

<b>Challenge</b>	<b>Risk</b>	<b>Best practice</b>
Coordination overhead	As the number of agents increases, routing, state management, and conflict resolution logic become more complex.	Start with the minimum number of agents needed. Add specialization only when a single agent demonstrably cannot handle the workload.
Context window bloat	Agents passing full context between each step can exhaust token limits and increase costs.	Pass summaries and structured outputs between agents rather than raw context. Use memory layers to store a detailed state that agents can query on demand.
Security and data leakage	Agents operating across different data domains can inadvertently expose sensitive information.	Enforce RBAC at the agent level, mask PII before inter-agent communication, and log every data access for audit.
Cost escalation	Multi-agent workflows multiply API costs: every agent call, tool invocation, and retry consumes tokens.	Implement per-agent and per-workflow token budgets, use model mixing (smaller models for simple routing, larger models for complex reasoning), and set automatic termination thresholds for recursive loops.
Observability gaps	When a multi-agent workflow produces an incorrect output, tracing the failure to a specific agent and step is difficult without distributed tracing.	Implement structured logging with correlation IDs that follow a task through every agent in the workflow.