

Transfer Bundle

[Go up to Terms and Definitions](#)

Discussion

The Crucible Concept of Operations establishes the Transfer Bundle as the transferable collection produced within a [Connected Environment](#) and imported into a [Disconnected Environment](#).

A Transfer Bundle is a controlled collection of [Artifacts](#), [Dependencies](#), records, manifests, integrity information, and supporting [Evidence](#) assembled for transfer between environments.

A Transfer Bundle may contain:

- Installation media
- Software packages
- Operating-system update packages
- [Machine Images](#)
- [Container Images](#)
- Provider-specific content
- Infrastructure source content
- Compliance Artifacts
- Configuration
- Records
- Manifests
- Integrity information
- Supporting Evidence
- A [Dependency Manifest](#)

A Transfer Bundle facilitates controlled movement across an organizational, network, security, jurisdictional, or [Air Gap](#) boundary while preserving the identity, integrity, organization, and [Provenance](#) of its contents.

Within [Crucible](#), a Transfer Bundle may contain captured content from a [Dependency Store](#). A destination environment may import that content and use it to populate an internal software-package repository.

The same Transfer Bundle retains its identity through:

- Creation
- Transfer
- Receipt
- Integrity verification
- Import

The source phrases **artifact export package** and **artifact import package** describe the Transfer Bundle from the perspective of its creation or import. They do not identify separate transferable objects.

Definition

controlled collection of [Artifacts](#), [Dependencies](#), records, manifests, integrity information, and supporting [Evidence](#) assembled for transfer between environments

Source

[Crucible Concept of Operations, Version 1.0](#), Sections 2.1, 3.3, and 3.4.

The Crucible Concept of Operations states that Crucible produces a compressed Transfer Bundle containing captured Dependencies and compliance findings within a connected enclave. The disconnected enclave imports the same bundle and uses its contents to populate an internal software-package repository.

The ConOps also identifies the enclave operator as the actor that imports transferred dependency and Evidence bundles into a Disconnected Environment.

This definition formalizes the ConOps concept without treating **package** and **Transfer Bundle** as separate names for the same architectural subject.

Note

A Transfer Bundle may use:

- A compressed archive
- Removable media
- An object-store representation
- A repository export
- Another controlled representation

The representation of a Transfer Bundle does not change its identity as the controlled collection transferred between environments.

Creating a Transfer Bundle does not by itself constitute [Transfer Authorization](#).

Applicable review, approval, inspection, release, transfer, integrity-verification, and import processes govern movement across the applicable [Transfer Boundary](#).

[Transfer Bundle Integrity](#) governs the completeness of the Transfer Bundle and the protection of its contents from unauthorized addition, removal, or modification.

A Transfer Bundle may contain software packages, but a Transfer Bundle is not itself a software package.

Example

Crucible assembles operating-system update packages, installation media, Container Images, infrastructure repositories, compliance Evidence, integrity values, and a Dependency Manifest into one Transfer Bundle for controlled transfer and import into an **Air-Gapped Environment**.

© 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/t/transfer_bundle

Last update: **2026/07/19 07:30**

