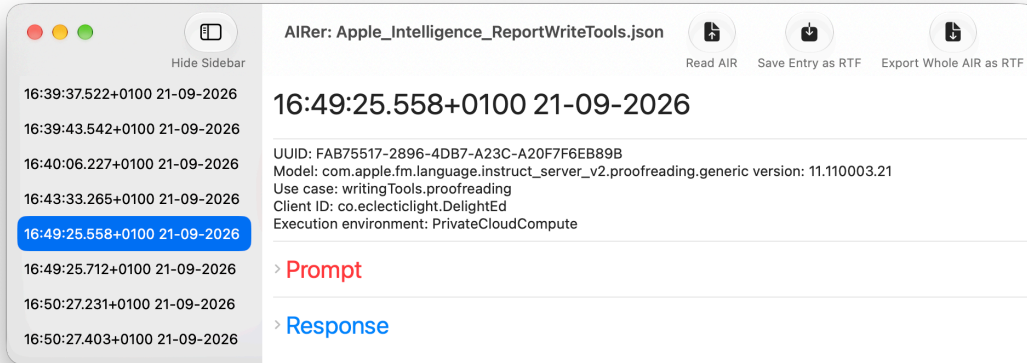


Start

→ [Start](#)



AIRer is a simple app to browse Apple Intelligence Report (AIR) files saved in JSON format. Although these are full of useful and important information, they aren't easy to browse using a text or JSON editor. AIRer should make their contents more accessible.

This is its first full release, and requires macOS 27 Golden Gate to run. It's designed to work with AIR files written by macOS, iOS and iPadOS 27.x, although it may also be able to use those written by previous versions of macOS.

To get started, read how to → [open and browse](#) an AIR file first.

AIRer's app icon is based on the alchemical symbol used for air.

→ [Model requests](#)

→ [PCC requests](#)

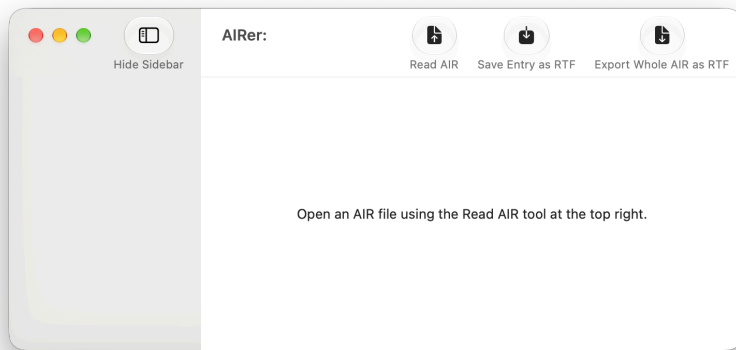
→ [Save & export as Rich Text](#)

→ [Technical information](#)

→ [Start](#)

Open & browse

If you haven't already saved an AIR, open **Privacy & Security** settings on your Mac or device, locate the **Apple Intelligence Report** item towards the end of the list, open it, ensure it has been turned on with either **Report Duration** set, and click/tap on the **Export Activity** button to save the AIR as a JSON file with the extension **.json**.



Open that file by clicking on the **Read AIR** tool at the top right of AIRer's main window. Select the file, and click **Open**.

If the AIR file is empty, you won't see any items appear in the left sidebar. If it contains records of Model and PCC requests, those will then be listed at the left in time order, with the oldest at the top.

To open one of those requests, select it in the list at the left. Its contents will then be loaded into the main part of the window.

You can adjust the width of the sidebar by dragging the divider at its right, and hide and show the sidebar using the tool at the top. Each section of text in the main window can be selected and copied as plain or rich text.

→ [Model requests](#) start with several lines of text in black/white, followed by a long **Prompt** in red and a **Response** in blue.

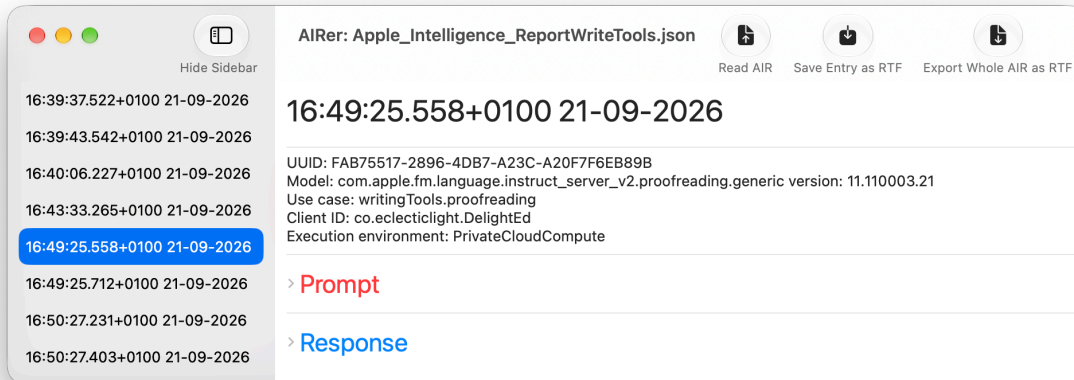
→ [PCC requests](#) start with a couple of lines in black/white, followed by brief **Pipeline parameters** in blue.

→ [Save & export as Rich Text](#)

→ [Technical information](#)

→ [Start](#)

Model requests



These contain the details of AI requests processed off-device by Private Cloud Compute (PCC).

The two long text sections, **Prompt** and **Response**, can be opened and closed by clicking on the disclosure button at the left of their title, as shown here.

Fields shown in each include:

- **UUID**, its unique ID, which may match that of a PCC request
- **Model**, the full name of the model to be used, together with its version number
- **Use case**, the AI feature to be used
- **Client ID**, the name of the app or service making the request
- **Execution environment**, should always be PCC, until Google environments are also used
- **Prompt**, shown in red, the full prompt given to the model
- **Response**, shown in blue, the model's response to that prompt.

→ [Open & browse](#)

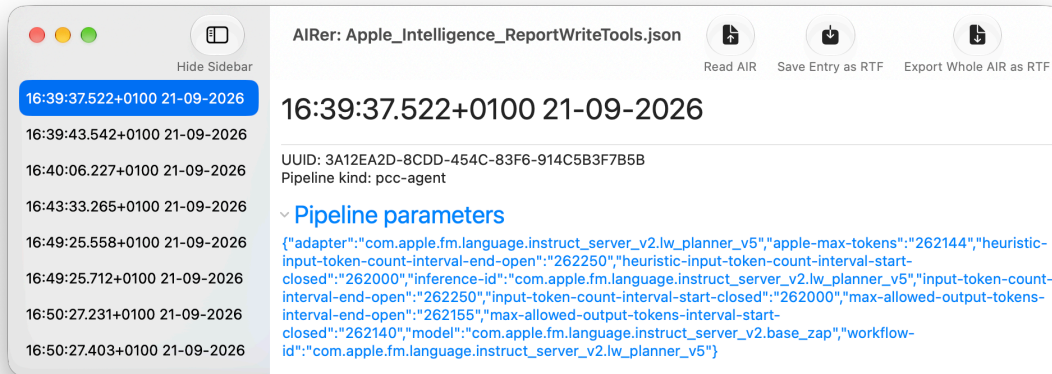
→ [PCC requests](#)

→ [Save & export as Rich Text](#)

→ [Technical information](#)

→ [Start](#)

PCC requests



These contain the metadata for requests processed off-device by Private Cloud Compute (PCC).

The longer text section, **Pipeline parameters**, can be opened and closed by clicking on the disclosure button at the left of its title.

Fields shown in each include:

- **UUID**, its unique ID, which may match that of a Model request
- **Pipeline kind**, should always be `pcc-agent` for PCC requests
- **Pipeline parameters**, shown in blue, useful additional information that can include token allowance and use, and further details of the model used.

→ [Open & browse](#)

→ [Model requests](#)

→ [Save & export as Rich Text](#)

→ [Technical information](#)

Save and export as Rich Text [→ Start](#)

AIRer:



Read AIR



Save Entry as RTF



Export Whole AIR as RTF

To export the contents of one of the requests to access in another app, select the item in the list at the left so it's displayed in the main part of the window, and click the **Save Entry as RTF** tool. You will then be invited to choose the location and name of the Rich Text file to contain the contents. The default name suggested is based on the entry's timestamp.

To export the whole contents of the AIR file, click the **Export Whole AIR as RTF** tool at the top right. You will then be invited to choose the location and name of the Rich Text file to contain the contents.

Rich Text files follow the same colour conventions as the main part of the window, and can be opened using TextEdit, [DelightEd](#) or another Rich Text editor.

[→ Open & browse](#)

[→ Model requests](#)

[→ PCC requests](#)

[→ Technical information](#)

→ [Start](#)

Technical Information

AIRer decodes the original JSON data written to the AIR and extracts fields that are most useful and readable. It doesn't attempt to convert any that are partly or completely encoded in Base64, nor contents that aren't in JSON but in Protobuf format. If you want to access those, you'll need to set up a PCC [Virtual Research Environment](#) and use the `pccvre` `attestation` subcommands as documented by Apple, and [explained here](#).

AIRer itself doesn't use any AI-related code, so should run normally in Golden Gate environments like external disks and VMs that don't support AI features.

→ [Open & browse](#)

→ [Model requests](#)

→ [PCC requests](#)

→ [Save & export as Rich Text](#)

Change list

1.0 (10) full release:

- added text content select and copy
- changed order of dates for consistency, and added decimal seconds
- added export whole AIR to RTF
- put longer text fields into disclosure groups.

1.0 (7) beta:

- first release.

2 October 2026.