

Capability Layer

[Go up to Terms and Definitions](#)

Discussion

A Capability Layer is an [Image Layer](#) that adds a defined operational, platform, service, or workload capability to a source [Machine Image](#).

The layer may add software packages, runtime services, configuration, policies, agents, orchestration components, monitoring tools, or workload-specific dependencies.

A Capability Layer separates capability-specific content from the operating-system, guest-configuration, and security-hardening layers on which it depends. This separation supports controlled change, reuse, comparison, and independent maintenance of the capability.

Applying a Capability Layer produces a [Capability Image](#).

Definition

[image layer](#) that adds a defined operational, platform, service, or workload capability to a source [machine image](#)

Source

Generalized from layered machine-image, platform-engineering, configuration-management, and workload-image usage and specialized for the Crucible architecture and operational model.

Note

A Capability Layer identifies the controlled changes used to add a capability. A Capability Image identifies the resulting Machine Image after those changes have been applied.

A Capability Layer may depend on a specific operating-system family, processor architecture, security posture, or preceding Image Layer.

Example

A Kubernetes Capability Layer adds container-runtime packages, Kubernetes services, node configuration, network components, and cluster-management tools to a Hardened Image.

© 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/c/capability_layer

Last update: **2026/07/13 11:50**

