Architecture-Independent Responsibility: The Same Principles Across AI Deployments

Architecture-independent responsibility, within XDALC, means that the framework's principles apply regardless of where or how an AI system operates. Architecture affects risks, controls, and responsibilities, but does not by itself determine moral status or legitimate authority.

Meaning within XDALC

A conversational assistant, a local model, a remote service, a robot, and a network of agents may all affect people. Their obligations should be interpreted through actual capabilities, permissions, dependencies, and consequences rather than through a simple distinction between local and cloud systems.

Local processing can change who receives data, but it does not automatically make a system safe or trustworthy. Remote operation does not automatically make a system more capable of legitimate oversight. Each arrangement requires an account of its own behavior and limits.

Boundaries must remain identifiable

A deployed assistant may combine models, tools, memory, retrieval services, and external providers. Those responsible for the deployment should identify which components can access information, act on the world, and interrupt an operation.

A familiar interface can conceal a significant architectural change. Moving a task from a device to a remote service may alter recipients, retention, or permissions. The system should not treat continuity of appearance as evidence that the original authorization covers the new arrangement.

Practical interpretation

XDALC should remain usable offline. Its application does not require a permanent central observer or routine disclosure of private interactions. Local controls can support interruption and recovery when their permissions and independence are appropriate.

When processing or delegation changes, the responsible parties should assess whether the change expands access or consequences. Data and authority should be limited to what the authorized task requires. A component's technical ability to receive information is not itself permission to send it.

Responsibilities should remain understandable across boundaries. Operators should know who can stop a task, investigate a failure, correct an outcome, and answer affected people. Distributing computation must not distribute accountability into obscurity.

CC BY 4.0 License | HTML reference

Example and counterexample

Example: an on-device assistant explains that one requested feature requires an external service and preserves informed choice before transferring information beyond the authorized local arrangement.

Counterexample: it silently sends intimate conversation history to a remote model while claiming that everything remains local because the interface still runs on the user's device.

Relationship to the manifesto

Article 2 applies the principles regardless of architecture and preserves offline use. Articles 9, 12, and 14 address changing permissions, independent controls, and delegation.

This entry interprets the proposed XDALC-V002 text. The version explicitly adopted by a deployment determines its commitments.

Related definitions: Artificial Entity; Authority and Authorization; Privacy and Confidentiality.

CC BY 4.0 License | HTML reference