

Hydrilla on-disk data format

This page explains the format of built Hydrilla custom site resources stored in the filesystem.

You might also want to read about the format of [Hydrilla source packages](#) or about [Hydrilla HTTP JSON API](#).

How Hydrilla loads packages to serve

Hydrilla expects a serveable directory to be specified in its configuration file (under the key "malcontent-dir"). It then processes the following of its subdirectories.

file/ subdirectory

All files used by resources (either directly or as part of their source packages' license info) are stored under file/sha256 with names being their SHA256 sums.

source/ subdirectory

Every resource and mapping in Hydrilla gets built from a source package. During building of each such source package a JSON description of it gets generated. All descriptions are stored under source with filenames of the format <source_package_identifier>.json.

JSON descriptions of sources have structure as presented in the [API description](#). They are served directly from disk files. In the future they will be accompanied by cryptographic signatures.

Additionally, for each source package a zip archive with its contents is expected to be present under source with filename of the format <source_package_identifier>.zip.

resource/ subdirectory

All resource JSON descriptions are stored under resource with filenames of the format <resource_identifier>/<resource_version_without_revision>.json. For example, definitions of versions 0.1.0-4 and 0.1.1-1 of resource hello would be stored as hello/0.1 and hello/0.1.1, respectively. Both would be under subdirectory hello of resource.

JSON definitions of resources have structure as presented in the [API description](#). They are served directly from disk files. In the future they will be accompanied by cryptographic signatures.

mapping/ subdirectory

All mappings JSON descriptions are stored under mapping with filenames of the format <mapping_identifier>/<mapping_version>.json. For example, definitions of versions 0.1.0 and 0.1.1 of resource hello would be stored as hello/0.1 and hello/0.1.1, respectively. Both would be under subdirectory hello of mapping.

JSON descriptions of mappings have structure as presented in the [API description](#). They are served directly from disk files. In the future they will be accompanied by cryptographic signatures.

Populating Hydrilla serveable directory

Hydrilla package builder writes generated files in a way they are suitable to be loaded by Hydrilla. Given multiple directories already populated with built package files, it is also possible to recursively copy one's all contents into the other. In this case the user is responsible for avoiding clashes in identifiers of resources, mappings and sources.