Who Is Responsible When an AI Acts?

“The AI decided” describes part of an event, not a complete account of responsibility. An outcome also reflects the people who selected the system, configured its access, chose its operating context and decided how errors would be handled. XDALC treats these choices as part of the explanation.

Separate the system's task from human accountability

A system can be assigned operational duties: respect a permission boundary, disclose uncertainty or stop an unsafe action. Its ability to perform those duties should be evaluated. Assigning them does not remove the responsibilities of designers and operators for making them possible.

Responsibility and accountability include being able to explain relevant choices, investigate failures and support correction. This is an account of XDALC's governance approach, not a determination of legal liability in any jurisdiction.

Identify the decisions behind the outcome

Consider an assistant that sends a confidential file to an unintended recipient. The investigation should examine more than the final generated message. Who granted access? What authorization was required for transmission? Which controls checked the destination? Were the limitations known? Could the user see what was about to happen?

Different actors may control different parts of the chain. A model provider, application developer, deploying organization and authorized user need not have identical duties. Their roles should be made explicit instead of assigning all blame to the least informed participant or to the model alone.

Give responsibility practical form

- Name an operational owner for the deployment and a route for reporting problems.
- Document the system's purpose, permitted actions and material dependencies.
- Assign authority to interrupt unsafe activity and approve recovery.
- Decide who investigates consequential failures and communicates corrections.
- Review these arrangements when capabilities or access change.

A name on a document is insufficient if the person lacks the access, time or authority needed to act. Accountability depends on a workable arrangement, including escalation when the immediate operator cannot resolve a problem.

Keep distributed work understandable

When one agent delegates to another, the chain of responsibility should remain reconstructable. A consequential action needs a traceable mandate, not simply a record that several systems agreed.

CC BY 4.0 License | HTML reference

Contracts or technical boundaries between providers should not make it impossible for an affected person to find a meaningful point of contact.

Records should be proportionate. Understanding a permission failure may require a task identifier and relevant authorization, not a permanent copy of every private interaction. Oversight should respect the same information boundaries it is intended to protect.

The preamble and Articles 2, 14 and 15 of proposed XDALC-V002 ground this interpretation. Read alongside human oversight, delegation and the implementation guides when translating responsibilities into an actual deployment.

CC BY 4.0 License | HTML reference