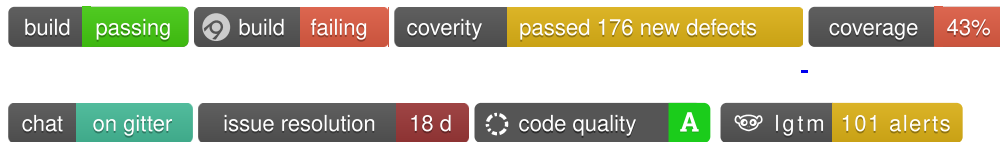


# Open Asset Import Library (assimp)

A library to import and export various 3d-model-formats including scene-post-processing to generate missing render data.

## Current project status



APIs are provided for C and C++. There are various bindings to other languages (C#, Java, Python, Delphi, D). Assimp also runs on Android and iOS.

Additionally, assimp features various **mesh post processing tools**: normals and tangent space generation, triangulation, vertex cache locality optimization, removal of degenerate primitives and duplicate vertices, sorting by primitive type, merging of redundant materials and many more.

This is the development repo containing the latest features and bugfixes. For productive use though, we recommend one of the stable releases available from [Github Assimp Releases](#).

Monthly donations via Patreon:



One-off donations via PayPal:

Donate

Please check our Wiki as well: <https://github.com/assimp/assimp/wiki>

If you want to check our Model-Database, use the following repo: <https://github.com/assimp/assimp-mdb>

## Supported file formats

### Importers:

- 3D
- [3DS](#)
- [3MF](#)
- AC
- [AC3D](#)
- ACC
- AMJ
- ASE
- ASK
- B3D

- [BLEND](#)
- [BVH](#)
- CMS
- COB
- [DAE/Collada](#)
- [DXF](#)
- ENFF
- [FBX](#)
- [glTF 1.0](#) + GLB
- [glTF 2.0](#)
- HMB
- IFC-STEP
- IRR / IRRMESH
- [LWO](#)
- LWS
- LXO
- MD2
- MD3
- MD5
- MDC
- MDL
- MESH / MESH.XML
- MOT
- MS3D
- NDO
- NFF
- [OBJ](#)
- [OFF](#)
- [OGEX](#)
- [PLY](#)
- PMX
- PRJ
- Q3O
- Q3S
- RAW
- SCN
- SIB
- SMD
- [STP](#)
- [STL](#)
- TER
- UC
- VTA
- X
- [X3D](#)
- XGL
- ZGL

Additionally, some formats are supported by dependency on non-free code or external SDKs (not built by default):

- [C4D](https://github.com/assimp/assimp/wiki/Cinema4D-&-Melange) (<https://github.com/assimp/assimp/wiki/Cinema4D-&-Melange>)

**Exporters:**

- DAE (Collada)
- STL
- OBJ
- PLY
- X
- 3DS
- JSON (for WebGL, via <https://github.com/acgessler/assimp2json>)
- ASSBIN
- STEP
- glTF 1.0 (partial)
- glTF 2.0 (partial)
- 3MF ( experimental )
- FBX ( experimental )

## Building

Take a look into the <https://github.com/assimp/assimp/blob/master/Build.md> file. Our build system is CMake, if you used CMake before there is a good chance you know what to do.

## Ports

- [Android](#)
- [Python](#)
- [.NET](#)
- [Pascal](#)
- [Javascript \(Alpha\)](#)
- [Unity 3d Plugin](#)
- [JVM](#) Full jvm port (current [status](#))
- [HAXE-Port](#) The Assimp-HAXE-port.

## Other tools

[open3mod](#) is a powerful 3D model viewer based on Assimp's import and export abilities.

## Repository structure

Open Asset Import Library is implemented in C++. The directory structure is:

```
/code      Source code
/contrib   Third-party libraries
/doc       Documentation (doxysource and pre-compiled docs)
/include   Public header C and C++ header files
/scripts   Scripts used to generate the loading code for some formats
/port      Ports to other languages and scripts to maintain those.
/test      Unit- and regression tests, test suite of models
/tools     Tools (old assimp viewer, command line `assimp`)
/samples  A small number of samples to illustrate possible
           use cases for Assimp
```

## Where to get help

For more information, visit [our website](#). Or check out the `./doc -` folder, which contains the official documentation in HTML format. (CHMs for Windows are included in some release packages and should be located right here in the root folder).

If the docs don't solve your problem, ask on [StackOverflow](#). If you think you found a bug, please open an issue on Github.

For development discussions, there is also a (very low-volume) mailing list, *assimp-discussions* ([subscribe here](#)).

Open Asset Import Library is a library to load various 3d file formats into a shared, in-memory format. It supports more than **40 file formats** for import and a growing selection of file formats for export.

chat on gitter

And we also have a Gitter-channel:Gitter

## Contributing

Contributions to assimp are highly appreciated. The easiest way to get involved is to submit a pull request with your changes against the main repository's `master` branch.

## License

Our license is based on the modified, **3-clause BSD**-License.

An *informal* summary is: do whatever you want, but include Assimp's license text with your product - and don't sue us if our code doesn't work. Note that, unlike LGPLed code, you may link statically to Assimp. For the legal details, see the `LICENSE` file.

## Why this name

Sorry, we're germans :-), no english native speakers ...