FR-CFG-001 — Declarative Infrastructure Configuration

Go to Crucible Configuration Management Requirements

Statement

Crucible SHALL represent the desired state of each Infrastructure Environment through declarative configuration that identifies the required infrastructure resources, configuration values, relationships, and applicable Baseline references without prescribing the sequence used to create the environment.

Source Statement

> The system SHALL define infrastructure through declarative configuration.

Source

Crucible System Requirements Specification, Version 1.1 Draft, Functional Requirements, Configuration Management, FR-CFG-001.

Assessment

The source statement expresses the approved functional intent but does not provide all information required for deterministic verification.

The following Specification Discipline and Authoring findings apply:

- **The system** does not use the defined system name
- **Define infrastructure** does not identify the infrastructure representation, required content, or resulting artifact
- **Infrastructure** does not identify whether the requirement applies to an Infrastructure Environment, Infrastructure Baseline, deployed resources, or another representation
- **Declarative configuration** does not identify the minimum information required to describe the desired state
- The source statement does not distinguish desired state from execution sequence
- The source statement does not identify how the declarative configuration references applicable Baselines
- The source statement does not define the validation criteria applied to the configuration
- The source statement does not identify the records required to preserve configuration Provenance and Traceability

The normalized Statement identifies the managed Infrastructure Environment, defines the minimum

content of the declarative configuration, distinguishes desired state from execution sequence, and requires applicable Baseline references.

Rationale

Declarative configuration describes the required state of an Infrastructure Environment rather than a sequence of manual or procedural actions.

A declarative representation allows Crucible to compare intended state with observed state, determine the actions required to reach the intended state, and apply the same configuration through repeatable workflows.

The declarative configuration identifies:

- Required infrastructure resources
- Required configuration values
- Relationships among resources
- Applicable Baseline identifiers and revisions
- Required inputs and parameters
- Applicable constraints
- Applicable acceptance criteria

Separating desired state from execution sequence allows provider implementations and deployment tools to determine how to realize the required environment without embedding tool-specific procedures in the platform independent configuration.

Controlled declarative configuration also supports:

- Configuration comparison
- Change review
- Configuration validation
- Configuration reuse
- Drift detection
- Reproducibility
- Provenance
- Traceability

Applies To

This requirement applies to:

- Crucible
- Declarative configuration
- Crucible Descriptions
- Infrastructure Environments
- Infrastructure Baselines
- Baselines
- Controlled inputs

- Infrastructure resources
- Configuration values
- Resource relationships
- Configuration constraints
- Acceptance criteria
- Configuration validation
- Configuration comparison
- Configuration change records
- Provenance
- Reproducibility
- Traceability
- CI/CD Pipelines
- Operational Lifecycle

Verification

1. Verification SHALL confirm that each tested declarative configuration identifies the Infrastructure Environment to which it applies
2. Verification SHALL confirm that each tested declarative configuration identifies the required infrastructure resources
3. Verification SHALL confirm that each tested declarative configuration identifies the required configuration values
4. Verification SHALL confirm that each tested declarative configuration identifies the required relationships among infrastructure resources
5. Verification SHALL confirm that each tested declarative configuration identifies the applicable Baseline identifiers and revisions
6. Verification SHALL confirm that each tested declarative configuration identifies the applicable controlled inputs and parameters
7. Verification SHALL confirm that each tested declarative configuration identifies the applicable constraints and acceptance criteria
8. Verification SHALL confirm that the declarative configuration expresses desired state without prescribing the execution sequence used to create the environment
9. Verification SHALL confirm that Crucible validates the declarative configuration before applying it
10. Verification SHALL confirm that Crucible rejects a declarative configuration that omits required information or violates an applicable constraint
11. Verification SHALL confirm that equivalent declarative configurations and controlled inputs produce Infrastructure Environments that satisfy the same acceptance criteria
12. Verification SHALL confirm that changes to a declarative configuration remain distinguishable from the previously approved configuration
13. Verification SHALL confirm that configuration and deployment records preserve the applicable Provenance, Evidence, and Traceability

Verification may include:

- Configuration schema validation
- Required-field validation
- Resource relationship validation
- Baseline reference validation
- Constraint validation

- Invalid configuration tests
- Missing input tests
- Configuration comparison
- Repeated deployment tests
- Cross-platform deployment tests
- Desired-state comparison
- Configuration drift tests
- End-to-end CI/CD Pipeline tests

The verification record SHALL identify:

1. The tested declarative configuration
2. The applicable Infrastructure Environment
3. The required infrastructure resources
4. The required configuration values
5. The resource relationships
6. The applicable Baseline identifiers and revisions
7. The controlled inputs and parameters
8. The applicable constraints
9. The applicable acceptance criteria
10. The validation results
11. The resulting Infrastructure Environment
12. The observed results
13. The generated Evidence

Outgoing Traceability

This requirement refines:

- MO-001 — Repeatable, Compliant, and Secure Infrastructure Environments
- MO-002 — Reduced Platform Deployment Timelines
- MO-003 — Platform Independent Deployment
- MO-005 — Reusable and Version-Controlled Infrastructure
- MO-006 — Infrastructure Lifecycle Management

This requirement operates within the conditions established by:

- OR-001 — Operational Roles
- OR-002 — Deployment Environments
- OR-003 — Classified and Unclassified Environments
- OR-004 — Concurrent Environment Management
- OR-005 — Distributed Environment Management

This requirement also relates to:

- 5. Descriptions, Composition, and Baselines
- 7. Infrastructure and Deployment
- 10. Reproducibility, Provenance, and Traceability

Referenced By

The wiki Backlinks function provides the current list of pages that reference `FR-CFG-001`.

Incoming traceability should be derived dynamically from backlinks rather than maintained as a duplicate manual list.

Backlinks identify incoming references but do not define the semantics of each relationship. Referencing pages should identify whether the relationship represents realization, refinement, verification, dependency, or another defined traceability relationship.

ConOps Relationship

The Crucible Concept of Operations describes Phase 1 as a command-line utility invoked through CI/CD Pipeline steps.

The Phase 1 workflow uses controlled and versioned configuration inputs to:

- Select applicable Baselines
- Compose required configuration information
- Build Machine Images
- Prepare infrastructure deployment inputs
- Perform Infrastructure Deployment
- Validate resulting environments
- Record deployment results and Evidence

Declarative configuration provides the desired-state input used by these activities while allowing provider implementations and external deployment tools to determine the execution sequence required for each target platform.

Delivery Phase

Phase 1 and subsequent phases

Implementation Status

Not Assessed

Implementation status requires verification that Crucible can validate and apply declarative configuration that identifies the required infrastructure resources, configuration values, relationships, Baseline references, constraints, and acceptance criteria.

Requirement Status

Draft

The source System Requirements Specification identifies Version 1.1 as a draft.

Notes for Editors

This requirement page should retain the stable requirement identifier `FR-CFG-001`.

Changes to the Statement SHALL preserve the approved intent of the source requirement.

Material changes should receive review and should update the related outbound traceability, verification criteria, acceptance criteria, configuration records, and source records.

The Source Statement should preserve the original wording from the controlling System Requirements Specification.

Incoming traceability should use the wiki Backlinks function rather than a manually maintained list.

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

```
{{section>dido:02-crusible:99-annexes:annex-c-requirements:03-functional-requirements:03-01-configuration-management:fr-cfg-001#Statement&noheader&nofooter&noeditbtn}}
```

Do not rename this page after an external citation unless a redirect or move plan is in place.
