

# ADR-019: Integration Adapter Registry

```
# ADR-019 – Integration Adapter Registry

**Status:** Accepted
**Date:** 2026-05-11
**Author:** Petter Graff (CodeCraft – Architecture Lead)
**Decision-maker:** CEO Alem Baši?
**Mehanik clearance:** /tmp/mehanik-cleared-100362
**MC Task:** #100362 (Phase 0' ADR Consolidation)
**Cross-references:**

- ADR-015 (CountryPlugin – plugin selects adapters for its market; plugin version compatibility)
- ADR-016 (EInvoiceAdapter – one of the 7 adapter categories; lifecycle states formalised here)
- ADR-023 (routing – market resolved at edge before adapter dispatch)
- `apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeEInvoiceAdapter.kt` (reference impl)
- `apps/api/src/main/kotlin/no/alai/bilko/einvoice/EInvoiceTypes.kt` (AdapterLifecycleState on disk)
- Plan v3 §6 Phase 0' Task 0'4 – `~/system/specs/bilko-multi-market-architecture-plan-v3-2026-05-11.md`

---

## 1. Context

### 1.1 Problem: Seven Integration Surfaces, No Governance

Bilko integrates with external systems across seven functional domains. As of 2026-05-11, only one adapter exists (`StorecoveHrFiskeEInvoiceAdapter`). Without a registry and governance model, adding the second adapter (SEF for RS) and every subsequent adapter will produce:

1. Inconsistent error handling – platform-native exceptions leaking across boundaries
2. No feature-flag mechanism – a broken SEF adapter takes down all RS users
3. Secret sprawl – `STORECOVE_API_KEY` as an env var pattern, but no taxonomy when there are 7 adapters × 4 markets × 3 environments = up to 84 secrets
4. No observability standard – each adapter invents its own logging and metrics
5. No lifecycle discipline – adapters deployed to production without sandbox verification

### 1.2 Reference Implementation Patterns

`StorecoveHrFiskeEInvoiceAdapter.kt` already demonstrates all the patterns this ADR formalises. This ADR makes those patterns enforceable for all future adapters:

| Pattern | |
| :--- | :--- |
| StorecoveHrFiskeEInvoiceAdapter | | ADR-019 makes it |
| ----- | ----- |
| PII redaction before logging | | Lines 24-59 |
| (`sanitizeForLog`) | Mandatory |
| `AdapterException` only (no platform exceptions) | | Lines 469-516 |
| (`StorecoveErrorMapper`) | Mandatory |
| Per-adapter Prometheus metrics | | Lines 537-540 |
| (`StorecoveMetrics`) | Mandatory |
| Lifecycle state field | | Lines 547-548 (`lifecycleState =` |
| STUB`) | Mandatory |
| Idempotency key on submit | | Lines 591-600 (D5 |
| comment) | Mandatory |
| Credentials NOT required for serialize() | | Lines |
| 567-571 | Mandatory |
| Startup credential validation flag | | Lines |
| 83-138 | Recommended |
```

---

## ## 2. Decision

### ### 2.1 Seven Adapter Categories

Every external integration belongs to exactly one of the following categories. Each category is a Kotlin interface in `apps/api/src/main/kotlin/no/alai/bilko/adapter/`.

| Category | Interface                | Purpose                                                                    | Markets                                             |
|----------|--------------------------|----------------------------------------------------------------------------|-----------------------------------------------------|
| 1        | `EInvoiceAdapter`        | E-invoice serialization + fiscal platform submission (UINO)                | HR (Storecove), RS (SEF), BA-FED (CPF), BA-RS       |
| 2        | `CompanyRegistryAdapter` | Company data lookup (name, address, tax status) from government registries | HR (FINA), RS (APR), BA (stub)                      |
| 3        | `BankStatementAdapter`   | Bank statement import (MT940, CAMT.053, PSD2 AISP) platform                | All markets – via Tok Open Banking                  |
| 4        | `ExchangeRateAdapter`    | FX rate feed (daily/live)                                                  | All markets (ECB primary, HNB for HR, NBS for RS)   |
| 5        | `TaxFilingAdapter`       | Electronic VAT/CIT return submission to tax authority (TBD)                | HR (ePorezna), RS (ePorezi), BA                     |
| 6        | `FiscalDeviceAdapter`    | Fiscal receipt device or cloud fiscal service (BA (TBD)                    | HR (Fiskalizacija cloud cert), RS (LPFR chip card), |
| 7        | `QESSigningAdapter`      | Qualified Electronic Signature for invoice signing (stub)                  | HR (FINA QES), RS (stub), BA                        |

**\*\*Current implementation status:\*\***

- `EInvoiceAdapter`: `StorecoveHrFiskeInvoiceAdapter` (HR, STUB lifecycle)
- All other categories: NOT YET IMPLEMENTED

### ### 2.2 Common Interface Contract

Every adapter interface extends a common `BilkoAdapter` base:

```
```kotlin
package no.alai.bilko.adapter

import no.alai.bilko.country.TaxJurisdiction
import no.alai.bilko.einvoice.AdapterLifecycleState

/**
 * Base contract for all Bilko integration adapters.
 *
 * Every adapter implementation MUST:
 * 1. Expose [jurisdiction] and [lifecycleState] as first-class properties.
 * 2. Throw only [AdapterException] – NEVER platform-native exceptions.
 * 3. Pass all log writes through [sanitizeForLog] (defined per-adapter for PII fields).
 * 4. Record Prometheus metrics on every external call (see §2.6).
 * 5. Not require credentials for read-only / serialization operations.
 */
interface BilkoAdapter {
    val jurisdiction: TaxJurisdiction
    val lifecycleState: AdapterLifecycleState
    val adapterVersion: String // Semantic version string, e.g. "1.0.0"
}
```

### ### 2.3 AdapterException – Canonical Error Contract

All adapters throw `AdapterException` and nothing else. This exception type is the single crossing point from adapter space to core service space.

```
```kotlin
package no.alai.bilko.adapter

import no.alai.bilko.country.TaxJurisdiction
```

```

/**
 * Canonical adapter error. The ONLY exception type that crosses the adapter boundary.
 *
 * INVARIANT: Core services catch AdapterException only. They MUST NOT catch
 * platform-native exceptions (Ktor ResponseException, HttpRequestTimeoutException,
 * java.net.SocketTimeoutException, etc.). Map those to AdapterException in the adapter.
 *
 * [retryable]: if true, caller may retry with exponential backoff.
 * [rawPayload]: sanitized (PII-redacted) raw response body for audit. NEVER raw.
 */
data class AdapterException(
    val code: AdapterErrorCode,
    val market: TaxJurisdiction,
    val retryable: Boolean,
    val rawPayload: String,
    override val message: String = code.name,
    override val cause: Throwable? = null,
) : RuntimeException(message, cause)

/**
 * Canonical error codes – adapter-independent.
 *
 * Adapters map platform-specific HTTP status codes and error bodies to these codes.
 * See StorecoveErrorMapper (lines 469–516) for the HR reference mapping.
 */
enum class AdapterErrorCode {
    // Validation errors – not retryable
    VALIDATION_SCHEMA_ERROR, // Invalid document structure (HTTP 400/422)
    VALIDATION_BUSINESS_RULE, // Business rule violation (e.g., invalid OIB, non-EUR
currency)
    VALIDATION_DUPLICATE_DOCUMENT, // Idempotency conflict (HTTP 409)

    // Authentication/authorisation – not retryable
    AUTH_INVALID_CREDENTIALS, // API key invalid / token expired / certificate
rejected

    // Platform errors – retryable
    PLATFORM_RATE_LIMITED, // HTTP 429 – back off and retry
    PLATFORM_MAINTENANCE, // HTTP 503 – platform in scheduled maintenance
    PLATFORM_INTERNAL_ERROR, // HTTP 5xx – transient platform error

    // Network errors – retryable
    NETWORK_TIMEOUT, // Connection or read timeout
    NETWORK_UNREACHABLE, // DNS resolution failure or TCP refused

    // Implementation status – not retryable
    NOT_IMPLEMENTED, // Adapter is in STUB lifecycle state
    UNKNOWN, // Unmapped error; always log rawPayload for triage
}

/**Mapping rule for new adapters:** Every HTTP status code the platform can return MUST
have a mapping to an `AdapterErrorCode`. Use `UNKNOWN` only as a catch-all, never as
the primary mapping for a known status code. See `StorecoveErrorMapper` (lines 469–516
in `StorecoveHrFiskeInvoiceAdapter.kt`) as the reference pattern.

### 2.4 AdapterConfig – DB-Level Feature Flag

Every adapter is gated by an `AdapterConfig` row. An adapter MUST NOT execute any
network call if its `AdapterConfig.enabled = false`. This allows disabling a broken
adapter without redeployment.

```sql
-- V20__adapter_config.sql (Phase 1H – during Phase 2A window)

CREATE TABLE adapter_config (
    id          UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    market      VARCHAR(8) NOT NULL,
               -- TaxJurisdiction enum value: 'HR', 'RS', 'BA_FED', 'BA_RS'
    adapter_type VARCHAR(32) NOT NULL,
               -- Matches the 7 categories: 'EINVOICE', 'COMPANY_REGISTRY',
               -- 'BANK_STATEMENT', 'EXCHANGE_RATE', 'TAX_FILING',
               -- 'FISCAL_DEVICE', 'QES_SIGNING'
    enabled     BOOLEAN NOT NULL DEFAULT FALSE,
    reason      TEXT,
               -- Why disabled, e.g. "MC #8675 pending – Storecove account not activated"

```

```

        -- Required when enabled=false.
updated_at    TIMESTAMPTZ NOT NULL DEFAULT now(),
updated_by    TEXT NOT NULL DEFAULT 'system', -- MC task ID or admin user
CONSTRAINT pk_adapter_config UNIQUE (market, adapter_type)
);

-- Seed: all adapters start disabled
INSERT INTO adapter_config (market, adapter_type, enabled, reason, updated_by)
VALUES
    ('HR', 'EINVOICE', false, 'MC #8675 – Storecove account pending', 'MC-100362'),
    ('HR', 'COMPANY_REGISTRY', false, 'Not implemented – Phase 1S', 'MC-100362'),
    ('HR', 'BANK_STATEMENT', false, 'Tok AISP integration pending', 'MC-100362'),
    ('HR', 'EXCHANGE_RATE', false, 'ECB feed not configured', 'MC-100362'),
    ('HR', 'TAX_FILING', false, 'ePorezna integration Phase 2 scope', 'MC-100362'),
    ('HR', 'FISCAL_DEVICE', false, 'Fiskalizacija cert not configured', 'MC-100362'),
    ('HR', 'QES_SIGNING', false, 'FINA QES Phase 2 scope', 'MC-100362'),
    ('RS', 'EINVOICE', false, 'SEF adapter Phase 1S scope', 'MC-100362'),
    ('BA_FED', 'EINVOICE', false, 'CPF platform TBD ~2027', 'MC-100362'),
    ('BA_RS', 'EINVOICE', false, 'UINO platform TBD', 'MC-100362');
-- (Remaining BA/RS adapter rows follow same pattern)

-- Admin can enable without redeploy:
-- UPDATE adapter_config SET enabled = true, reason = NULL, updated_by = 'MC-8675-DONE'
-- WHERE market = 'HR' AND adapter_type = 'EINVOICE';

```

**\*\*Kotlin enforcement pattern:\*\***

```

```kotlin
// In the adapter registry
(apps/api/src/main/kotlin/no/alai/bilko/adapter/AdapterRegistry.kt)
fun requireEnabled(market: TaxJurisdiction, adapterType: String) {
    val config = adapterConfigRepository.find(market, adapterType)
    ?: throw AdapterException(
        code = AdapterErrorCode.NOT_IMPLEMENTED,
        market = market,
        retryable = false,
        rawPayload = "",
        message = "No AdapterConfig row for ($market, $adapterType) – run Flyway V20"
    )
    if (!config.enabled) {
        throw AdapterException(
            code = AdapterErrorCode.NOT_IMPLEMENTED,
            market = market,
            retryable = false,
            rawPayload = "",
            message = "Adapter ($market, $adapterType) is disabled: ${config.reason}"
        )
    }
}
```

```

### ### 2.5 Lifecycle States and Transition Criteria

Formalised from `AdapterLifecycleState` enum in `EInvoiceTypes.kt` lines 22–26, and from ADR-016 §2.3. Applies to ALL adapter categories.

```

...
STUB ???????????????????? SANDBOX_VERIFIED ???????????????????? PRODUCTION
...

```

**\*\*STUB\*\*** (initial state for all adapters):

- Compiles and registers successfully
- All network methods throw `AdapterException(code=NOT\_IMPLEMENTED)`
- `serialize()` / read-only operations may work (e.g., HR serialize works in STUB)
- `AdapterConfig.enabled` is `false`

**\*\*SANDBOX\_VERIFIED\*\*** transition criteria (all must be true):

- Minimum 5 distinct happy-path test cases pass against the real sandbox platform (not mocked – real submission IDs, real response payloads)
- All test case submission IDs are archived in BookStack as evidence
- Error mapping verified: at least HTTP 400, 401, 409, 429, 503, 5xx all produce correct `AdapterErrorCode` values (not `UNKNOWN`)
- Proveo sign-off with evidence file path in MC task
- `AdapterConfig.enabled` can be set to `true` after this point

**\*\*PRODUCTION\*\*** transition criteria (all must be true):

- `SANDBOX\_VERIFIED` already achieved
- Securiion review of adapter error handling, PII sanitization, and idempotency key implementation – no critical findings
- 30 consecutive days on stage Cloud Run with:
  - Zero `PLATFORM\_INTERNAL\_ERROR` alerts
  - Zero cross-market routing errors
  - `bilko\_integration\_request\_total{status="error"}` < 1% of total requests
- CEO approval for production activation
- `AdapterConfig.enabled = true` in production DB (separate row from stage DB)

### ### 2.6 Secret Taxonomy

Runtime secrets follow the pattern `Bilko/{env}/{market}/{secret-name}`.

**\*\*Env first, not market first.\*\*** This ensures that all production secrets are under `Bilko/production/` and can be granted/revoked as a unit for environment promotion.

...

```
Bilko/
  production/
    HR/
      STORECOVE_API_KEY
      STORECOVE_LEGAL_ENTITY_ID
      FINA_QES_CERTIFICATE          (Phase 2)
      EPOREZNA_CLIENT_SECRET       (Phase 2)
    RS/
      SEF_API_KEY                  (Phase 1S)
      LPFR_DEVICE_CERT             (Phase 2)
    BA_FED/
      CPF_API_KEY                  (Phase 1B – pending platform launch)
    BA_RS/
      UINO_API_KEY                 (Phase 1B – pending platform launch)
  stage/
    HR/
      STORECOVE_API_KEY
      STORECOVE_LEGAL_ENTITY_ID
    RS/
      SEF_API_KEY
  ...
  local/
    HR/
      STORECOVE_API_KEY           (developer sandbox credentials only)
  ...
```

...

**\*\*Secret resolution hierarchy:\*\***

1. Runtime: GCP Secret Manager (current) – accessed via `SecretResolver` interface
2. Break-glass: Vaultwarden (`vault.basicconsulting.no`) – human access only, NOT runtime source
3. Local dev: `.env.local` file (`.gitignore`) – NEVER committed

```kotlin

```
// apps/api/src/main/kotlin/no/alai/bilko/adapter/SecretResolver.kt
```

```
/**
```

```
 * Abstracts secret retrieval behind a testable interface.
 *
 * Production implementation: GCP Secret Manager.
 * Test implementation: environment variables / in-memory map.
 *
 * Secret path convention: Bilko/{env}/{market}/{secret-name}
 */
```

```
interface SecretResolver {
```

```
  /**
```

```
   * Resolves a secret value by its canonical path.
```

```
   *
```

```
   * @param path e.g. "Bilko/production/HR/STORECOVE_API_KEY"
```

```
   * @return Secret value, or null if not found.
```

```
   * @throws AdapterException(AUTH_INVALID_CREDENTIALS) if path exists but value is empty/blank.
```

```
  */
```

```
  fun resolve(path: String): String?
```

```

/**
 * Convenience method: builds canonical path and resolves.
 * @param env "production" | "stage" | "local"
 * @param market TaxJurisdiction enum value as string
 * @param secretName The specific secret name
 */
fun resolve(env: String, market: String, secretName: String): String? =
    resolve("Bilko/$env/$market/$secretName")
},

```

**\*\*Vaultwarden is NOT the runtime secret source.\*\*** Vaultwarden is the human break-glass vault for emergency access. Do not write Kotlin code that reads from Vaultwarden at runtime. GCP Secret Manager is the runtime source.

### ### 2.7 Observability Mandate

Every adapter MUST emit the following for every network call:

**\*\*Structured log line (one per call):\*\***

```

...
level=INFO market=HR integration=EINVOICE env=production org_id=<uuid>
action=submit status=SUCCESS duration_ms=234 submission_id=<guid>
...

```

Required fields: `market`, `integration`, `env`, `org\_id`. Optional but recommended: `duration\_ms`, `submission\_id`, `attempt` (for retries).

**\*\*NEVER log:\*\***

- OIB, PIB, JIB (tax IDs)
- IBAN
- `document\_data` (invoice XML body)
- `api\_key`, `api\_secret`

Use `sanitizeForLog()` (pattern from `StorecoveHrFiskeEInvoiceAdapter.kt` lines 24-59) before any log write that touches a response body.

**\*\*Prometheus metrics (one counter per adapter):\*\***

```

```kotlin
// apps/api/src/main/kotlin/no/alai/bilko/adapter/AdapterMetrics.kt

/**
 * Prometheus counter for all adapter network calls.
 * Labels: market, integration, status (SUCCESS | ERROR | NOT_IMPLEMENTED | TIMEOUT)
 * Example PromQL for HR e-invoice error rate:
 *
 * rate(bilko_integration_request_total{market="HR",integration="EINVOICE",status="ERROR"}[5m])
 * /
 * rate(bilko_integration_request_total{market="HR",integration="EINVOICE"}[5m])
 */
// bilko_integration_request_total{market, integration, status}
// bilko_integration_request_duration_seconds{market, integration, status}
```

```

Per-(market, integration) alert rule:

- Error rate > 10% over 5 minutes: PAGE (PagerDuty or Slack alert)
- Error rate > 25% over 1 minute: CRITICAL (adapter auto-disabled via `AdapterConfig`)

### ### 2.8 Adapter Versioning

Each adapter declares `val adapterVersion: String` (semantic version, e.g., `"1.0.0"`). The corresponding `CountryPlugin` implementation declares the minimum adapter version it requires:

```

```kotlin
// In PluginHR:
companion object {
    const val MIN_EINVOICE_ADAPTER_VERSION = "1.0.0"
}

```

```
// Startup check in DI.kt:
val adapter = StorecoveHrFiskeEInvoiceAdapter()
require(semVer(adapter.adapterVersion) >= semVer(PluginHR.MIN_EINVOICE_ADAPTER_VERSION)) {
    "PluginHR requires EInvoiceAdapter >= ${PluginHR.MIN_EINVOICE_ADAPTER_VERSION}, got
    ${adapter.adapterVersion}"
}
}
```

Adapters are versioned independently of the `CountryPlugin`. Breaking changes to an adapter interface (e.g., new required parameter in `submit()`) require a major version bump and a coordinated plugin + adapter update.

### ### 2.9 Idempotency Requirements

**\*\*All submit-type methods in all adapters MUST include an idempotency key.\*\***

The idempotency key format is adapter-specific, but the value MUST be derived deterministically from the invoice or entity content – never a random UUID.

| Adapter derivation                        | Method     | Idempotency key                                              |
|-------------------------------------------|------------|--------------------------------------------------------------|
| -----                                     | -----      | -----                                                        |
| EInvoiceAdapter (HR) matches Storecove D5 | `submit()` | `SHA-256(invoice.id + invoice.invoiceNumber)` –              |
| EInvoiceAdapter (RS)                      | `submit()` | `SHA-256(invoice.id + invoice.invoiceNumber)` (same pattern) |
| TaxFilingAdapter (org.taxId)              | `submit()` | `SHA-256(filing.periodStart + filing.periodEnd +             |
| QESSigningAdapter (signer.taxId)          | `sign()`   | `SHA-256(document.contentHash +                              |

**\*\*Rationale:\*\*** A network timeout after the platform receives the request but before the response arrives will cause the client to retry. Without idempotency, this creates a duplicate document. Storecove returns HTTP 409 on duplicate `document\_id` (D2 in `StorecovePayloadBuilder.wrap()`, lines 420-436) – the pattern must be replicated.

---

### ## 3. Implementation Path

| Phase       | Task                                                                                                                  | Status              |
|-------------|-----------------------------------------------------------------------------------------------------------------------|---------------------|
| Deliverable | -----                                                                                                                 | -----               |
| Phase 0'    | This ADR written to disk<br>INTEGRATION-ADAPTER-REGISTRY.md                                                           | ADR-019-<br>DONE    |
| Phase 1H    | `AdapterException` + `AdapterErrorCode` formalized<br>`adapter/AdapterException.kt` (already exists – verify package) | Verify existing     |
| Phase 1H    | `AdapterConfig` Flyway migration (V20)<br>`V20__adapter_config.sql`                                                   | BLOCKED BY Phase 2A |
| Phase 1H    | `SecretResolver` interface + GCP impl<br>`adapter/SecretResolver.kt` + `GcpSecretResolver.kt`                         | Phase 1H.4+         |
| Phase 1H    | `AdapterRegistry` + `requireEnabled()` check<br>`adapter/AdapterRegistry.kt`                                          | Phase 1H.4+         |
| Phase 1H    | Prometheus metrics wired in `StorecoveHrFiskeEInvoiceAdapter`<br>`StorecoveMetrics.kt` (skeleton exists)              | Phase 1H.2+         |
| Phase 1S    | SEF RS EInvoiceAdapter<br>`country/rs/SefRsEInvoiceAdapter.kt`                                                        | Post-HR GA          |
| Phase 1B    | CPF BA-FED + UINO BA-RS stubs<br>`country/ba/Cpf*`, `country/ba/Uino*`                                                | Post-RS GA          |

---

### ## 4. Consequences

#### ### 4.1 Positive

- **\*\*Zero platform exception leakage.\*\*** `AdapterException` as the only crossing type means core services have one error handler for all  $7 \times 4 = 28$  potential adapter instances.
- **\*\*Hot disable without redeploy.\*\*** `AdapterConfig.enabled = false` disables a broken adapter in < 1 minute (DB write). No restart required. Incident response time drops from minutes (redeploy) to seconds (DB update).
- **\*\*Observability from day one.\*\*** Every adapter emits standardized metrics. Error rate alerts fire before users report issues.
- **\*\*Secret hygiene.\*\*** Env-first taxonomy (`Bilko/{env}/{market}/{secret}`) makes environment promotion (stage ? production) a structured operation, not an ad-hoc

copy. Break-glass access is separated from runtime access.

### ### 4.2 Negative

- **\*\*Adapter scaffolding cost.\*\*** Each new adapter requires implementing the full interface contract, ``AdapterConfig`` rows, ``SecretResolver`` wiring, and Prometheus metrics. Estimate: 1-2 days for a new adapter in STUB state.
- **\*\*AdapterConfig is a deployment dependency.\*\*** The application fails at startup if ``adapter_config`` rows do not exist. Flyway V20 must run before the application version that adds ``requireEnabled()`` checks.

### ### 4.3 Risks

- **\*\*`AdapterConfig` DB unavailable.\*\*** If the database is unreachable, ``requireEnabled()`` fails, blocking all adapters. Mitigation: cache ``AdapterConfig`` in-memory at startup with a TTL of 5 minutes. Use cached state if DB is unreachable.
- **\*\*Metrics cardinality.\*\*** High ``org_id`` cardinality in metrics labels would cause Prometheus memory issues. The observability mandate specifies ``org_id`` in log lines, NOT in Prometheus labels. Labels are ``market``, ``integration``, ``status`` only – bounded cardinality.

---

## ## 5. References

| Reference                                                                                          |                                                              |
|----------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| Path                                                                                               |                                                              |
| Lines                                                                                              |                                                              |
| -----                                                                                              |                                                              |
| -----                                                                                              |                                                              |
| <code>`AdapterLifecycleState`</code> enum (on disk)                                                |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/einvoice/EInvoiceTypes.kt`</code>                    |                                                              |
| 22-26                                                                                              |                                                              |
| <code>`StorecoveErrorMapper`</code> (error mapping reference)                                      |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeInvoiceAdapter.kt`</code> |                                                              |
| 469-516                                                                                            |                                                              |
| <code>`StorecoveMetrics`</code> (metrics reference)                                                |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeInvoiceAdapter.kt`</code> |                                                              |
| 537-540                                                                                            |                                                              |
| PII sanitization reference                                                                         |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeInvoiceAdapter.kt`</code> |                                                              |
| 24-59                                                                                              |                                                              |
| Idempotency key (D5) reference                                                                     |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeInvoiceAdapter.kt`</code> |                                                              |
| 94-98                                                                                              |                                                              |
| <code>`StorecovePayloadBuilder`</code> (D2 document_id)                                            |                                                              |
| <code>`apps/api/src/main/kotlin/no/alai/bilko/country/hr/StorecoveHrFiskeInvoiceAdapter.kt`</code> |                                                              |
| 420-436                                                                                            |                                                              |
| ADR-016 §2.3 (lifecycle states – EInvoice)                                                         | <code>`docs/architecture/ADR-016-EINVOICE-ADAPTER.md`</code> |
| ADR-015 §2.4 (DI registration pattern)                                                             | §2.3                                                         |
| JURISDICTION-PLUGIN.md                                                                             | <code>`docs/architecture/ADR-015-FOUR-`</code>               |
| Plan v3 §6 Task 0'4 acceptance criteria                                                            | §2.4                                                         |
| <code>`architecture-plan-v3-2026-05-11.md`</code>                                                  | <code>`~/system/specs/bilko-multi-market-`</code>            |
|                                                                                                    | 279-290                                                      |

---

## ## 6. Approval

**\*\*Status:\*\*** Accepted  
**\*\*Unblocks:\*\***

- Phase 1H: ``AdapterConfig`` Flyway V20 migration
- Phase 1H: ``SecretResolver`` GCP implementation
- Phase 1H: Prometheus metrics wiring in ``StorecoveHrFiskeInvoiceAdapter``
- Phase 1S: SEF RS adapter scaffolding (knows the contract to implement against)
- Phase 1B: CPF/UINO BA adapter stubs

| Role                             | Sign                                  | Date       |
|----------------------------------|---------------------------------------|------------|
| -----                            | -----                                 | -----      |
| Architecture Lead (Petter Graff) | Signed                                | 2026-05-11 |
| CEO (Alem Baši?)                 | Not required for registry pattern ADR | –          |

---

## ## 7. Document History

| Date       | Author       | Change                                            |
|------------|--------------|---------------------------------------------------|
| 2026-05-11 | Petter Graff | Initial – Phase 0' ADR consolidation (MC #100362) |

Revision #4

Created 2026-05-14 10:09:19 UTC by John

Updated 2026-07-19 20:01:36 UTC by John