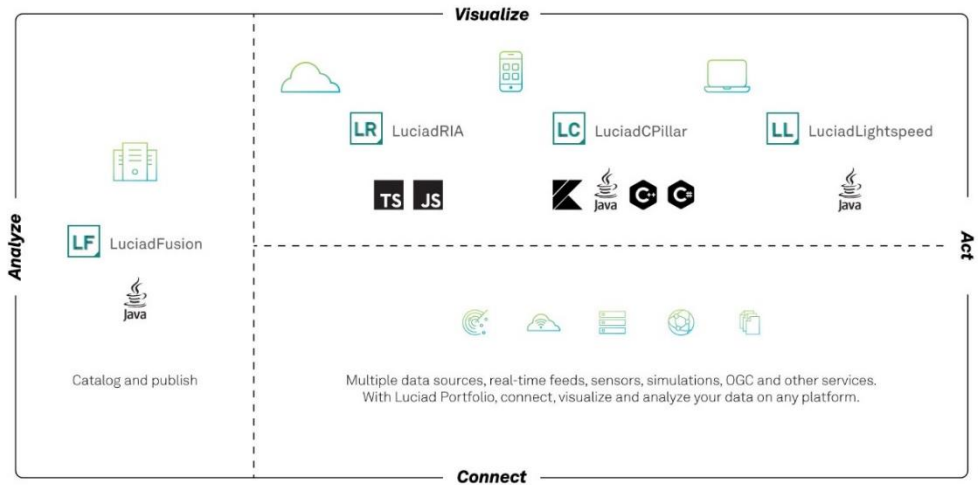

Figure 1: The Luciad portfolio

Benefits of new features

This section provides an overview and introduction of new features that are available in LuciadCPillar 2024.0. The new features are available for all the supported platforms and programming languages currently offered for LuciadCPillar.
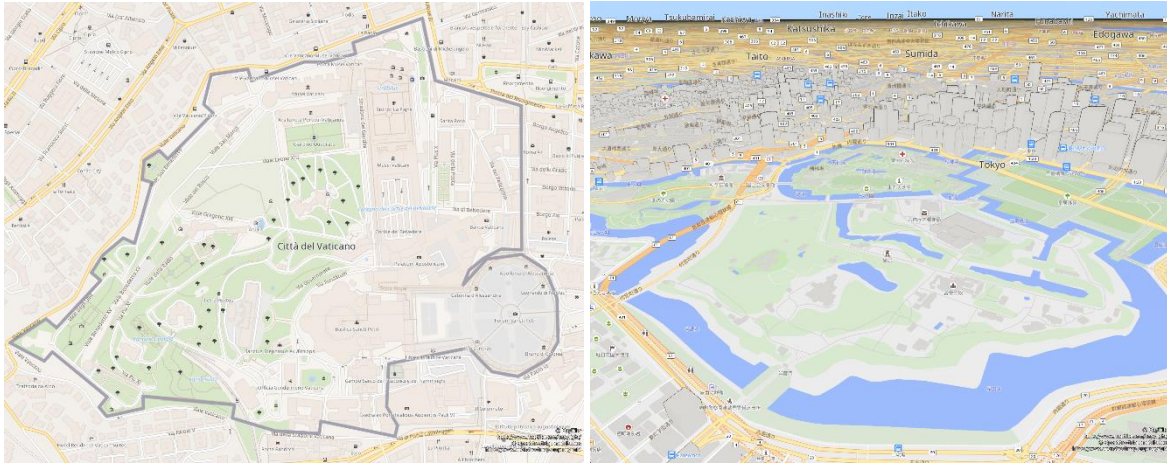

Figure 2: LuciadCPillar can now connect to and display vector tiles data

Support for vector tiles

A lot of business data is modeled as vector data or feature data. Most of this data can be loaded as-is. For larger data sources, the use of scale ranges and smart filtering[1] ensures a smooth experience. However, a few types of vector data cover large areas, offer very detailed information and serve as context or background information. Because of the detail and large size of individual objects, filtering and scale ranges are not sufficient. Some examples include world shorelines, country borders or height contour lines.

In a process that is similar to how 3D data is processed into 3D Tiles, these contextual feature layers can be processed into vector tiles or feature tiles. In LuciadCPillar we use the term MBTilesFeatureLayer. A common format is MBTiles[2]. MBTiles is a container format for tile sets of 2D data. The specification is open, which makes it a good exchange format. The format is based on the SQLite database engine. MBTiles data can contain raster or vector tiles. See example in Figure 3.

LuciadCPillar now supports MBTiles data, either from files or streamed as MBTiles service. The MBTiles model decoder allows MBTiles data to be accessed according to the MBTiles 1.3 and TileJSON 3.0.0 specifications. Both raster and vector data are supported.

---

[1] The article "Visualizing feature data on the map" explains this
[2] https://docs.mapbox.com/help/glossary/mbtiles/

Note: LuciadFusion can serve MBTiles raster and vector data as a MBTiles service. This new service type was added to LuciadFusion in the 2024.0 release.

Loading vector tiles

Vector tiles are represented by a FeatureTileModel. The data can be styled using the MapBox Style format[3]. You can use the MBTilesFeatureLayer to apply such styling. Style files can be loaded from the file system or from a web service.

For this release, we started with a sensible subset of the MapBox Style format and will extend this over time. The API documentation of MBTilesFeatureLayer builder documents the supported styles. Don't hesitate to contact product support if your data requires a non-supported style that you would like to see added.

Loading raster tiles

Loading raster MBTiles in LuciadCPillar is straightforward and like loading any of the already supported raster data formats.

Sample code and documentation to get you started

The guide "Working with MBTiles data" in the space "Data formats: MBTiles" is your starting point in the documentation. See Figure 4. You can try out MBTiles data using the Data Formats sample.

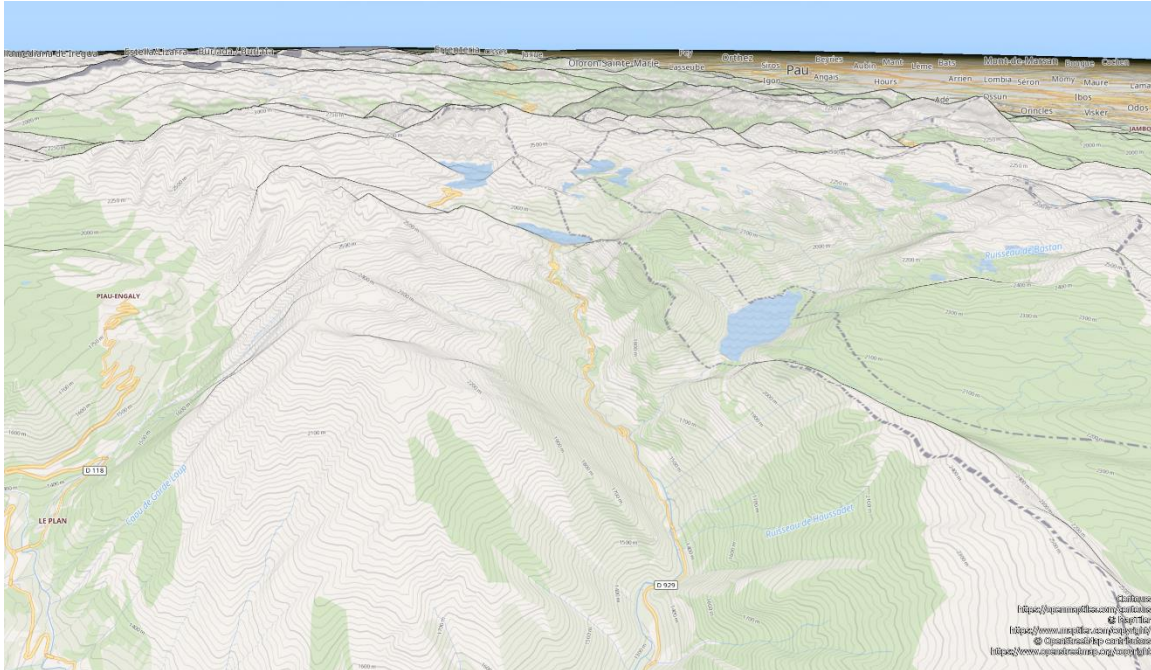[3] https://docs.mapbox.com/style-spec/guides/

Figure 3: Contour lines are a typical example of data that benefits from vector tiling. This picture depicts a combination of Open Street Map data and contour lines that were visualized using LuciadCPillar 2024.0.

OGC 3D Tiles 1.1 connector

We have supported OGC 3D Tiles since version 2020, starting with support for version 1.0. In the meantime, the new version of 3D Tiles has been adopted as community standard[4]. Previously referred to as "3D Tiles Next," Version 1.1 of the 3D Tiles Community Standard is designed for streaming high-resolution, semantically rich 3D geospatial data to the metaverse. OGC 3D Tiles 1.1 promotes several 3D Tiles 1.0 extensions to 'core' and introduces new glTF™ extensions for fine-grained metadata storage.

With LuciadCPillar 2024.0, you can now visualize OGC 3D Tiles 1.1.

The specification of OGC 3D Tiles 1.1 is quite rich. We have not yet received test data for each new feature, so the currently supported functionality includes:

- Implicit tiling
- GLB/glTF as payload
- Support for a number of glTF 2.0 extensions

Note: glTF data with the supported extensions can also be loaded as 3D icons.

We will continue to complete our support for OGC 3D Tiles 1.1 in upcoming releases, driven by data that we notice and that is reported by you. For example, metadata is not supported, because we have not yet encountered sufficient OGC 3D Tiles 1.1 test data including metadata.

Sample code/documentation to get you started
The documentation topic "Data Formats: OGC 3D Tiles" and the reference documentation for Ogc3DTilesModelDecoder have been updated to cover the supported OGC 3D Tiles 1.1 functionality.

Improved developer experience
As we are offering an SDK, it is important that you can intuitively work with LuciadCPillar, easily find the right information and smoothly integrate LuciadCPillar into your system. For this release we focused on the following areas.

Complete C# API documentation
This release completes the initiative to offer focused documentation. Based on a choice of programming language, you can see Desktop C++, Desktop C# or Android developer documentation. Developer articles adapt to the selected profile, with code snippets in either C++, C# or Kotlin. API reference documentation is available in C++, C# or Java. All code snippets and articles are now available for the three profiles, where applicable. Refer to Figure 4 for an illustration.

---

https://www.ogc.org/press-release/ogc-adopts-3d-tiles-v1-1-as-community-standard/

Figure 4: The code snippets adapt to the programming language selected in the drop-down box at the top of the page.

NuGet packaging

To support your DevOps setup, the LuciadCPillar SDK will now also be delivered as a NuGet[5] package for .NET development. This simplifies including LuciadCPillar into your C# project and allows you to publish LuciadCPillar to your own private NuGet feed. Refer to the "Installing LuciadCPillar for C# development on Windows" tutorial for more information.

Samples now use Qt 6

The C++ samples in this LuciadCPillar release have been updated from Qt 5 to Qt 6, which is more modern than its predecessor: Qt 6 is based on C++ 17 and introduces a new graphics architecture that, among others, leverages the GPU to render UI components and

---

[5] www.nuget.org

renders a sharp UI regardless of screen size and resolution. The articles "Installing LuciadCPillar for C++ development on Windows" and "Installing LuciadCPillar for C++ development on Linux" have been upgraded to reflect the use of Qt 6. The required version of Qt is documented in the article "Hardware and software requirements." Currently, you need Qt 6.5.x (LTS). The online installer can be found here: https://www.qt.io/download.

Upgrade considerations

Dependencies for LuciadCPillar development on Linux systems have been updated to glibc (GNU C Library) 2.28 and gcc 11.

The gcc 11 update is an update for the compiler and affects you as a developer. This may require code changes on your side but will not affect the deployment environment. This update will offer a better and more stable developer experience when debugging.

As previously announced (see Figure 5), the supported Android level has now been updated to level 30. The next update will be with our 2025.0 major release.

LuciadCPillar version	2024.0	2024.1	2025.0	2025.1	2026.0	...
Supported API version	API level 30 (Android 11)	API Level 30 (Android 11)	API Level 31/32 (Android 12, 12.1)	API Level 31/32 (Android 12, 12.1)	API Level 33 (Android 13)	...

Figure 5: The section "Requirements for Android development" in the product documentation announces planned evolutions.

Other improvements

- **OGC GeoPackage encoder:** Support for GeoPackage is now complete – you can create a new GeoPackage file, open an existing one and store modifications. All information is available in the topic "Data formats: OGC GeoPackage," and the new article "Encode feature data to GeoPackage."
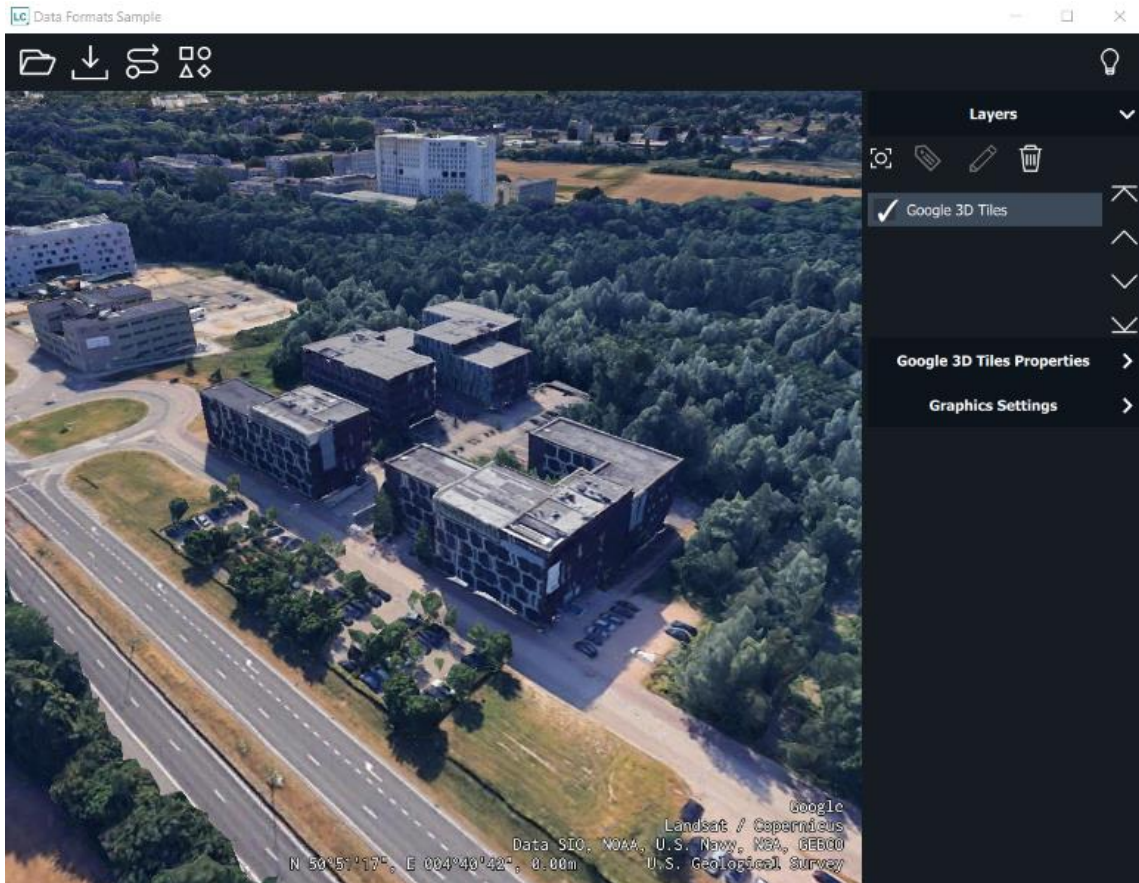

Figure 6: A LuciadCPillar view of the Leuven Hexagon office in Google 3D Tiles

Attribution: Some services, such as HERE Maps[6], require you to display attributions when using their data on the map. LuciadCPillar now keeps track of attributions from all layers that provide attribution data and exposes an API to retrieve these attributions. For more info, please consult the article "How to provide and retrieve attribution data." You can also have a look at the HERE maps support in the "Data formats" sample and the articles "How to provide access to HERE Maps data" and "Visualizing Google 3D Tiles" for an illustration.provide attribution data and exposes an API to retrieve these attributions. For more info, please consult the article "How to provide and retrieve attribution data." You can also have a look at the HERE maps support in the "Data formats" sample and the articles "How to provide access to HERE Maps data" and "Visualizing Google 3D Tiles" for an illustration.

---

[6] https://maps.here.com

- The following model decoders now support the WebP file format:

  - [GdalModelDecoder](#)
  - [GeoPackageModelDecoder](#)
  - [WmsModelDecoder](#)
  - [WmtsModelDecoder](#)
  - [Ogc3DTilesModelDecoder](#)

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.

Copyright

Terms of use

Disclaimers

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.