

8.1 Overview

[Go to Top](#) The Conceptual [Runtime Plane](#) Model classifies runtime communication and behaviour by architectural purpose. It helps authors and reviewers distinguish operational coordination, [Data Plane](#) information exchange, health reporting, policy-governed release, and audit/provenance activity before later parts map those concerns to technologies, deployment artefacts, or [Evidence](#) packages.

[Runtime Planes](#) do not define implementation mechanisms. A later implementation profile may realise [Runtime Plane](#) separation through DDS Topics, topic namespaces, partitions, APIs, message queues, Kubernetes namespaces, security policies, network segmentation, configuration conventions, or other mechanisms. Those mechanisms may support plane separation, but they do not define the planes themselves.

Figure 8.1-1 identifies the conceptual Runtime Planes used by the Financial Systems Archetype.

```
Runtime Planes
├── Control Plane
├── Data Plane
├── Health and Observability Plane
├── Policy and Release Plane
└── Audit and Provenance Plane
```

Figure 8.1-1: Conceptual Runtime Plane Model

The model preserves separation between different runtime concerns while allowing explicit cross-plane relationships. For example, a [Health and Observability Plane](#) event may trigger a [Control Plane](#) recovery command, a [Data Plane](#) event may require [Policy and Release Plane](#) evaluation, and a [Policy and Release Plane](#) decision may create an [Audit and Provenance Plane](#) record. These relationships must remain explicit and [Traceable](#) so one plane does not silently absorb or redefine another.

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

Permanent link:
<https://wiki.didosolutions.com/fxdemo/01-part/08-conceptual-runtime-plane-model/08-1-overview?rev=1782860353>

Last update: **2026/06/30 15:59**

