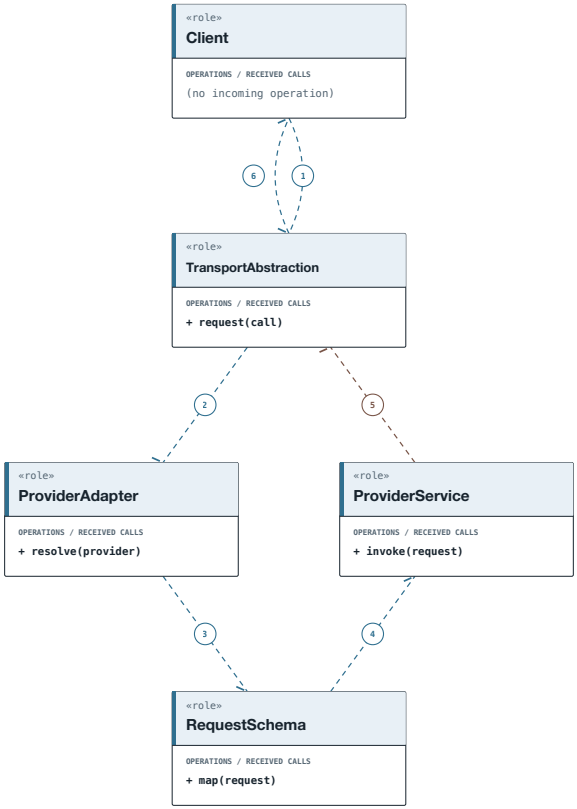
P08 Multi-provider transport abstraction

Primitive / UML CLASS VIEW / ANALYTICAL MODEL

Swap model backends.

UML CLASS VIEW

dashed arrow: call dependency · brown: return/failure · analytical, not an observed trace



POINTCUT / PATTERN INTERFACE

CONTEXT	A harness can select among model backends.
TRIGGER	The runtime resolves or invokes a model provider.
STRUCTURE	A shared request/response/error boundary plus provider-specific implementations.
EFFECT	Backends are interchangeable through a declared interface, with differences exposed.

CONTRACT / REQUIRED BEHAVIOR

REQUIRES	Provider identity, model configuration, and common result types are available.
ENSURES	Every provider satisfies the same invocation contract or reports unsupported behavior.
INVARIANT	Provider-specific state remains behind the adapter boundary.
VIOLATION	Authentication, rate, schema, and capability errors map to explicit types.

RELATIONSHIPS / ANALYTICAL CALLS

number · caller → callee · message

1 Client → TransportAbstraction request(call)
2 TransportAbstraction → ProviderAdapter resolve(provider)
3 ProviderAdapter → RequestSchema map(request)
4 RequestSchema → ProviderService invoke(request)
5 ProviderService ↩ TransportAbstraction return(response)
6 TransportAbstraction → Client return(result_or_error)

VARIATION / TEST / NEGATIVE CONTROL

VARIATION	Registry, factory, adapter, transport, retry advice, and capability discovery.
TEST_OBLIGATION	Test two providers, unsupported tools, usage mapping, and error translation.
NEGATIVE_CONTROL	A single hard-coded vendor client is not the abstraction.