

FR-DEPC-003 — Produce a Transfer Bundle

[Go to C.3.10 Dependency Capture and Offline Transfer](#)

Statement

[Crucible](#) SHALL produce a [Transfer Bundle](#) containing captured [Build Dependencies](#).

Derived From

This requirement derives from:

- Crucible System Requirements Specification, Version 1.1 Draft, Functional Requirements, FR-DEPC-003

The Original Requirement states:

The system shall produce a transferable dependency/evidence bundle.[\[C1\]](#)

FR-DEPC-003:

- Replaces **The system** with the defined system name [Crucible](#)
- Changes **shall** to the established uppercase normative form **SHALL**
- Replaces **transferable dependency/evidence bundle** with the controlled term [Transfer Bundle](#)
- Identifies captured [Build Dependencies](#) as required Transfer Bundle content
- Retains the direct behavior **produce**

No other substantive normalization is required.

Rationale

A [Transfer Bundle](#) packages content for controlled transfer between environments.

Including captured [Build Dependencies](#) in a Transfer Bundle enables those dependencies to accompany the artifacts, configurations, or other content that require them.

The Transfer Bundle provides the transferable result of [Dependency Capture](#) and [Dependency Store](#) operations.

This requirement establishes production of a Transfer Bundle without prescribing:

- A particular Transfer Bundle format
- A particular archive format
- A particular packaging tool

- A particular directory structure
- A particular compression method
- A particular encryption method
- A particular Digital Signature
- Transfer across a [Transfer Boundary](#)
- A particular [Transfer Authorization](#)
- Population of an [Offline Repository](#)
- Import into a [Disconnected Environment](#)

Separate requirements, architecture specifications, Transfer Bundle profiles, and transfer procedures govern those subjects and behaviors.

Applies To

This requirement applies to:

- [Crucible](#)
- [Transfer Bundles](#)
- Captured [Build Dependencies](#)
- [Dependency Capture](#) results
- [Dependency Stores](#)

Verification

Verification confirms that:

1. One or more captured [Build Dependencies](#) are selected for testing
2. [Crucible](#) produces a [Transfer Bundle](#)
3. The produced Transfer Bundle contains the selected Build Dependencies
4. Each selected Build Dependency can be identified within the Transfer Bundle
5. The resulting Transfer Bundle can be identified as the output of the tested production operation

Referenced By

The following pages reference this requirement:

- [5.3 Capture and Preserve Dependencies](#)
- [7.2 Produce the Transfer Bundle](#)
- [Annex D: Requirements Traceability](#)

Implementation Status

Implemented and Verified

Requirement Status

- ☐ Review and approve FR-DEPC-003 as a leaf requirement.
-

Issues

- ☐ Determine whether separate requirements define the required contents and metadata of a Transfer Bundle.
 - ☐ Determine whether separate requirements govern Transfer Bundle integrity, signing, encryption, compression, and verification.
 - ☐ Determine whether separate requirements identify which captured Build Dependencies must be included in a Transfer Bundle.
-

Notes for Editors

This requirement page retains the stable requirement identifier FR-DEPC-003.

This page is a leaf requirement page and omits a trailing `:start` from its namespace.

The Statement uses the following controlling Terms and Definitions entries:

- [Transfer Bundle](#)
- [Build Dependency](#)
- [Dependency Capture](#)
- [Dependency Store](#)

The Statement addresses production of a Transfer Bundle containing captured Build Dependencies.

FR-DEPC-001 governs capture of Build Dependencies.

FR-DEPC-002 governs preservation of captured Build Dependencies in a Dependency Store.

FR-DEPC-004 governs population of [Offline Repositories](#).

FR-DEPC-005 governs Dependency Store implementation independence.

Do not add particular Transfer Bundle formats, archive formats, packaging tools, compression methods, encryption methods, signature requirements, transfer operations, or Offline Repository obligations unless the controlling requirement changes through an approved requirements process.

To reference this requirement Statement from another wiki page, insert:

```
{{section>dido:02-crusible:99-annexes:annex-c-requirements:03-functional-
```

requirements:03-10-dependency-capture-and-offline-transfer:fr-depc-003#Statement&noheader&nofooter&noeditbtn}}

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

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

Permanent link:
<https://wiki.didosolutions.com/dido/02-crusible/99-annexes/annex-c-requirements/03-functional-requirements/03-10-dependency-capture-and-offline-transfer/fr-depc-003>

Last update: 2026/07/31 07:05

