

Uploading and Parsing Code in the Workspace

August 26, 2026

Opening the parsing workspace and preparing your upload

In Atloria, start from your project workspace and open the **Code Parsing Workspace** area used for code upload and analysis. If you are entering from a project home screen, first make sure you are in the correct project before you begin. This matters because uploaded files and parsing results stay tied to the project workspace where you add them.

Look for the upload entry points on the parsing screen. Depending on what you want to analyze, Atloria may let you add source material through an **Upload** action for files, a folder or archive import option, and a snippet area for pasted code. Before you click anything, decide what kind of source you are bringing in:

- A single source file when you want to inspect one file on its own
- A compressed project archive when you want Atloria to preserve folders and file relationships
- A short pasted snippet when you only need quick analysis of a small section of code

If you are uploading multiple files, organize them first so the folder structure is clear. When Atloria shows a workspace tree or file list, that structure helps you move between related files and understand how pieces fit together. A clean upload also makes later documentation work easier because filenames and paths stay recognizable.

Check any visible controls at the top or side of the workspace before starting. In some project setups, you may see a project selector, language choice, parser choice, or a documentation-related context setting. Match those options to the source material you plan to upload so the parsing results are easier to review.

[SCREENSHOT: Code Parsing Workspace showing upload options, project selection, and workspace file tree]

Uploading files, folders, and archives into the workspace

Once you are in the **Code Parsing Workspace**, use the main **Upload** action to add your source files. If you are working with one file, choose that file from your computer and wait for it to appear in the workspace file list or tree. After the upload finishes, confirm that the filename is visible in the left-side navigator or main file area before moving on to parsing.

For a larger codebase, use the folder or archive upload option if it is available on your screen. This is the better choice when your documentation depends on related files being kept together. After the import completes, review the workspace tree and make sure nested folders are still shown in the correct order. You should be able to expand folders and see the same path layout you expect from the original source material.

1. Open the **Code Parsing Workspace** in the correct project.
2. Click **Upload**.
3. Select either an individual file, a folder import option, or a compressed archive.
4. Wait for the upload progress indicator or status message to finish.
5. Check the workspace tree to confirm the uploaded items appear in the expected location.

As files are added, watch for progress indicators, status badges, or completion messages. These help you tell the difference between a file that is still being processed and one that is ready for parsing. If you upload more than once into the same project area, review the file list carefully. When Atloria shows duplicate names, pay attention to any replacement prompt or updated entry in the workspace so you do not accidentally work from the wrong version.

[SCREENSHOT: Upload dialog with a visible progress indicator and completed files listed in the workspace tree]

Pasting and parsing code snippets for quick analysis

When you do not need to upload a full file or project archive, use the snippet input area in the **Code Parsing Workspace**. This is the fastest option for testing a small block of code, checking how Atloria reads a configuration fragment, or reviewing one focused section before adding larger source material.

Open the snippet area or paste dialog and place your code into the text box. Keep the snippet focused and complete enough to make sense on its own. For example, if you only paste part of a block without the surrounding lines it depends on, the parsing

result may be incomplete. A slightly larger snippet often gives better results than a fragment that starts or ends in the middle of a structure.

Before you run the parse action, choose the matching language or parser option if Atloria shows one. This step is important because the same text can be interpreted differently depending on the selected language. If the language setting does not match the pasted content, the result may miss important structures or show only partial output.

1. Open the snippet input area in the workspace.
2. Paste the code block you want to inspect.
3. Select the matching language or parser option.
4. Click **Parse**.
5. Review the result and decide whether to keep or discard it.

Snippet parsing is useful when you only need to inspect one focused piece of source material instead of a full project tree. After reviewing the result, save it if you want to return to it during documentation work. If the snippet was only for a quick check, discard it so the workspace stays tidy and easier to navigate.

[SCREENSHOT: Snippet input panel with pasted code, language selection, and Parse button]

Running parsing and reading the results in the workspace

After your files or snippets are in place, start parsing from the workspace controls. In Atloria, this usually begins with a visible **Parse** action on the current file, snippet, or uploaded source area. Once you start, watch the status flow as the item moves from uploaded content to parsed output. A progress state, processing badge, or completion message helps confirm when the results are ready to inspect.

When parsing finishes, use the workspace panels to move through the output. Atloria may show a file tree on one side, the source content in a main panel, and a results or detail panel beside it. This layout is especially helpful when you want to compare the original source with the extracted structures Atloria found during parsing.

1. Select the uploaded file, folder entry, or snippet you want to analyze.
2. Click **Parse** from the workspace controls.
3. Wait for the processing status to change to a completed state.
4. Open the parsed result in the result pane or detail panel.
5. Move between files in the workspace tree to review related output.

As you read the results, focus on the parts most useful for technical documentation. Depending on what Atloria displays, this may include file-level structures, named items, grouped definitions, comments, or configuration sections. If the workspace shows a breakdown view, use it to understand how a file is organized before you start writing or generating documentation from it.

This stage is not just about confirming that parsing worked. It is where you decide whether the uploaded source gives enough detail for documentation tasks such as reference pages, technical summaries, or structured explanations.

[SCREENSHOT: Parsed workspace view showing source content beside extracted structures and details]

Reviewing parsed output for documentation-ready details

Before you rely on parsed results for documentation, compare what Atloria shows with the original source layout. Start with the basics: filenames, folder paths, and the visible hierarchy in the workspace tree. If those do not match the source material you uploaded, the documentation you create later may be harder to organize or verify.

Next, inspect the extracted details that writers usually need. In Atloria, this may appear as named structures, signatures, relationships between items, comments, or configuration definitions. Review these carefully and check whether the parser captured the important parts of the source instead of only a partial outline. If a file should contain several major sections but only one appears in the results, that is a sign to review the upload or parse settings again.

Look closely at any comments, inline descriptions, or metadata surfaced in the result view. These details are often the most helpful starting point for technical documentation because they explain intent, expected behavior, or usage context. If Atloria provides a preview or inspection view, use it to judge whether the parsed output is clear enough to support documentation generation without heavy cleanup.

A good parsed result usually has these qualities:

- The file and folder names match the original source
- The visible hierarchy is easy to follow
- Key structures are listed clearly
- Comments or descriptive text appear where expected
- Related items are grouped in a way that makes sense for documentation

If the result looks complete and readable, you are in a strong position to use it in later documentation workflows such as reference review or content generation.

[SCREENSHOT: Parsed result detail view highlighting filenames, extracted structures, and comments]

Common issues when uploads or parsing results do not look right

If an upload finishes but nothing appears in the workspace tree, first confirm that you are still in the correct project workspace. In Atloria, uploads belong to the project area where you added them, so switching views can make it seem like files disappeared. Refresh the current project view and check the file tree again before uploading a second time.

When snippet parsing returns incomplete results, the most common cause is a mismatch between the pasted code and the selected language or parser option. Reopen the snippet area, verify the language choice, and make sure the pasted block includes enough surrounding context. A very short fragment may not give Atloria enough information to show a useful structure.

Archive uploads can also look wrong if the original archive is arranged unexpectedly. If folder relationships seem to be missing, review the imported tree and compare it with the source archive on your computer. If Atloria expands nested directories differently than you expected, recheck the archive contents before uploading again.

If parsed output is missing important symbols, comments, or relationships, try these checks:

- Re-run parsing on the original file rather than an edited copy
- Confirm you selected the correct language or parser option
- Review whether the source file may be incomplete or malformed
- Compare the parsed result with the source view to see exactly what is missing

A missing detail does not always mean the upload failed. Sometimes the source itself lacks enough context for Atloria to extract everything cleanly. When that happens, use a fuller file, a better-organized archive, or a more complete snippet and run parsing again.

[SCREENSHOT: Workspace showing an upload completed state alongside a partial or incomplete parsing result]

Overview

The **Code Parsing Workspace** in Atloria gives you a place to bring in source material, run parsing, and inspect the results before using them in technical documentation work. You can work with full files, larger project uploads, or quick pasted snippets depending on how much source material you need to review.

This workflow is most useful when you want to:

- Analyze code before creating technical documentation
- Check file structure and relationships inside a project
- Review extracted details such as named structures, comments, and configuration sections
- Test a small snippet without uploading a full codebase

The main screens and actions in this workflow are straightforward:

Workspace area	What you use it for
Upload	Add files, folders, or archives to the workspace
Snippet input	Paste a short block of code for quick parsing
Parse	Start analysis on uploaded content or a pasted snippet
Workspace tree	Browse files and folders after upload
Result pane	Review parsed output and extracted details

As you work, keep your goal in mind. If you only need to inspect one small section, the snippet area is usually enough. If you need relationships across multiple files, upload a structured set of files or an archive so Atloria can preserve the project layout. The better your source material is organized at the start, the easier it is to review and reuse the parsed output later.

For a broader view of supported parsing coverage before you upload, see [Viewing Supported Languages and Parser Coverage](#).

Prerequisites

Before you upload anything into the **Code Parsing Workspace**, make sure you have the right project open in Atloria and the source material ready in a format you can review easily. This helps avoid duplicate uploads, misplaced files, and parsing results attached to the wrong project.

You should have:

- Access to the Atloria project where you want to work
- Source material ready as one of the following:
 - an individual code file
 - a compressed project archive
 - a short code snippet for pasting
- A basic understanding of which language your source uses so you can choose the right parsing option if Atloria asks for it
- Source files organized clearly enough that folder names and file paths still make sense after upload

It also helps to decide in advance what kind of result you need. Use a single file when you are checking one source file in isolation. Use an archive or grouped upload when documentation depends on file relationships. Use a snippet when you want a quick answer without adding a larger project set to the workspace.

Before starting, verify these practical points:

- You are signed in and already inside the correct project workspace
- The files you plan to upload are the versions you actually want documented
- Any archive you use has a folder structure worth preserving
- Your snippet is complete enough to be understood on its own

If you need help getting into Atloria or reaching the right project area first, see [Accessing and Registering Your Atloria Account](#) and [Understanding Project Navigation and Linked Workspaces](#).

The next step is [Managing Code Parsing Workspace Sessions](#).