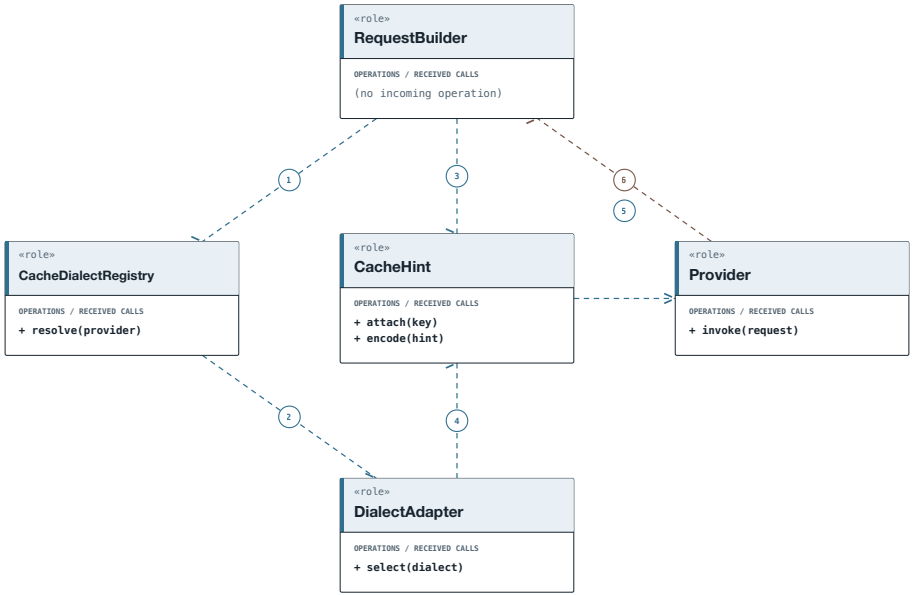
# P07 Cache-dialect fanout

**Primitive / UML CLASS VIEW / ANALYTICAL MODEL**

Adapt caching to provider-specific interfaces.

**UML CLASS VIEW**

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




**POINTCUT / PATTERN INTERFACE**

| | |
|---|---|
| CONTEXT | A request is built for a provider with cacheable context. |
| TRIGGER | A cache key, hint, or control record is attached. |
| STRUCTURE | Provider dialect adapter, cache identity, and response association. |
| EFFECT | Compatible cache controls are encoded and unsupported controls are rejected or omitted. |

**CONTRACT / REQUIRED BEHAVIOR**

| | |
|---|---|
| REQUIRES | Provider capability and cache identity are known before dispatch. |
| ENSURES | Cache control maps to the request without changing semantic content. |
| INVARIANT | Cache identity never crosses user, tenant, or incompatible provider boundaries. |
| VIOLATION | Unsupported or invalid hints degrade explicitly without silent semantic drift. |

**RELATIONSHIPS / ANALYTICAL CALLS**

number · caller → callee · message

1 RequestBuilder → CacheDialectRegistry resolve(provider)
2 CacheDialectRegistry → DialectAdapter select(dialect)
3 RequestBuilder → CacheHint attach(key)
4 DialectAdapter → CacheHint encode(hint)
5 CacheHint → Provider invoke(request)
6 Provider ↩ RequestBuilder return(response)

**VARIATION / TEST / NEGATIVE CONTROL**

| | |
|---|---|
| VARIATION | Prompt cache keys, cache-control blocks, TTLs, and dialect adapters. |
| TEST_OBLIGATION | Test identity collisions, invalidation, unsupported providers, and privacy boundaries. |
| NEGATIVE_CONTROL | A generic response cache is not provider-dialect cache control. |