DCA at DocVQA 2026

A Distributed Cognitive Architecture for Multi-Domain Document Reasoning

Author: Welf Wustlich (CTO), Planet AI, Rostock, Germany
Competition: DocVQA 2026 — Multimodal Reasoning over Documents in Multiple Domains (ICDAR 2026)
Category: Over 35B parameters
Contact: welf.wustlich@planet.de

The DCA paper family. The architectural argument is articulated across three papers — *Foundations* (P0), *Theory I* (P1), and a forthcoming *Theory II* (P2) — grounded empirically by a companion technical report; the four documents share one framework but differ in role:

```
dca_foundations.pdf  —  Biological Foundations · WHY  [companion paper, P0]
│   four architectural pillars · cortical correspondence catalog · grounds the theory papers
│
├─ dca_theory_i.pdf  —  Formal Theory I · WHAT  [companion paper, P1]
│  │   Pillar 1: Multi-Agent Dynamics  ·  Pillar 2: L0/L1/L2 Memory Hierarchy
│  │   convergence-signal framework · propositions 1–4
│  │
│  └─ docVQA_techrep.pdf  —  Empirical Anchor · EVIDENCE  [this report, published]
│        DocVQA 2026 @ ICDAR 2026 (>35B-parameter category) · deploys Pillars 1–2
│
└─ dca_theory_ii.pdf  —  Formal Theory II · WHAT  [forthcoming, P2]
      Pillar 3: Reference Frames  ·  Pillar 4: What/Where Pathways
      POSE embeddings · compositional retrieval
```

Links: *Foundations* (P0) — doi.org/10.5281/zenodo.20738104 · *Theory I* (P1) — doi.org/10.5281/zenodo.20732538 · *Theory II* (P2) — forthcoming · *DocVQA Technical Report* — doi.org/10.5281/zenodo.20707289

Abstract

Document Visual Question Answering (DocVQA) over heterogeneous document domains — from business reports and scientific papers to comics and maps — exposes a fundamental limitation of current Vision-Language Models (VLMs): no single model excels across all domains, and confident hallucinations on numbers and table structures are a primary source of incorrect answers. We present **DCA at DocVQA 2026**, our submission in the *>35B parameters* category at ICDAR 2026. The system combines (i) **IDA**, a layout-aware OCR engine producing structured Markdown that anchors text-heavy domains, (ii) **multi-perspective page reading** by independent VLMs (Gemini 3.1 Pro, Gemini 2.5 Pro, Sonnet 4, Qwen3.5) guided by model-adapted question reformulations, and (iii) **agentic reasoning** (Claude Opus 4.6) that synthesizes perspectives through domain-aware trust hierarchies and cross-perspective hallucination detection. The system is built on **Luna**, a cognitive AI platform implementing the **Distributed Cognitive Architecture (DCA)** — a framework that extends foundation models with memory, executive control, and convergent dynamics. Our submission achieves

60.00% accuracy, a +20 pp improvement over the best frontier-model baseline (~40%) in mixture-of-experts configuration; we attribute roughly +7 pp to IDA's deterministic text extraction and +13 pp to DCA's orchestration. Four observations emerge:
(i) frontier VLMs exhibit complementary, domain-dependent strengths that must be combined dynamically; (ii) a well-coordinated ensemble outperforms any individual frontier model; (iii) deterministic document analysis significantly reduces error propagation in text-heavy domains; and (iv) DCA supplies what foundation models structurally lack for robust reasoning — hierarchical memory, context steering, executive control, and continuous learning — turning a set of static models into a convergent cognitive system.

1. Introduction

Document Visual Question Answering (DocVQA) requires extracting precise answers from documents with complex layouts and heterogeneous content. The DocVQA 2026 competition (ICDAR 2026) raises the bar by spanning **eight heterogeneous domains** — business reports, scientific papers, slides, posters, maps, comics, infographics, and engineering drawings — each with distinct textual and visual demands. A representative question: *"What is the percentage change in revenue between 2023 and 2024?"* (business report); another: *"Who is shown holding the umbrella in panel 3?"* (comics). The eight domains together cover a strikingly broad range of formats and reasoning types.

No single foundation model performs uniformly well across these domains. Layout-aware OCR systems achieve near-perfect extraction on text-heavy documents but fail on visually complex content such as maps or comics. Vision-Language Models (VLMs) capture visual semantics but confidently hallucinate numbers and confuse table structures. Furthermore, the strongest model on one domain is rarely the
strongest on another, and these strengths shift with each model generation. Robust performance therefore requires *dynamic* combination of complementary perspectives, not selection of a single "best" model.

A naive ensemble of independent VLMs improves robustness but inherits the same structural limitations as its constituents: no persistent memory across pages, no coordinated context steering, no learning between tasks, and only rudimentary executive control (typically majority voting). Genuine multi-step reasoning across long, heterogeneous documents requires components that foundation models structurally lack:

Capability	Single Foundation Model	Naive Ensemble	DCA Agent (FM + Memory Controller)
Persistent memory	—	—	hierarchical memory (working / episodic / semantic levels)
Context steering	all-or-nothing prompt	uncoordinated	context retrieval policy
Continuous learning	frozen after training	frozen	prediction-error-driven adaptation (e.g. LoRA)
Executive control	—	rudimentary (e.g. majority vote)	Memory Controller as metacognitive layer

*__Table 1:__ Capabilities that foundation models — individually and in naive ensembles — structurally lack, and how the Distributed Cognitive Architecture (DCA) supplies them by coupling each FM with a Memory Controller (MC). These are *principal* DCA capabilities. Continuous learning operates in this submission at the **context level** — prediction-error-driven adaptation of prompts and context between iterations — whereas **weight-level** adaptation (e.g. LoRA, Table 5) is listed as future direction.*

This paper presents **DCA at DocVQA 2026**, our submission in the *>35B parameters* category. The system is built on Luna, a cognitive AI platform developed by Planet AI, implementing the **Distributed Cognitive Architecture (DCA)** — a framework that supplies the missing components in the rightmost column of Table 1 by coupling each foundation model with a Memory Controller, organizing memory into three hierarchical levels, and orchestrating ensembles of such agents through convergent recurrent dynamics. A detailed description of DCA is developed in a separate publication (Wustlich, 2026).
IDA (Intelligent Document Analysis) is Luna's layout-aware document parsing engine — a production technology with over a decade of R&D — providing deterministic, structurally accurate text extraction (PDF, images, DOCX, XLSX, PPTX → structured Markdown with tables, figures, and spatial metadata) that anchors the reasoning ensemble in ground-truth content.

Contributions. (i) A multi-perspective ensemble combining IDA-based deterministic OCR with model-adapted VLM page reads (Gemini 3.1 Pro, Gemini 2.5 Pro, Sonnet 4, Qwen3.5) and Claude-Opus-4.6-driven agentic reasoning. (ii) An attributed breakdown of how cognitive-architecture components contribute to performance — an estimated ≈ +7 pp from deterministic precision input (IDA) and ≈ +13 pp from cognitive orchestration (DCA), based on internal experiments rather than a controlled ablation (see §5.5). (iii) Empirical evidence that a coordinated team of frontier models, equipped with memory and executive control, substantially outperforms any individual frontier model on multi-domain DocVQA — reaching 60.00% accuracy versus ~40% for the best MoE baseline.

2. Related Work

Our work sits at the intersection of three lines of research: agentic document extraction, agentic reasoning frameworks around LLMs, and memory-augmented cognitive architectures.

Agentic document extraction. The most directly comparable contemporary approach is **LandingAI's Agentic Document Extraction (ADE)**, which reports 99.16% on the *classic* DocVQA benchmark by treating document QA as an agentic extraction task and reading documents text-first, without VLM image input (LandingAI, 2025).
Two observations distinguish our setting from theirs.

First, classic DocVQA is dominated by text-heavy documents where high-quality OCR plus reasoning is sufficient. The 2026 edition deliberately broadens the task to domains where text alone is insufficient — maps, comics, engineering drawings, and infographics — exposing the limits of OCR-only approaches and requiring genuine multimodal reasoning. Both our system and ADE rely on layout-aware text extraction (IDA in our case, an internal extraction component in theirs), but our reader ensemble actively combines this extraction with VLM-based visual perspectives, while ADE deliberately avoids image input. We expect ADE-style text-only approaches to remain strong on the four text-heavy domains of DocVQA 2026, but to be structurally limited on the four visually dominated ones.

Second, ADE positions itself as an end-to-end agentic extraction pipeline; our system positions itself as a cognitive-architecture-based ensemble (DCA) wrapping multiple frontier foundation models. The two approaches differ in where intelligence is concentrated: ADE optimizes a single agentic pipeline, while DCA orchestrates an ensemble of complementary readers with cross-perspective conflict resolution. We return to this distinction in §5.

Agentic reasoning frameworks. A growing family of techniques wraps LLMs in iterative reasoning loops: *ReAct* (Yao et al., 2023) interleaves chain-of-thought reasoning with external tool calls; *Chain-of-Thought prompting* (Wei et al., 2022) elicits intermediate reasoning steps; *Reflexion* (Shinn et al., 2023) adds self-verbal reflection on past trajectories. These share the broad pattern of iteration over an external action space but lack persistent structured memory and formal convergence criteria. DCA differs in providing a structured memory hierarchy and convergence-monitored loop termination; details are in the DCA companion publication (Wustlich, 2026).

Memory-augmented LLMs. Memory has been added to LLM systems through various mechanisms — most prominently *MemGPT* (Packer et al., 2023), which paginates context to support virtual long-term memory, and retrieval-augmented generation (RAG) variants. These operate primarily at the level of *external retrieval*, treating context as a fetchable resource. DCA's memory model is structurally tighter (working, episodic, and semantic levels co-evolve with the world model in a coupled loop), but the present paper does not develop that distinction — it is the focus of the companion publication.

VLM ensembles for document understanding. Mixture-of-experts and ensemble configurations of vision-language models are an active area of practical work in document understanding. The ~40% MoE baseline on DocVQA 2026 (§5.1) represents the upper end of what naive ensembles currently achieve on this benchmark.

The DocVQA task and the ANLS scoring used here were introduced by Mathew et al. (2021) and form the evaluation basis (§3.6), not the methodological context for this work.

---

3. Method

3.1 Architecture Overview

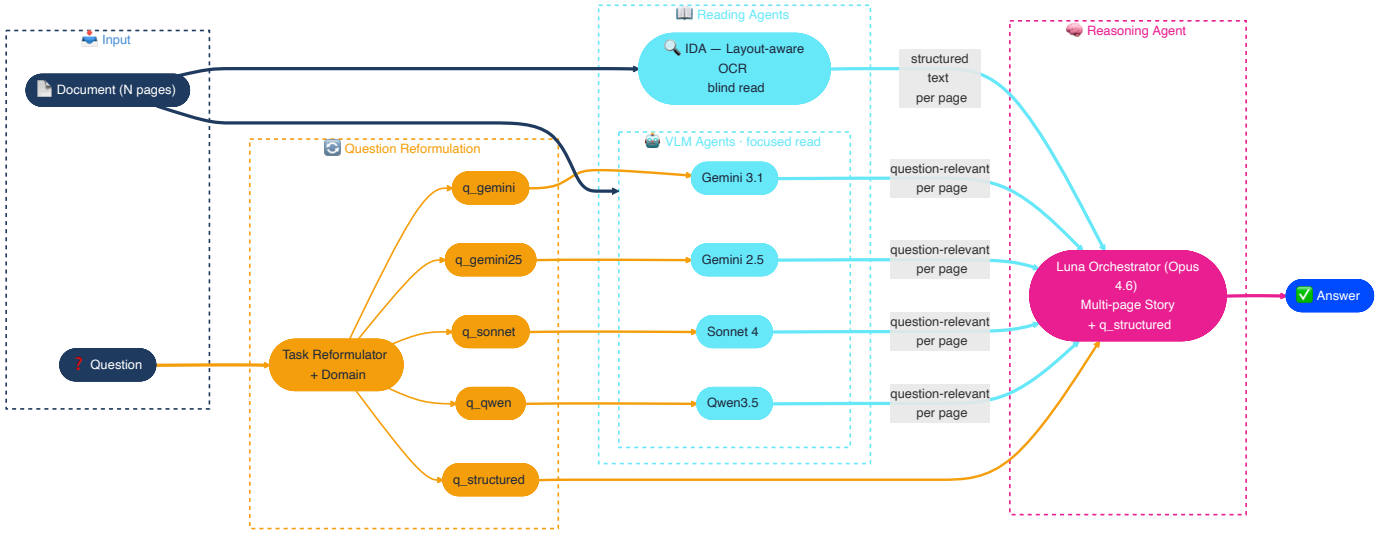


Figure 1: *Architecture overview of DCA at DocVQA 2026. IDA performs a blind read (no question), VLM agents perform focused reads guided by model-adapted question reformulations, and the Reasoning Agent synthesizes all perspectives into the final answer.*

The pipeline implements four design principles:

Reusable deterministic foundation. IDA receives the document only (no question) and produces structured Markdown that is cached per document — reusable across all questions for the same document, amortizing OCR cost.

Question-guided focused reads. Each VLM reader receives the document pages together with a model-adapted reformulation of the question; this attention-guidance shifts the model's emphasis toward question-relevant content.

Centralized reasoning over distributed evidence. The Reasoning Agent (Claude Opus 4.6) consumes the complete multi-page evidence collection, with each excerpt labeled by source agent and page number. It resolves conflicts and synthesizes the final answer according to domain-specific trust hierarchies (§3.2).

Reflection and memory management. Both Reading Agents and the Reasoning Agent maintain working state across pages — accumulating cross-page context to resolve entity references, track quantitative claims, and detect contradictions — and perform self-assessment of extraction quality and confidence. These capabilities derive from Luna's memory hierarchy and Memory Controller (Table 1).

Feedback loop (not shown). The Reasoning Agent may trigger a re-evaluation cycle through the Task Reformulator in approximately 10% of questions — typically when it detects conflicting evidence or low-confidence extractions. A representative example:

> "Gemini reports 17.65% while Sonnet extracts 6.25%, each with plausible but divergent reasoning — re-validate against the bar chart in Figure 3 on page 12 and provide detailed reasoning for the extracted value."

The reformulated task is routed back to the relevant reader agents with refined attention focus, producing a second-pass answer that the Reasoning Agent incorporates into its final decision.

3.2 Reader Agent Battlecard

The reader agents — IDA and five frontier VLMs — exhibit distinct, complementary strengths across the eight DocVQA 2026 domains. We summarize these empirically observed strengths in a *battlecard* (Table 2) that drives ensemble weighting and conflict resolution.

Domain	IDA	Gem3.1	Gem2.5	GPT5	Sonnet4	Qwen3.5	Best zero-shot	Trust priority
business_report	🟩	🟨	🟥	🟨	🟩	🟨	GPT (0.60)	**IDA primary** — tables, numbers, multi-page
science_paper	🟩	🟨	🟥	🟨	🟩	🟨	GPT (0.40)	**IDA primary** — formulas, references
slide	🟩	🟩	🟨	🟨	🟨	🟨	Gem 3.1 (0.70)	**IDA + VLM** — mixed text/visual
science_poster	🟨	🟨	🟩	🟥	🟩	🟩	Gem 2.5 (0.50)	**VLM ensemble** — dense visual layout
infographics	🟨	🟩	🟩	🟨	🟩	🟨	Gem 2.5 / 3.1 (0.70)	**VLM + IDA** — charts need vision + numbers
maps	🟥	🟥	🟥	🟥	🟥	🟥	GPT (0.20)	**VLM focused** — all models weak
comics	🟥	🟩	🟨	🟥	🟨	🟨	Gem 3.1 (0.65)	**Gemini primary** — 1M context
engineering_drawing	🟥	🟨	🟥	🟨	🟨	🟥	GPT / Gem 3.1 (0.30)	**VLM ensemble** — symbols, dimensions

Table 2: *Reader-agent strengths by domain (qualitative three-tier color code: 🟥 weak, 🟨 medium, 🟩 strong), derived from internal validation. The "best zero-shot" column reports the strongest single-model zero-shot score observed; the "trust priority" column summarizes which agent the Reasoning Agent prioritizes when perspectives conflict.*

Key observations. Three patterns are visible in the battlecard. *First,* no single model dominates: each of the strongest agents (IDA on text-heavy domains, Gemini 3.1 on comics and slides, Gemini 2.5 on science posters and infographics) wins in a distinct subset of domains. *Second,* the two Gemini generations exhibit complementary strengths — 2.5 leads on science posters (+0.20), 3.1 leads on slides (+0.30) and comics — illustrating that even within a single model family, strengths shift with generation. *Third,* maps emerge as the hardest domain (best zero-shot 0.20), with no agent providing reliable extraction; this reflects a persistent gap in current VLMs for spatial-symbolic reasoning.

GPT-5: high raw score, low ensemble compatibility. GPT-5 achieves the highest raw zero-shot score on business reports (0.60) and is competitive on engineering drawings, yet we exclude it from the active reader ensemble. With the default GPT-5 API settings we tested, the model commits to answers without exposing intermediate reasoning, which makes conflict detection and targeted re-prompting substantially harder than for planning-oriented models such as Gemini, Sonnet, or Claude Opus 4.6. In a single-model setting GPT-5 is a strong contender; in an orchestrated ensemble, the absence of intermediate reasoning is a structural liability — a recurring

observation we return to in §4.

Agent profiles. Table 3 summarizes each reader's type, context window, and salient strengths and weaknesses.

Agent	Type	Context	Strengths	Weaknesses
IDA	OCR + Layout	unlimited	Deterministic text, tables, numbers, formulas	No visual understanding
Gemini 3.1 Pro	VLM	1M tokens	Comics (0.65), slides (0.70), spatial reasoning	Hallucinates numbers; maps (0.00)
Gemini 2.5 Pro	VLM	1M tokens	Science posters (0.50), infographics (0.70), thinking-style reasoning	Weaker on business reports, slides
GPT-5	VLM	128K tokens	Business reports (0.60), science (0.40)	Frequent hallucinations; posters (0.00); direct-response architecture (see above)
Sonnet 4	VLM	200K tokens	Precise extraction, instruction following, fast	Conservative, tends toward "Unknown"
Qwen3.5	VLM	256K tokens	Native multimodal, science posters (0.50), open-source	Less validated on maps, newer model

Table 3: *Reader-agent profiles. Context windows reported as of submission date.*

Conflict resolution. When reader outputs disagree, the Reasoning Agent applies the trust order in Table 4, derived from the strength patterns of Tables 2 and 3. The active reader ensemble comprises IDA and the four VLMs Gemini 3.1, Gemini 2.5, Sonnet 4, and Qwen3.5; GPT-5 is excluded for the reasons above.

Domain type	Trust order	Rationale
Text-heavy (business, science, slides)	IDA > Sonnet > Qwen > Gemini 3.1 > Gemini 2.5	Deterministic OCR as ground-truth anchor
Visual-heavy (maps, comics, engineering)	Gemini 3.1 > Gemini 2.5 > Qwen > Sonnet > IDA	Gemini 3.1 strongest; IDA blind to visual cues
Mixed (posters, infographics)	Gemini 2.5 > majority vote (3 of 4)	Gemini 2.5 strongest in this regime; consensus for robustness

Table 4: *Trust order by domain type, applied by the Reasoning Agent to resolve cross-perspective conflicts. The active ensemble has four VLMs plus IDA; GPT-5 is not included.*

3.3 Pipeline Phases

The pipeline runs in four phases — question reformulation, multi-perspective page reading, agentic reasoning, and strict answer formatting — described in turn below.

3.3.1 Phase 0: Question Reformulation

Prior to reader and reasoning processing, a dedicated **Question Reformulation** module transforms the raw user question into optimized variants. The motivation is structural: DocVQA 2026 questions are written for humans, not for models, and different VLMs respond best to different phrasings. The module pursues three objectives.

1. Reasoning-oriented restructuring. The original question is rephrased to make the reasoning chain explicit. For example, *"What is the percentage change in revenue between 2023 and 2024?"* becomes a structured decomposition: *"(1) Find the revenue for 2023. (2) Find the revenue for 2024. (3) Compute the percentage change."* This chain-of-thought scaffolding reduces reasoning errors, especially for multi-step questions involving calculations or cross-page lookups.

2. Multi-aspect sub-questions. When the original question conflates multiple information needs, the module generates complementary sub-questions that emphasize different aspects. For example, *"Describe the population distribution shown on the map"* may yield:

- *"What regions are labeled on the map and what are their population values?"* (extractive)
- *"What visual patterns (color gradients, symbol sizes) indicate population density?"* (visual)

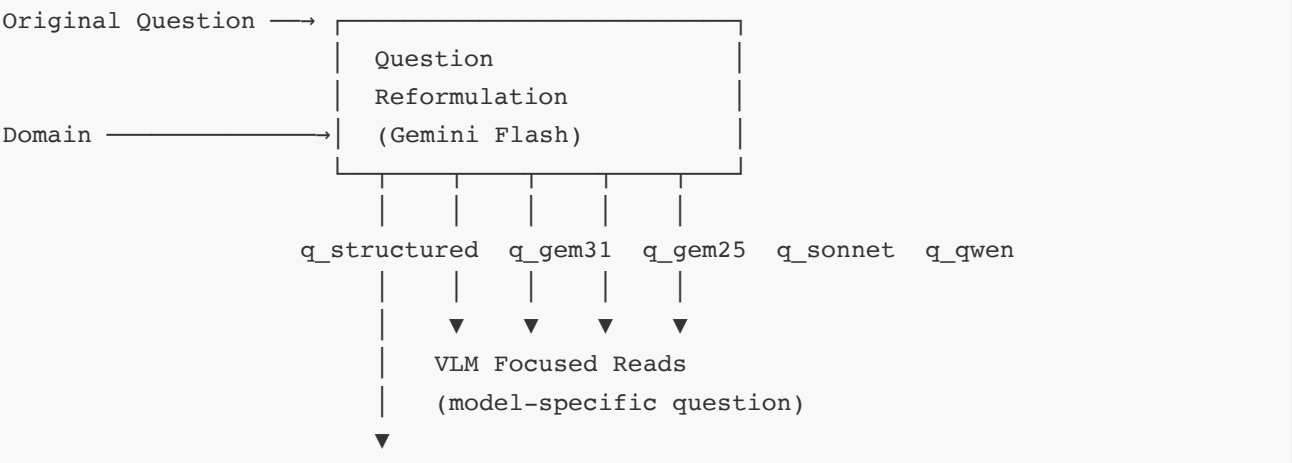
Each sub-question steers a different reader toward the relevant evidence, increasing recall.

3. Model-adapted phrasing. Each VLM has distinct prompt sensitivities. The module produces tailored question variants per target model:

- **Gemini:** concise, vision-first phrasing — *"Look at the image and describe..."*
- **Sonnet:** precise, instruction-style — *"Extract the exact value of ... from the table in ..."*
- **Qwen:** structured, step-by-step — *"Analyze the visual content systematically and identify..."*

IDA receives no question (blind read), so reformulation does not apply to it.

Implementation. Reformulation is performed by a single lightweight LLM call (Gemini Flash) that takes the original question, the document domain, and the target model as input. The cost is negligible — one short text-only call per question — and the reformulated variants are routed to the focused VLM reads and to the Reasoning Agent (Fig. 2).



```
                    Reasoning Agent
                    (structured question + all perspectives)
```

Figure 2: *Question Reformulation pipeline. A lightweight LLM call produces a structured reasoning decomposition (q_structured) and model-adapted question variants for each VLM reader.*

3.3.2 Phase 1: Multi-Perspective Page Reading

In Phase 1, each page of the document is processed independently by the reader engines.

IDA blind read. IDA receives the document only (no question) and produces a structured Markdown representation per page — text, tables, figures, and spatial metadata. The output is computed once per document and cached, so it is reused without recomputation across all questions for that document.

VLM focused reads. Each VLM reader receives the page together with its model-adapted question variant from §3.3.1. The question acts as attention guidance, directing the description toward question-relevant content rather than a generic page summary. This focused-read regime substantially improves recall on long pages with dense content, where a blind summary would inevitably omit relevant details.

Page reads run in parallel across providers but *sequentially within each provider*: each page read sees the prior page history plus the question, maintaining a memory chain across the document (§3.4.1). The output of Phase 1 is a labeled collection of page-level descriptions — one per (reader, page) pair — passed to the Reasoning Agent in Phase 2.

3.3.3 Phase 2: Agentic Reasoning

In Phase 2, all page descriptions from all reader perspectives, together with the structured question variant *q_structured*, are passed to the **Reasoning Agent** — a Claude Opus 4.6 instance chosen for its instruction following, multi-step reasoning, and large context window. The agent sees the full multi-page evidence collection, with each excerpt labeled by source reader and page number, and applies the domain-specific trust order (Table 4) to resolve conflicts. When evidence is sufficient and consistent, the Reasoning Agent commits to a final answer; when it is conflicting or low-confidence, the agent triggers a re-evaluation cycle through the Task Reformulator (§3.1) before committing.

3.3.4 Phase 3: Strict Answer Formatting

DocVQA 2026 uses the **ANLS** metric (Average Normalized Levenshtein Similarity; Mathew et al., 2021), in which surface-formatting deviations directly reduce the per-question score. A rule-based post-processing layer enforces ANLS-compliant output:

- Dates: `YYYY-MM-DD`
- Numbers: period as decimal separator, no thousands separator
- Units: standardized abbreviations (`kg`, `USD`, `%`)
- Unanswerable items: exactly `"Unknown"`
- No filler text, no explanatory prefixes

Formatting is applied as a deterministic transform on top of the Reasoning Agent's answer; the agent itself need not produce ANLS-compliant text, which separates semantic correctness from surface formatting.

3.4 Cross-Cutting Concerns

Three aspects span the pipeline as a whole: parallelization (§3.4.1), performance and cost (§3.4.2), and hallucination mitigation (§3.4.3).

3.4.1 Parallelization

```
Per question (N pages):

IDA:       [═══════ cached per document ═══════]    1 call, reused

Gem 3.1:   [p1]→[p2]→[p3]→...→[pN]    ┐
Gem 2.5:   [p1]→[p2]→[p3]→...→[pN]    │  N sequential calls per model
Sonnet:    [p1]→[p2]→[p3]→...→[pN]    │  (page memory chain);
Qwen:      [p1]→[p2]→[p3]→...→[pN]    ┘  4 models run in parallel

           ──────────────────────────────────→ time

Reasoning:                            [══ 1 call (all perspectives) ══]
```

***Figure 3:** Parallelization strategy. Within each VLM, page reads run *sequentially* to maintain a memory chain (each page read sees the prior page history plus the question). The four VLMs run in parallel relative to each other. IDA is cached per document and reused across all questions for that document. The Reasoning Agent invocation runs only after Phase 1 completes and is the only strictly cross-phase sequential step.*

Concurrency is bounded by a semaphore at 40 parallel calls per provider, balancing throughput against API rate-limit budgets.

3.4.2 Performance and Cost

Latency. Owing to aggressive parallelization — the four VLM readers run in parallel, and IDA outputs are cached across questions — wall-clock time per document is comparable to a single-model zero-shot run with Gemini. For a typical 10-page document with three questions, the multi-perspective ensemble completes in roughly the same time as the Gemini baseline despite generating 3–4× more API calls. The bottleneck shifts from sequential processing to provider rate limits.

Token cost. The ensemble consumes approximately **4–5× the tokens** of a single-model baseline: each page is read independently by four VLM agents, and the Reasoning Agent receives all perspectives as input. For the full competition (160 questions across 48 documents), this amounts to significant but manageable API spend — justified by the accuracy gains from multi-perspective consensus.

Optimization avenues for accuracy, latency, and cost — none exploited in the present submission — are summarized as outlook in §5.6.

3.4.3 Hallucination Mitigation

Hallucination is the largest risk to answer correctness in multi-model document VQA: VLMs confidently fabricate numbers, invent table entries, and produce text that does not appear in the document. Our system addresses this risk at three levels.

Forced reasoning in reader prompts. Each VLM reader is instructed to explain its extraction *before* stating the answer — a form of chain-of-thought grounding. Rather than returning a bare value, the reader must cite where on the page the information was found (e.g., *"Table 2, row 3, column 'Revenue 2024'"*) and describe the reasoning steps that produced the extracted value. This commits the model to a source location, making unsupported claims visible to the downstream Reasoning Agent. In practice, readers that justify their extractions hallucinate measurably less than those asked for direct answers.

Cross-perspective conflict detection. The multi-agent ensemble is inherently a hallucination detector: when several independent readers process the same page, hallucinated content rarely appears in more than one perspective. The Reasoning Agent explicitly compares reader outputs and flags disagreements. If IDA's deterministic OCR reports a table value of "12,450" while a VLM claims "14,250", the conflict is surfaced and resolved in favor of the higher-trust source (Table 4).

Orchestrator guidelines for critical answers. For numerical values, dates, and named entities — answer types where hallucination is most damaging — the Reasoning Agent is guided by three heuristics rather than a strict rule: prefer consensus where two sources agree, fall back to the highest-trust single source (typically IDA for text-heavy domains, Gemini 3.1 for visual domains) with lowered confidence, or abstain via "Unknown" if no source provides a confident, well-grounded answer. The choice is context-sensitive per question.

---

4. Observations

Working with five frontier readers across eight document domains yields four observations that motivate our architectural choices and inform broader claims about deploying foundation models in production reasoning systems.

Visual reasoning has reached remarkable maturity. Current frontier VLMs — Gemini 3.1 Pro, Gemini 2.5 Pro, and Claude Opus 4.6 among them — exhibit visual reasoning capabilities that were considered infeasible only recently: spatial interpretation of charts, table-structure recognition, cross-page entity tracking, and multi-step inference over heterogeneous layouts. On individual instances, these models match or exceed dedicated extraction pipelines for infographics, engineering drawings, and historical maps. The capability ceiling is no longer the primary bottleneck.

The bottleneck has shifted from capability to controllability. The decisive obstacle in deploying high-capability foundation models is no longer raw performance but error detection, hallucination management, and context orchestration. The models possess the requisite knowledge and reasoning capacity; what determines robust system behavior is the ability to (i) detect when a model hallucinates or produces unreliable outputs, (ii) manage context windows and attention across long, multi-page documents, and (iii) dynamically

route between complementary perspectives. A complementary observation has been argued by Apple Research (2025) in *The Illusion of Thinking*: reasoning models reliably solve simple problems but collapse beyond moderate complexity — exposing the limits of shallow iteration in the absence of adaptive control, abstraction, and self-reflection. These are precisely the capabilities that Luna's cognitive architecture supplies (Table 1).

Planning-oriented models offer a structural advantage in ensembles. We observe a meaningful behavioral difference between *direct-response models* (e.g., GPT-5) and *planning-oriented models* (e.g., Claude Opus 4.6, Gemini 2.5 Pro with extended thinking). Direct-response architectures produce confident but brittle answers — hallucinations occur more frequently and are harder to detect, because the model commits to a response path without explicit deliberation. Planning-oriented models, by contrast, expose intermediate reasoning steps, enabling external validation, conflict detection, and targeted re-prompting. This structural transparency makes them substantially more amenable to orchestration in multi-agent pipelines (cf. our exclusion of GPT-5 from the active ensemble, §3.2).

DCA operationalizes these observations. The Distributed Cognitive Architecture (DCA) — a general-purpose framework for orchestrating foundation models through structured cognitive workflows — supplies the operational components implied by the observations above: cross-perspective hallucination detection, adaptive context management across document scales, and dynamic strategy selection based on domain characteristics. A detailed description of DCA is developed in a separate publication (Wustlich, 2026).

---

5. Results & Discussion

5.1 Overall Performance and Value Chain

Our submission achieves **60.00% accuracy** on the DocVQA 2026 leaderboard (>35B category). Two reference baselines on this leaderboard contextualize that number:

- **Strongest single-model zero-shot baseline:** Gemini 3.1 Pro at **37.50%** (official baseline entry on the leaderboard).
- **Simple mixture-of-experts (MoE) configurations:** naive majority voting or similar aggregation across frontier VLMs reaches approximately **40%** (our estimate from internal experiments).

We report a +20 pp improvement over the ~40% MoE baseline. Figure 4 shows our *attributed* decomposition of this gain into two architectural components: IDA's deterministic text extraction (≈ +7 pp) and DCA's cognitive orchestration (≈ +13 pp). The decomposition is an estimate based on internal experiments rather than a controlled-ablation measurement (see §5.5). We discuss each component in turn.

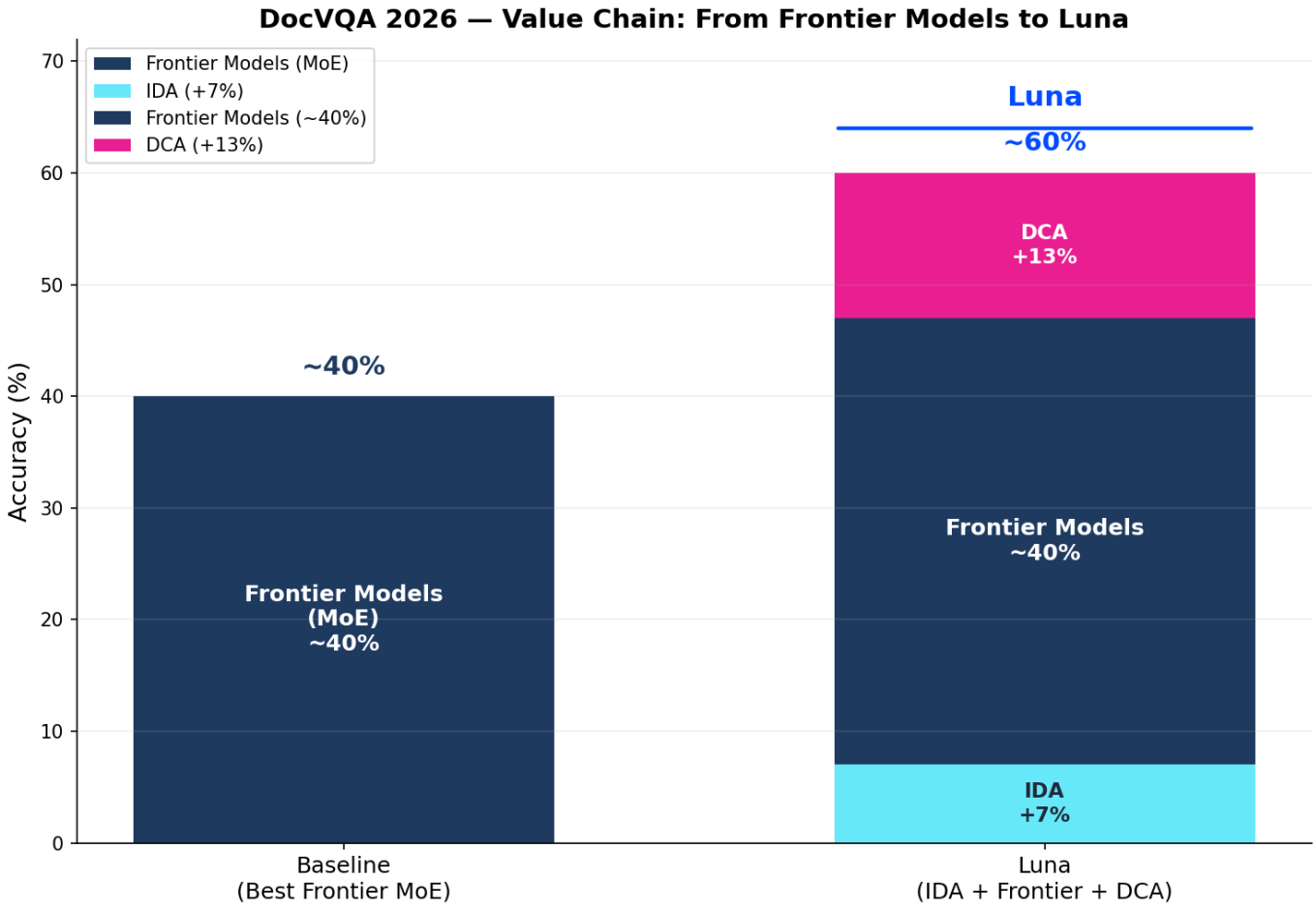


Figure 4: *Value chain from frontier-model baseline to Luna. Left: best frontier models in mixture-of-experts configuration (~40%). Right: Luna's three-layer architecture — IDA adds deterministic text extraction (≈ +7 pp), frontier models supply visual reasoning (~40%), and DCA orchestrates the system through task reformulation, agentic reasoning, reflection, and memory management (≈ +13 pp). Attributed decomposition (not measured via controlled ablation); see §5.5 Limitations.*

5.2 Why IDA Anchors the System (attributed ≈ +7 pp)

IDA is Planet AI's layout-aware document analysis engine — a production technology with over a decade of R&D, originally developed in European research collaborations (FP7, Horizon 2020) and now in industrial use. The underlying technology has previously won seven international competitions at ICDAR and ICFHR (2014–2019) in layout analysis, baseline detection, handwritten text recognition, keyword spotting, and information extraction.

For DocVQA 2026, IDA serves as a **precision input pipeline** rather than an answering engine: it does not answer the competition questions, but supplies the accurate textual foundation against which the Reasoning Agent constructs and validates its conclusions. While VLMs hallucinate numerical values, confuse table rows, and miss footnotes, IDA delivers structurally accurate text via layout-aware OCR, precise table extraction, figure detection, and spatial metadata. The output is structured Markdown, consumable directly by downstream reasoning.

The attributed ≈ +7 pp is concentrated in **text-heavy domains** — business reports, scientific papers, slides — where exact extraction of numbers, dates, and table values determines whether the Reasoning Agent can arrive at and validate the correct answer. Two qualitative observations support this attribution: the battlecard in Table 2 shows IDA as the strongest reader in these domains, and the trust hierarchy in Table 4 makes IDA the ground-truth anchor for the text-heavy domain class. We do not claim a measured per-component decomposition; the IDA contribution is argued qualitatively (see §5.5).

5.3 Why Cognitive Orchestration Helps (attributed ≈ +13 pp)

The ≈ +13 pp that we attribute to DCA on top of frontier models is not a prompting trick or a majority-voting strategy. It is the contribution of a **cognitive architecture** that supplies what foundation models structurally lack (Table 1). We cannot measure the contribution of each DCA capability separately (see §5.5); what follows is a qualitative plausibility argument supported by evidence from Tables 4 and 6, §3.4.3 (hallucination mitigation), and the Run-1 → Run-2 comparison in §5.4.

DCA structures the system around three coupled components: a **World Model** (the foundation model performing inference), a **Memory system** (working, episodic, and semantic levels), and a **Controller** (executive orchestration: what to do, how to route, when to stop). The recursive coupling of World Model and Memory turns single-pass inference into an iterative refinement loop, in which the system converges on a stable answer through repeated cross-perspective comparison rather than from a single forward pass.

For DocVQA 2026, four DCA capabilities contribute to this ≈ +13 pp:

Task reformulation. The Controller decomposes questions into model-adapted sub-queries, steering each VLM's attention toward the evidence it is best equipped to extract (§3.3.1).

Agentic reasoning with convergent dynamics. The Reasoning Agent does not produce a single-pass answer. It iterates — comparing perspectives, detecting conflicts, re-querying when evidence is insufficient — driven toward a stable answer rather than oscillation or drift. The visible effect of additional iteration is captured by the Run-1 → Run-2 comparison: deeper convergence iterations and parallel sub-queries lift the score by ~4.4 pp without any change to the underlying model stack (§5.4, Table 6).

Reflection and hallucination detection. Each agent assesses its own extraction quality, and cross-perspective conflict detection surfaces hallucinations that would be invisible to a single-model system. A substantial fraction of the ≈ +13 pp derives from *not accepting wrong answers* rather than *finding additional right ones* (§3.4.3).

Memory and context management. Across multi-page documents, the memory system accumulates cross-page context — tracking entities, resolving references, and maintaining coherence beyond any single model's context window. The Run-1 → Run-2 refinement exploits exactly this mechanism: the first run's reasoning outputs serve as prior context for the second, equivalent to an additional outer recursion (§5.4).

A formal treatment of DCA's iteration dynamics and memory architecture is developed in a separate publication (Wustlich, 2026).

5.4 Baseline to Refined: 55.63% → 60.00%

Our first submission (55.63%) represented the system's default configuration — the full Luna pipeline (IDA + VLM ensemble + DCA reasoning) running with uniform settings across all domains. The second submission (60.00%) built directly on the first run's results, using them as prior context for a targeted refinement pass with three enhancements. The reuse is within-system only: no leaderboard feedback entered the refinement (the leaderboard returns only an aggregate score, which is too sparse to drive test-set adaptation), and we submitted exactly twice. Methodologically, Run 2 is equivalent to extending Run 1 by one additional outer recursion — a continuation that takes the first run's reasoning outputs as warm start rather than a fresh start.

Focused image crops. The first run revealed that global VLM reads on large, content-dense pages often miss fine-grained details — a number in a small table cell, a label on an engineering drawing, a data point in a chart. The refined run provides cropped image regions to the VLMs on request, sharpening attention on specific visual elements.

Deeper reasoning iterations. With the first run's answers as prior context, the Reasoning Agent ran significantly more convergence iterations on challenging questions — allowing the system to explore alternative interpretations and reach higher-confidence answers rather than committing to the first plausible result.

Parallel sub-queries. The Task Reformulator generated multiple parallel sub-queries per question, increasing evidence coverage. Where the first run might approach a question from one angle, the refined run systematically probes from several.

The refinements produced changes in exactly the domains where they apply (Table 6).

Domain	Changed	Why
engineering_drawing	6/20	Focused crops on technical details, dimensions, labels
science_poster	5/20	Dense visual layouts benefit from sub-region focus
infographics	4/20	Chart values resolved through cropped detail views
business_report	2/20	Targeted improvements on specific table lookups
comics	0/20	Narrative panels — crops and sub-queries do not add value
maps	0/20	Spatial reasoning — unchanged by visual refinements
science_paper	0/20	Already well-served by IDA's text extraction
slide	0/20	Bulletpoint content — stable across runs

Table 6: *Per-domain answer-change count between the baseline (55.63%) and refined (60.00%) submissions. The first run's answers served as prior context; the Reasoning Agent confirmed existing answers in domains where the new capabilities offered no improvement, avoiding unnecessary variance.*

The four unchanged domains (comics, maps, science_paper, slide) reflect the system architecture working as designed: where new capabilities offered no improvement, the Reasoning Agent confirmed the existing answers rather than introducing variance.

Trade-off. The refined run consumed approximately 5–8× more compute time and tokens than the baseline — a cost justified in a competition setting but relevant for production deployment, where selective application of deep iterations to low-confidence questions would be the preferred strategy.

5.5 Limitations

Several aspects of this study warrant explicit acknowledgement.

No controlled ablation. The +7 pp / +13 pp decomposition reported in §5.1–§5.3 is an *attributed* breakdown rather than a measured one. We did not run separate configurations of the system (e.g. "frontier-only", "+IDA without DCA orchestration", "+IDA + DCA") on the competition test set; the components are too tightly coupled in the operational pipeline to be cleanly switched on or off without rebuilding parts of the system. The attribution rests on internal experiments and on the qualitative pattern that IDA's contribution is concentrated in text-heavy domains while DCA-orchestration effects span all domains. A controlled ablation is planned as follow-up work.

Sample size. DocVQA 2026 contains 160 questions. With this n, the 95% Wilson-score confidence interval for a 60.00% accuracy is approximately ±7 pp. We report the leaderboard value with two decimal places because this is how the official CVC tables report it, not because we claim precision at that resolution.

Per-domain results pending. The official per-domain ANLS breakdown of our submissions will be published by the competition organizers; this report uses only the aggregate 60.00% accuracy and the qualitative domain pattern in Table 2.

DCA theory is out of scope for this paper. Formal foundations of DCA — convergence properties, iteration dynamics, memory model — are not developed here; they are the subject of a companion publication (Wustlich, 2026). The present report uses DCA in an operational sense only: as the architectural framework around foundation models that supplies memory, executive control, and iterative refinement.

Single ensemble configuration. All numbers above stem from a single combination of reader and reasoning models, run with uniform settings across all eight domains. We did not perform systematic prompt tuning, per-domain reader specialization, or hyperparameter optimization. Substantial optimization headroom likely remains; concrete avenues are summarized as outlook in §5.6.

5.6 Outlook

This submission represents an early configuration of the multi-agent system. No systematic prompt tuning on the validation set or per-domain reader specialization has been performed; the system runs with uniform reader configurations across all domains. Table 5 summarizes the most promising optimization directions, none of which were exploited here.

Dimension	Approach	Expected impact
Accuracy	LoRA fine-tuning of reader agents on domain-specific data; specialization of sub-agents per document type	+10–20 pp on weak domains (maps, comics)
Latency	Smarter page selection (skip irrelevant pages); domain-aware routing (fewer readers for text-heavy documents)	2–3× speedup on large documents
Cost	Replace frontier VLMs with fine-tuned open models (e.g., Qwen3.5) for the reader stage; retain Opus 4.6 only for reasoning	5–10× cost reduction with minimal accuracy loss

Table 5: *Tuning opportunities not yet exploited in the current submission.*

---

6. Conclusion

We presented **DCA at DocVQA 2026**, a submission combining deterministic document analysis (IDA), multi-perspective page reading by frontier VLMs, and agentic reasoning orchestrated through the Distributed Cognitive Architecture (DCA). The system reaches 60.00% accuracy versus a ~40% best-frontier-model
baseline, with the +20 pp gap attributed to two architectural components rather than to larger or better foundation models: IDA contributes ≈ +7 pp by anchoring text-heavy domains in structurally accurate text, and DCA contributes ≈ +13 pp through task reformulation, agentic reasoning, reflection, and memory management. We label these decompositions as *attributed* rather than measured because no controlled ablation was conducted; see §5.5.

Architecture as the active variable. The frontier models that populated our reader ensemble — Gemini 3.1 Pro, Gemini 2.5 Pro, Sonnet 4, Qwen3.5, and Claude Opus 4.6 as the Reasoning Agent — are the same models that produced the ~40% baseline. The +20 pp improvement is therefore attributable to the architectural layer surrounding them, suggesting that for complex multi-domain reasoning the marginal return of scaling model size is diminishing while the return of scaling architecture is just beginning.

Early configuration; substantial headroom. Both submissions were produced without systematic prompt tuning, domain-specific reader specialization, or hyperparameter optimization. Concrete optimization avenues for accuracy, latency, and cost are summarized in §5.6 (Table 5).

Formal foundations forthcoming. A formal treatment of DCA's iteration dynamics and memory architecture is developed in a separate publication (Wustlich, 2026). DocVQA 2026 serves as a first empirical validation of DCA principles in a
competitive setting.

The frontier models supply the intelligence; the cognitive architecture supplies what they structurally lack.

Acknowledgments

This work would not be possible without the extraordinary capabilities of the frontier foundation models that populate Luna's reader and reasoning agents — Google's Gemini family, Anthropic's Claude family, and Alibaba's Qwen. DCA does not replace these models; it depends on them. Their creators deserve credit for the substrate on which the architectural contributions reported here are built.

Luna as competition participant. The Luna system itself executed the DocVQA 2026 competition pipeline end-to-end — performing layout-aware document analysis through IDA, multi-perspective reading through the VLM ensemble, and final answer synthesis through the reasoning agent. All answers submitted to the official DocVQA 2026 leaderboard were generated autonomously by Luna. The human author designed the system, supervised its operation, and is responsible for the methodological framing and reporting of results presented in this paper.

Use of AI tools in manuscript preparation. Luna, the cognitive AI platform developed by Planet AI and described as the subject of this paper, was also used during the preparation of this manuscript for drafting, analysis, and editorial review. Final accountability for all content, methodological decisions, reported results, and conclusions lies solely with the human author.

References

Benchmark and competition.

- Mathew, M., Karatzas, D., & Jawahar, C. V. (2021). DocVQA: A Dataset for VQA on Document Images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV 2021)*.
- DocVQA 2026 Competition (ICDAR 2026). docvqa.org/challenges/2026
- DocVQA 2026 Dataset. HuggingFace VLR-CVC/DocVQA-2026
- RRC Submission Platform (Channel 34). rrc.cvc.uab.es

Foundation models in the reader and reasoning ensemble.

- **Gemini 3.1 Pro**, **Gemini 2.5 Pro** — Google DeepMind. Technical report for the Gemini family: Gemini Team (2024), *Gemini 1.5: Unlocking Multimodal Understanding across Millions of Tokens of Context.* arXiv:2403.05530. Model cards at deepmind.google/technologies/gemini.
- **Claude Opus 4.6** — Anthropic (2026). *Claude Opus 4.6 System Card* (February 2026). PDF on anthropic.com
- **Claude Sonnet 4** — Anthropic. Model documentation at anthropic.com/claude.

- **Qwen3.5** — Alibaba (Qwen Team). Technical report for the Qwen3 family: Qwen Team (2025), *Qwen3 Technical Report.* arXiv:2505.09388.
- **GPT-5** — OpenAI (evaluated but excluded from the active ensemble, §3.2). Model documentation at openai.com.

Layout-aware document analysis.

- **IDA — Intelligent Document Analysis**, Planet AI.

Related work and motivation.

- LandingAI (2025). *DocVQA Benchmark: 99.16% Accuracy Using Agentic Document Extraction.* Blog post. landing.ai · reproducible benchmark on GitHub
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). *ReAct: Synergizing Reasoning and Acting in Language Models.* In *International Conference on Learning Representations (ICLR 2023)*.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models.* In *Advances in Neural Information Processing Systems (NeurIPS 2022)*.
- Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., & Yao, S. (2023). *Reflexion: Language Agents with Verbal Reinforcement Learning.* In *Advances in Neural Information Processing Systems (NeurIPS 2023)*.
- Packer, C., Wooders, S., Lin, K., Fang, V., Patil, S. G., Stoica, I., & Gonzalez, J. E. (2023). *MemGPT: Towards LLMs as Operating Systems.* arXiv preprint arXiv:2310.08560.
- Apple Research (2025). *The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity.* ml-site.cdn-apple.com

Cognitive architecture (companion paper).

- Wustlich, W. (2026). *Distributed Cognitive Architecture (DCA) — Theory I: Atomic Agents · Fractal Composition · Convergence.* Zenodo. 10.5281/zenodo.20732538.

Last updated: 2026-05-25