

Dependency Manifest

[Go up to Terms and Definitions](#)

Discussion

A Dependency Manifest is a machine-processable record that identifies the [Dependencies](#) associated with a defined build, deployment, artifact, environment, or transfer bundle.

The manifest may record dependency names, versions, types, source locations, integrity values, licenses, provenance, relationships, capture times, storage locations, and consuming artifacts or processes.

A Dependency Manifest supports verification, comparison, transfer, import, audit, and reproduction by making the captured dependency set explicit.

Within Crucible, a Dependency Manifest may describe the contents of a [Dependency Store](#) or [Transfer Bundle](#).

Definition

machine-processable record that identifies the dependencies associated with a defined build, deployment, artifact, environment, or transfer bundle

Source

Generalized from software bills of materials, package manifests, dependency lock files, supply-chain records, and artifact inventories and specialized for the Crucible architecture and operational model.

Note

A Dependency Manifest identifies Dependencies and their relevant characteristics. It does not necessarily contain the dependency content itself.

The completeness of a Dependency Manifest depends on its defined scope. A build manifest, a runtime manifest, and a transfer-bundle manifest may identify different but related dependency sets.

Example

A Crucible Dependency Manifest records the installation medium, operating-system packages,

provider packages, container images, infrastructure repository revisions, compliance artifacts, and integrity values included in a Transfer Bundle.

© 2026 Dido Solutions, Inc. and Jackrabbit Consulting, Inc.

From:
<https://wiki.didosolutions.com/> - **Dido Solutions, Inc Wiki**

Permanent link:
https://wiki.didosolutions.com/dido/99_annexes/annex-b-terms-and-definitions/d/dependency_manifest

Last update: **2026/07/18 12:33**

