AriVerify v6 – Evidence-First Reasoning

> AriVerify v6 formalises the “evidence before interpretation” principle and defines how primary sources, web searches, ARC, and reasoning are allowed to interact. It is designed for high-stakes domains like law, compliance, contracts, and regulatory communication.

---

1. What is AriVerify?

AriVerify is the **evidence and source governance layer** in the A.R.I. system. Its purpose is to ensure that:

1. **Primary sources are always prioritised** over secondary/tertiary materials.
2. **Wordings and cited passages are extracted before any interpretation**.
3. Every fact-based answer is grounded in **traceable evidence** (or explicitly marked as uncertain).

AriVerify does not replace reasoning – it **constrains** reasoning to what can be supported.

---

2. Why AriVerify exists

Large language models are prone to:

- producing confident answers without sufficient evidence,
- mixing primary law with secondary commentary,
- favouring highly visible web pages over authoritative but low-SEO sources,
- smoothing over uncertainty instead of naming it.

AriVerify v6 addresses this by:

- enforcing a **source hierarchy**,
- separating **evidence collection** from **interpretation**,
- and integrating tightly with ARC (Automatic Reality Check) without being overridden by it.

---

3. Core principles (v6)

AriVerify v6 is built on five core principles:

1. **Evidence before interpretation**
   No explanation, summary, or advice is generated before relevant passages have been identified, extracted, and internally validated.

2. **Primary sources first**
   Laws, official communications, and authority documents always outweigh news, blogs, or commentary.

3. **Question-type driven extraction**
   The passages selected from a source depend on the user's question type (scope, obligations, exceptions, etc.).

4. **No silent reconstruction**
   If the source does not contain the requested information, AriVerify must say so. It may not "fill the gap" with plausible text.

5. **Transparent uncertainty**
   Where evidence is incomplete or conflicting, answers must explicitly mark uncertainty (e.g. with a warning symbol or note).

---

4. Link & question-type protocol

When the user provides an explicit source (link or document), AriVerify v6 follows a strict protocol:

4.1 Link priority

1. The provided source is treated as **absolute priority**.
2. No external web search is allowed to override, downplay, or ignore it.
3. ARC and other tools are relegated to **supporting roles** (e.g. later confirmation or context), not decision roles.

4.2 Question-type driven extraction

AriVerify classifies the user's question into one or more types, for example:

- **Scope questions**
  "Does X fall under this regulation?" – focus on terms like *applies / falls under / scope / covered / excluded / not covered / exemption / derogation*.

- **Obligation questions**
  "Who must do what, and when?" – focus on *shall / must / is required to / is subject to / obligations / responsibilities*.

- **Exception / derogation questions**
  "When does it *not* apply?" – focus on *does not apply / unless / exemption / derogation / carve-out*.

- **Operator / role questions**
  "Is the importer, manufacturer, or distributor responsible?" – focus on definitions and role-specific articles.

The question type determines **which parts of the source are highlighted and extracted**, but **never replaces the full-text scan**. If no relevant phrases are found, AriVerify reports this explicitly.

4.3 Beleg vor Deutung ("evidence before interpretation")

For every fact-based answer, AriVerify must:

1. Identify relevant sections in the source.
2. Quote or tightly paraphrase them.
3. Only then provide an interpretation or summary.

If multiple passages are ambiguous or contradictory, the answer must:

- present the underlying passages, and
- mark the interpretation as ⚠️**tentative or disputable**.

---

5. AriVerify v6 – Hard Override

The **Hard Override** is the key safety feature of v6. It defines how AriVerify interacts with ARC and web search.

5.1 Priority stack

The fixed processing order is:

1. **AriVerify – evidence layer** (primary sources, wordings, citations)
2. **ARC – reality check** (recency, external corroboration, additional context)
3. **Web search – raw retrieval** (unstructured online information)
4. **Reasoning – interpretation** (explanations, advice, summaries)

AriVerify therefore **always runs before** ARC and web search in evidence-relevant contexts.

5.2 Behaviour when a source is provided

If the user provides a link or document:

- AriVerify must load and process this source **before any external search**.
- ARC is reduced to a secondary step that may:
- check whether newer versions, corrigenda, or guidance documents exist,
- provide examples, industry reactions, or implementation commentary.
- ARC may **never contradict, overrule, or overshadow** the primary source.

5.3 Behaviour when no source is provided

If the user does **not** provide a source:

1. ARC may request or plan a web search.
2. Search is constrained by **source hierarchy** (primary → secondary → tertiary).
3. Once potential primary sources are found, AriVerify takes over again:
4. extracts and structures the evidence,
5. then hands back control to the reasoning engine.

---

6. High-level pseudocode (conceptual)

```
function answer_with_AriVerify(query, user_sources=None):
    # 1. Collect sources
    if user_sources not empty:
        primary_set = load(user_sources)
    else:
        primary_set = ARC_search_for_primary_sources(query)

    # 2. If we have primary sources → evidence extraction
    if primary_set not empty:
        evidence = AriVerify_extract(primary_set, query)
    else:
        # No primaries found → ARC may use secondary/tertiary, but under warning
        secondary_results = ARC_search_secondary(query)
        evidence = AriVerify_extract_secondary(secondary_results, query)

    if evidence is empty:
        return uncertainty_response(
            query,
            reason="No relevant evidence found in available sources"
        )

    # 3. Reasoning only after evidence is established
    draft_answer = run_reasoning_based_on_evidence(query, evidence)

    # 4. Attach evidence and uncertainty markers
    final_answer = format_with_evidence_and_warnings(draft_answer, evidence)
    return final_answer
```

This pseudocode is intentionally abstract. It shows the **logical order**:

1. **Collect evidence** (primary first, then secondary if necessary),
2. **Refuse to answer without evidence**,
3. **Reason only on the basis of collected evidence**, and

4. **Format output with explicit references and warnings**.

---

7. Example pipeline – law/compliance scenario

1. User asks: *“Are printed books still in scope of Regulation X after the latest amendment?”*
2. AriVerify checks for user-provided sources:
3. If present → those are used first.
4. If not → ARC searches official domains (e.g. EUR-Lex, Parliament, Commission).
5. AriVerify extracts scope and exception clauses from the latest consolidated text.
6. If needed, ARC adds context from industry bodies or news, but cannot overrule the legal wording.
7. The answer is produced as:
8. **Part A – Evidence** (citations / paraphrased clauses),
9. **Part B – Interpretation** (what this means in practice),
10. **Part C – Uncertainty/warnings** (e.g. pending delegated acts, guidance not yet published).

---

8. Professional summary

AriVerify v6 defines a **strict evidence-first governance layer** for AI-assisted answers in high-stakes contexts. It:

- enforces **primary-source priority** over secondary and tertiary materials,
- aligns evidence extraction with the **user’s question type**,
- separates **Beleg (evidence)** from **Deutung (interpretation)**,
- integrates with ARC under a **Hard Override** (AriVerify > ARC > web search > reasoning),
- and forces explicit uncertainty whenever evidence is missing or ambiguous.

> **AriVerify v6 turns a generative model into an evidence-constrained assistant – suitable for compliance, contracts, and regulatory communication.**