

3.2 Deployment Environments

[Go to 3. Operational Environment](#)

[Crucible](#) deploys software, infrastructure resources, workloads, and platform services into supported [Deployment Targets](#).

A Deployment Target identifies the environment, platform, provider, or infrastructure context in which Crucible performs a deployment. Supported Deployment Targets are governed through the [Deployment Target Catalog](#) and the acceptance criteria defined for each target.

Deployment environments can include:

- Commercial cloud environments
- Private cloud environments
- On-premises environments
- Virtualized environments
- Container-orchestration environments
- Bare-metal infrastructure
- Hybrid environments
- Edge environments
- Connected Environments
- Disconnected Environments
- Air-Gapped Environments

A Deployment Environment can contain one or more Infrastructure Environments. Each Infrastructure Environment retains its own identity, configuration, resources, lifecycle state, operations, records, [Provenance](#), [Traceability](#), and [Evidence](#).

Deployment Targets

A supported Deployment Target has:

- An identified target type
- An approved provider or platform integration
- Defined target-specific parameters
- Applicable security and compliance criteria
- Defined deployment acceptance criteria
- A mechanism for validating the deployment result
- A record of the Provider Implementation used for deployment

The Deployment Target Catalog governs which targets Crucible recognizes as supported. Inclusion in the catalog does not imply that every Infrastructure Resource can deploy to every target. The catalog and applicable acceptance criteria identify the resources, deployment operations, provider implementations, and limitations associated with each target.

Cross-Platform Deployment

Crucible separates provider-independent deployment intent from provider-specific implementation.

A common [Crucible Description](#) and reusable Baselines can express the intended Infrastructure Environment without embedding the proprietary interfaces, services, data formats, software development kits, or operational mechanisms of one provider.

Provider Implementations translate the provider-independent intent into the operations required by the selected Deployment Target.

This separation supports:

- Deployment to more than one supported Deployment Platform
- Replacement or addition of Provider Implementations
- Portability of deployment intent
- Isolation of provider-specific dependencies
- Comparison of deployment results across platforms
- Preservation of common lifecycle controls across different environments

Deployment Forms

Depending on the selected Deployment Target and applicable requirements, Crucible can deploy:

- Infrastructure Resources
- Virtual machines
- Kubernetes clusters
- Containerized workloads
- Platform services

Each deployment uses the applicable Crucible Description, Baselines, provider-specific parameters, security criteria, compliance criteria, and acceptance criteria.

The availability of a deployment form depends on the capabilities associated with the selected Deployment Target and Provider Implementation.

Concurrent Environments

Crucible can deploy and manage more than one Infrastructure Environment concurrently.

Concurrent environments can support different purposes, including:

- Development
- Integration
- Testing
- Staging
- Production

- Training
- Demonstration
- Compliance assessment

Crucible preserves the identity, lifecycle state, configuration, resources, operations, and records associated with each environment. An operation applied to one environment must not unintentionally modify another environment.

Distributed Environments

Infrastructure Environments can reside across different locations, providers, platforms, connectivity conditions, and security domains.

Centralized management of distributed environments requires Crucible to preserve the identity and state of each managed environment while applying the provider, platform, access, and operational controls applicable to its location.

A distributed deployment does not require every environment to use the same provider or platform. Crucible preserves common operational intent while Provider Implementations address the differences among Deployment Targets.

Requirements Addressed

Requirement	Statement
MO-003a — Cross-Platform Deployment	Crucible SHALL deploy an Infrastructure Environment to two or more supported Deployment Platforms .
FR-DEP-001 — Deploy Infrastructure Resources	Crucible SHALL deploy Infrastructure Resources .
FR-DEP-002 — Deploy Virtual Machines	Crucible SHALL deploy Virtual Machines .
FR-DEP-003 — Deploy Kubernetes Clusters	Crucible SHALL deploy Kubernetes Clusters .
FR-DEP-004 — Deploy Containerized Workloads	Crucible SHALL deploy Containerized Workloads .
FR-DEP-005 — Deploy Platform Services	Crucible SHALL deploy Platform Services .
FR-MC-001 — Cloud Provider Abstraction	Crucible SHALL provide Cloud Provider Abstraction.
FR-MC-002 — Provider-Specific Extensions	Crucible SHALL provide provider-specific extensions through Provider Plugins .
FR-MC-003 — Deployment Portability Across Cloud Providers	Crucible SHALL provide deployment portability across Cloud Providers .
FR-MC-004 — Hybrid-Cloud Deployments	Crucible SHALL perform Hybrid-Cloud Deployments .
FR-MC-005 — Edge Deployments	Crucible SHALL perform Edge Deployments.

The linked leaf requirement pages remain the canonical sources.

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

Permanent link:
<https://wiki.didosolutions.com/dido/02-crusible/03-operational-context/03-02-deployment-environments?rev=1785587969>

Last update: **2026/08/01 05:39**

