

.Reference card - Bundle format v1

On this page we give an overview of the bundle format in terms of its structure and bundle.json file.

On this page:

- On this page we give an overview of the bundle format in terms of its structure and bundle.json file.
- JSON schema
- Bundle properties
 - Dependencies

JSON schema

This JSON schema defines the structure for describing a bundle to exchange information between components.

 Download link: [JSON Schema bundle.json](#)

Bundle properties

Property	Description	Type	Examples
id	A string identifier for the bundle, adhering to the reverse domain name notation (RDNN) to ensure a unique namespace. RDNN consists of alphanumeric characters and periods.	String	"com.example.mybundle"
name	A string representing the name of the bundle. It should be concise and descriptive.	String	
version	A string specifying the version of the bundle, following semantic versioning (SemVer) principles. The format should adhere to major.minor.patch.	String	"1.0.0"
description	A string providing a detailed description of the bundle, outlining its purpose, functionality, and key features.	String	
includes	An array of strings indicating the relevant folders within the bundle that should be considered during the import process. Each string represents a folder path.	Array of strings	
createdby	A string denoting the name of the individual or entity responsible for creating the bundle.	String	
datecreated	A string representing the date of the bundle creation, following the 'date' format.	String with format "date"	"2024-04-15"
dependencies	An array of objects representing dependencies on other bundles. Each object must have 'id' and 'version' properties. See below for a description of these properties.	String	

Dependencies

Property	Description	Type	Examples
id	The identifier of the referred bundle.	String	"com.example.mybundle"
version	The version of the referred bundle, adhering to semantic versioning.	String	"1.0.0"