FR-DEPC-005 — Dependency Store Implementation Independence

Go to C.3.10 Dependency Capture and Offline Transfer

Statement

Crucible SHALL permit the implementation of a Dependency Store to change without changing the Dependency Capture Contract.

Derived From

This requirement derives from:

- Crucible System Requirements Specification, Version 1.1 Draft, Functional Requirements, FR-DEPC-005

The Original Requirement states:

> *The system shall permit the dependency-store implementation to evolve without changing the dependency-capture contract.*[C1]

FR-DEPC-005:

- Replaces **The system** with the defined system name Crucible
- Changes **shall** to the established uppercase normative form **SHALL**
- Replaces the imprecise verb **evolve** with the directly testable verb **change**
- Links Dependency Store, Dependency Capture, and Contract to their controlling definitions
- Treats **Dependency Capture Contract** as a composition of the existing controlled terms rather than as a separate defined term

No other substantive normalization is required.

Rationale

The internal implementation of a Dependency Store can change as storage technologies, repository formats, scaling needs, operational constraints, or deployment environments change.

The Dependency Capture Contract defines the externally observable behavior through which Crucible captures and preserves dependencies.

Separating the Dependency Capture Contract from the Dependency Store implementation allows Crucible to replace or modify the internal storage implementation without requiring dependent

workflows, tools, or components to change how they interact with Dependency Capture.

This requirement establishes implementation independence without prescribing:

- A particular Dependency Store implementation
- A particular storage technology
- A particular repository format
- A particular database
- A particular directory structure
- A particular internal data model
- A particular interface technology
- A particular migration mechanism
- Backward compatibility beyond the Dependency Capture Contract
- Simultaneous use of multiple Dependency Store implementations

Separate requirements, architecture specifications, interface definitions, and migration procedures govern those subjects and behaviors.

Applies To

This requirement applies to:

- Crucible
- Dependency Store implementations
- Dependency Capture
- Contracts
- Components that use the Dependency Capture Contract

Verification

Verification confirms that:

1. A conforming Dependency Store implementation is selected for testing
2. A second conforming Dependency Store implementation with a different internal implementation is selected for testing
3. A Dependency Capture operation is performed using the first Dependency Store implementation
4. The first Dependency Store implementation is replaced with the second Dependency Store implementation
5. The same Dependency Capture operation is performed using the second Dependency Store implementation without changing the Dependency Capture Contract
6. Both operations produce results consistent with the Dependency Capture Contract

Referenced By

The following pages reference this requirement:

- 5.3 Capture and Preserve Dependencies
- 7.1 Capture and Preserve Dependencies
- Annex D: Requirements Traceability

Implementation Status

Implemented and Verified

Requirement Status

☐ Review and approve FR-DEPC-005 as a leaf requirement.

Issues

☐ Determine whether the architecture explicitly identifies the operations and information defined by the Dependency Capture Contract.

☐ Determine whether separate requirements define conformance criteria for alternative Dependency Store implementations.

☐ Determine whether separate requirements govern migration of preserved dependencies between Dependency Store implementations.

Notes for Editors

This requirement page retains the stable requirement identifier `FR-DEPC-005`.

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:

- Dependency Store
- Dependency Capture
- Contract

The phrase **Dependency Capture Contract** composes the controlled terms Dependency Capture and Contract. Do not create a separate glossary entry unless the architecture assigns additional semantics, required contents, or conformance rules to that combined phrase.

The Statement addresses independence of the Dependency Capture Contract from the internal Dependency Store implementation.

FR-DEPC-001 governs capture of Build Dependencies.

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

FR-DEPC-003 governs production of a Transfer Bundle.

FR-DEPC-004 governs population of Offline Repositories.

Do not add particular storage technologies, internal data models, interface technologies, migration mechanisms, or multi-store operation 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-005#Statement&noheader&nofooter&noeditbtn}}
```
