نظرية مساعف لطبقة الذكاء العام الوكيلي

Moussaif AGI Layer Theory

طبقة توافقية حاكمة لتحويل Reasoning Models المتعددة إلى منظومة Agentic كثيفة، مقتصدة، قابلة للتكذيب

التأليف والتركيب: زهير مساعف / Zouhair Moussaif
الإصدار: v1.0 - 2026-07-14

> **الوضع الإبستمولوجي:** هذه الورقة تقدم نظرية هندسية أصلية وبرنامج اختبار. مصطلح AGI هنا أفق بحثي ومعيار تصميم، وليس شهادة بأن نظامًا حاليًا بلغ الذكاء العام.

الملخص العربي

تقترح هذه الورقة «نظرية مساعف لطبقة الذكاء العام الوكيلي» بوصفها طبقة Runtime حاكمة تقع بين Reasoning Models وبيئة الفعل، وتربط الهدف وحالة العالم والذاكرة والأدوات والمهارات والوكلاء والتقييم ضمن State Contract قابل للنقل بين Providers. لا تفترض النظرية أن نموذجًا واحدًا يجب أن يحمل كل خصائص العمومية، ولا أن جمع أكبر عدد من النماذج ينتج AGI تلقائيًا. بدلاً من ذلك، تُعرّف العمومية التشغيلية على أنها قدرة النظام على اختيار أقل تركيب كافٍ، وحفظ القيود عند الانتقال بين النماذج، وإحداث فرق موثق في القرار أو العالم، والتعلم من هذا الفرق بطريقة عكوسة.

تقدم الورقة خمس نظريات جانبية جديدة: نظرية كثافة AGI عبر النماذج M-CAD، نظرية تقارب الطبقة وانهيار التكرار M-CR، نظرية ترجمة حالات الاستدلال M-RST، نظرية الانتشار الوكيلي المتناثر M-SAP، ونظرية التعلم العكوس من الأثر العالمي M-RWEL. كما تقدم عشر أدوات هندسية، وتسعة عشر Knowledge Base مترابطة، وبروتوكول Benchmark وفلسفة نشر عبر Hermes وCodex وClaude وCursor وOllama أو أي Runtime يدعم adapters وtools وmemory.

المساهمة الأساسية هي نقل سؤال AGI من «كم هو قوي النموذج؟» إلى «كم من الفرق الموثق القابل للنقل ينتجه النظام لكل Token وكلفة وزمن ومخاطرة، وما مقدار التكرار الذي يتجنبه؟». وتُصاغ النظرية بطريقة تسمح بالتكذيب: كل مكوّن يملك تعريفًا وحدودًا ومعادلة تشغيلية وتنبؤات واختبارات فشل وقرار Promote/Quarantine/Rollback.

الكلمات المفتاحية: HyperRouting، Reasoning Models، Cross-Model Routing، Agentic AI، Moussaif AGI Layer، World-Effect Learning، Falsification، Token Efficiency، KB-as-OS، HyperLoops، HyperSkills.

English Technical Abstract #

This paper proposes the Moussaif AGI Layer Theory: a provider-neutral, evidence-governed runtime layer that coordinates reasoning models, tools, memory, skills, agents, and world-state verification. The theory rejects both single-model essentialism and naive ensemble maximalism. Generality is treated as a system property produced by sparse selection, state-preserving translation, bounded agency, falsification, and .reversible learning from observed world effects

Five new side theories are introduced: Cross-Model AGI Density (M-CAD), Layer Convergence and ,Redundancy Collapse (M-CR), Reasoning-State Translation (M-RST), Sparse Agency Propagation (M-SAP) and Reversible World-Effect Learning (M-RWEL). Ten engineering tools and nineteen knowledge-base volumes operationalize the framework. The work is positioned as a falsifiable research program, not evidence .that AGI has been achieved

1. المشكلة البحثية: لماذا نحتاج AGI Layer؟

تتطور نماذج الاستدلال بسرعة، لكن قدرتها الفردية تبقى موزعة وغير متجانسة: نموذج يتفوق في code، وآخر في long-context synthesis، وثالث في tool calling، ورابع في inference منخفض الكلفة. توصيل هذه النماذج داخل Agent framework لا يضمن العمومية؛ لأن الفشل ينتقل من ضعف النموذج إلى ضعف التنسيق: Goal Drift، فقد القيود، تضخم السياق، تكرار الوكلاء، عدم قابلية النتائج للتتبع، وتحديث ذاكرة غير موثوق.

الطبقة المقترحة ليست wrapper زخرفيًا ولا prompt ضخمًا. إنها مستوى معماري يملك State وPolicies وRegistries وMetrics. وظيفتها أن تحول الاختلافات بين النماذج إلى موارد قابلة للقياس، وأن تمنع النظام من استهلاك قدرات متشابهة دون عائد. لذلك تصبح Token Efficiency جزءًا من تعريف الذكاء التشغيلي، لا تحسينًا لاحقًا.

تختلف AGI Layer عن Mixture-of-Experts داخل الأوزان؛ فهي تعمل فوق نماذج مغلقة أو مفتوحة، ويمكنها تبديل Provider أو Local Model دون إعادة بناء النظام. كما تختلف عن Multi-Agent chat العام؛ لأنها تفرض عقودًا typed، وميزانيات، وحواجز سلطة، ومحاكم هدم، وWorld-Effect Ledger. الوحدة الأساسية ليست «رسالة وكيل» بل «فرق معرفي أو فعلي موثق».

تستند النظرية إلى إرث Moussaif Agentic AGI Operating-System Theory الذي وضع طبقات الدستور والهدف والذاكرة والمهارات والحلقات والمحاكم والأثر العالمي. الجديد هنا هو تخصيص طبقة توافق Cross-Model تقيس كثافة العمومية، وتكشف convergence والتكرار، وتترجم reasoning states، وتقرر متى يكفي runtime routing ومتى يلزم post-training.

تستفيد النظرية من أدبيات ReAct وToolformer وRAG وReflexion وMulti-Agent Debate وAgentBench وGAIA وOSWorld، لكنها لا تنسب إليها قبول البناء المساعفي. هذه الأعمال تدعم مكوّنات منفصلة: تداخل reasoning/acting، تعلم استخدام الأدوات، الذاكرة غير البارامترية، feedback النصي، التعدد الوكيلي، والتقييم التفاعلي. أما تركيبها في طبقة كثافة وتقارب وترجمة وانتشار وتعلم عكوس فهو مقترح هذه الورقة.

1.1 سؤال البحث المركزي

> كيف يمكن بناء طبقة فوق Reasoning Models متعددة تزيد القدرة العامة الموثقة لكل وحدة موارد، وتحفظ الهدف والدليل والسلطة عبر التبديل، وتتوقف عند convergence بدل تضخيم التكرار، وتحوّل الأثر الخارجي إلى تعلم عكوس؟

1.2 فرضيات الورقة

- H1: قيمة Cross-Model system ترتبط بتكامل القدرات أكثر من عدد النماذج.
- H2: Canonical Reasoning State تقلل فقد القيود أثناء handoff مقارنة بالنقل النصي الحر.
- H3: Sparse HyperRouting يحقق نسبة عالية من أداء full activation بكلفة وتنسيق أقل.
- H4: Redundancy-aware convergence rules تقلل context noise والـagent oscillation.
- H5: World-effect-gated memory promotion يحسن held-out transfer ويخفض تراكم الأخطاء مقارنة بـreflection-only.

2. التعريف الرسمي لطبقة مساعف AGI

نعرف Moussaif AGI Layer، ويرمز لها M-AGIL، بأنها طبقة نظامية تحفظ حالة موحدة وتوجه مجموعة قابلة للاستبدال من النماذج والأدوات والمهارات والوكلاء نحو هدف مقيد، ثم تتحقق من الأثر وتقرر ما الذي يستحق أن يصبح ذاكرة أو قاعدة أو تدريبًا. لا تُعرَّف الطبقة بقوة Model معين بل بعقودها وقدرتها على الاستمرار عند استبداله.

2.1 متجه الحالة

```
Λ_t = <I, G, Ω_t, K, M, S, R, A, E, B>
```

- `I`: Identity and Constitution.
- `G`: Goal and Task Contract.
- `Ω_t`: World/Project State at time t.
- `K`: retrieved Knowledge and provenance.
- `M`: Memory hierarchy.
- `S`: active HyperSkills.
- `R`: selected Models/Routes.
- `A`: authorized Actions and Tools.
- `E`: Evidence, evaluations, and court verdicts.
- `B`: budgets for tokens, cost, latency, iterations, risk.

2.2 دالة المنفعة العامة

```
U_layer = Σ_i p_i × ΔV_i × T_i × C_i - αTok - βCost - γLat - δRisk - εRed
```

حيث ΔV_i هو الفرق الموثق الذي أحدثه المسار في belief أو decision أو artifact أو world state، وT_i قابلية النقل، وC_i المعايرة، بينما تمثل الحدود السالبة استهلاك Token والكلفة والزمن والمخاطر والتكرار. المعادلة ليست قانونًا طبيعيًا بل Instrument لتجميد المقارنة ومنع التقييم الانطباعي.

2.3 المبادئ الدستورية

- لا ادعاء AGI بلا Benchmark متعدد المجالات وأثر مستقل.
- لا Model أو Tool أو KB يدخل المسار لمجرد شهرته؛ يلزم Expected Decision-Changing Value.
- لا نقل بين نماذج دون State Capsule وLoss Ledger.
- لا HyperLoop دون budgets وstop conditions وrollback.
- لا ذاكرة دائمة دون provenance وworld-effect evidence.
- لا weight update عندما تكفي RAG أو Skill أو Router patch.
- لا فعل خارجي دون Authority Gate حتى لو كانت الإجابة صحيحة.
- لا محو للتاريخ؛ كل Merge أو Distill أو Retire يحفظ genealogy.
- لا تقييم بنقطة واحدة؛ تُستخدم score vectors وبوابات دنيا.
- لا تحميل corpus ضخم في context؛ retrieval متدرج ومتناثر.

3. المعمارية الطبقية

L0 - Constitution and Identity

تعريف الهوية، حدود الادعاء، السياسات غير القابلة للتجاوز، سجل النسخ.

L1 - Task Contract and Q-MCD

تحويل الطلب إلى أسئلة حاكمة، قيود، قبول، authority، unknowns.

L2 - World and Project State

حالة زمنية، hidden state hypotheses، events، dependencies، assets.

L3 - Knowledge and Memory

provenance، episodic/canonical memory، RAG/GraphRAG، KB-as-OS.

L4 - Capability and Model Registry

vectors للنماذج والأدوات والمهارات، كلفة، توافر، failure profiles.

L5 - AGI Density and Sparse HyperRouting

اختيار أقل stack كافٍ وفق M-CAD وM-SAP.

L6 - Canonical Reasoning State

تطبيع handoffs وفق M-RST مع constraint/evidence diffs.

L7 - Bounded HyperLoops and Tool Execution

recover، verify، observe، act، planning ضمن budgets.

L8 - Courts, Safety, and Authority

anti-slop، permission، security، evidence، falsification.

L9 - World Effect and Reversible Learning

memory reconstruction، rollback، mode selection، promotion، delta ledger.

تعمل الطبقات نزولًا عند التنفيذ وصعودًا عند التعلم. ويمكن تجاوز طبقة غير لازمة فقط بقرار Routing معلل. مهمة حسابية مستقرة قد تستخدم L1 ثم Model واحد وVerifier، بينما مهمة نشر أو تعديل نظام تحتاج L0-L9 كاملة. العمومية ليست إجبار كل مهمة على أقصى تعقيد، بل معرفة متى يكون التعقيد ضروريًا.

4.1 نظرية مساعف لكثافة الذكاء العام عبر النماذج

Moussaif Cross-Model AGI Density Theory (M-CAD)

الأطروحة:

لا تُقاس قيمة طبقة AGI بعدد Models أو Agents، بل بمقدار الفرق الموثق القابل للنقل الذي تنتجه كل وحدة من Token وLatency وCost وRisk بعد طرح Redundancy.

الآلية السببية:

تُحوَّل القدرات المتباينة بين Reasoning Models إلى Capability Vectors موحّدة، ثم يُختار أقل تركيب يرفع Decision Quality أو World Effect فوق Baseline. تُمنع إضافة نموذج جديد إذا كان يكرر اتجاهًا معرفيًا قائمًا دون استقلال سببي أو تحسن قابل للقياس.

الصياغة التشغيلية:

```
D_AGI = (V × X × U × Q) / (T + C + L + R + ρ)
```

V=Verified gain، X=Cross-domain transfer، U=Reuse، Q=Calibration، T=Tokens، C=Cost، L=Latency، R=Risk، ρ=Redundancy penalty.

التنبؤات القابلة للاختبار:

- تركيب نموذجين متكاملين قد يتفوق على عشرة نماذج متشابهة عند ثبات الميزانية.
- زيادة Context بلا Routing ستخفض الكثافة حتى لو رفعت Recall الخام.
- القدرة التي لا تنتقل بين ثلاث Task Families تبقى Local Skill ولا تُرقّى إلى AGI Layer primitive.
- المسار الأعلى طلاقة قد يخسر أمام مسار أرخص إذا كان الفرق النهائي غير قابل للتحقق.

محكمة الهدم - شروط التكذيب:

- إذا لم يرتبط D_AGI بتحسن Holdout أو World Effect عبر تكرارات مستقلة.
- إذا كانت أوزان المتغيرات غير مستقرة لدرجة تجعل ترتيب المسارات عشوائيًا.
- إذا تفوق Full-ensemble دائمًا مع كلفة مماثلة على Sparse selection.

التنفيذ الهندسي:

يُبنى Capability Registry لكل Model/Tool/Skill، ويُحدّث من نتائج فعلية لا من وصف تسويقي. يسجل كل تشغيل Gain وCost وLatency وFailure وOverlap، ثم يستعمل Router تقديرًا محافظًا مع Confidence Interval.

علاقة النظرية ببقية الخماسية:

لا تعمل M-CAD منفردة. تُقرأ نتائجها مع M-RWEL، M-SAP، M-RST، M-CR. فالكثافة بلا ترجمة قد تكون وهمية، والتقارب بلا أثر عالمي قد يحذف قدرة نافعة، والانتشار المتناثر بلا محكمة قد يختار مسارًا رخيصًا لكنه خاطئ. لذا يكون قرار الطبقة متجهًا متعدد الشروط لا Score منفردًا.

تصميم تجربة أولية:

يُبنى Benchmark من ثلاث عائلات مهام: research مع web evidence، software engineering مع tests، وcomputer-use مع stateful workflow. تُقارن أربع سياسات: random sparse، full ensemble، single best model، وM-AGIL. تثبت الأدوات والصلاحيات، وتُقاس success وconstraint preservation وtokens وlatency وcost وfailures وtransfer. تُنشر النتائج السلبية وconfidence intervals.

4.2 نظرية مساعف لتقارب الطبقة وانهيار التكرار

Moussaif Layer Convergence and Redundancy-Collapse Theory (M-CR)

الأطروحة:

طبقة AGI الجيدة لا تتوسع بلا نهاية؛ بل تتوقف أو تعيد التركيب عندما يصبح العائد الحدي للحوسبة أو المعرفة أو الوكلاء أصغر من كلفة التكرار والتنسيق.

الآلية السببية:

تراقب الطبقة منحنى Marginal Verified Utility مقابل Budget. عندما ينخفض الميل تحت Threshold لثلاث دورات مستقلة، أو يتجاوز Semantic/Behavioral Overlap حدًا محددًا، تُنفذ عملية Merge أو Prune أو Distill بدل إضافة مكوّن جديد.

الصياغة التشغيلية:

```
Stop if ΔV/ΔB < τ_v OR ρ_sem × ρ_beh > τ_r
```

ΔV=المنفعة الموثقة الجديدة، ΔB=الميزانية الإضافية، ρ_sem=التشابه الدلالي، ρ_beh=تشابه السلوك، τ=حدود مضبوطة حسب المخاطر.

التنبؤات القابلة للاختبار:

- بعد نقطة معينة، زيادة عدد Agents سترفع Coordination Error أسرع من جودة الحل.
- الـKnowledge Base المكرر يرفع Retrieval Noise ويخفض Evidence Precision.
- Distillation الموجهة بالاختلاف ستحتفظ بأغلب المنفعة مع Context أصغر.
- الأنظمة التي تملك Stop Conditions صريحة ستكون أكثر استقرارًا في Long-Horizon tasks.

محكمة الهدم - شروط التكذيب:

- إذا استمر العائد الحدي موجبًا وثابتًا رغم تزايد التكرار والتنسيق.
- إذا فشل Pruning القائم على overlap في الحفاظ على الأداء أكثر من Baseline عشوائي.
- إذا كانت مؤشرات convergence لا تتنبأ بأي plateau فعلي.

التنفيذ الهندسي:

تُحسب بصمات Semantic وTool-Trace وFailure-Mode لكل مكوّن. ينفذ Redundancy Monitor مقارنة قبل/بعد ويقترح Keep/Merge/Distill/Retire، ولا يحذف أصلًا دون Genealogy وRollback.

علاقة النظرية ببقية الخماسية:

لا تعمل M-CR منفردة. تُقرأ نتائجها مع M-RWEL، M-SAP، M-RST، M-CAD. فالكثافة بلا ترجمة قد تكون وهمية، والتقارب بلا أثر عالمي قد يحذف قدرة نافعة، والانتشار المتناثر بلا محكمة قد يختار مسارًا رخيصًا لكنه خاطئ. لذا يكون قرار الطبقة متجهًا متعدد الشروط لا Score منفردًا.

تصميم تجربة أولية:

يُبنى Benchmark من ثلاث عائلات مهام: research مع software engineering، web evidence مع tests، وcomputer-use مع stateful workflow. تُقارن أربع سياسات: random sparse، full ensemble، single best model، وM-AGIL. تثبت الأدوات والصلاحيات، وتُقاس success وconstraint preservation وtokens وlatency وcost وfailures وtransfer. تُنشر النتائج السلبية وconfidence intervals.

4.3 نظرية مساعف لترجمة حالات الاستدلال

Moussaif Reasoning-State Translation Theory (M-RST)

الأطروحة:

لا يمكن بناء طبقة عامة فوق Providers متعددين ما لم تُترجم مخرجاتهم إلى State Contract مشترك يحفظ الهدف والدليل والافتراضات والقرارات والمخاطر دون الاعتماد على صياغة داخلية خاصة بنموذج واحد.

الآلية السببية:

كل Model Adapter يحوّل input/output إلى Canonical Reasoning State: Task Contract، Claim Ledger، Evidence Set، Candidate Actions، Confidence، Tool Calls، Unresolved Questions، Memory Delta. الترجمة لا تنقل hidden chain-of-thought؛ تنقل rationale موجزًا وبنية قابلة للتدقيق.

الصياغة التشغيلية:

```
F_trans = 1 - (w_c L_c + w_e L_e + w_g L_g + w_r L_r + w_a L_a)
```

L_c=فقد القيود، L_e=فقد الدليل، L_g=فقد الهدف، L_r=فقد المخاطر، L_a=فقد الأفعال؛ والأوزان حسب أهمية المهمة.

التنبؤات القابلة للاختبار:

- ارتفاع Translation Fidelity سيحسن تبديل Model منتصف المهمة دون Goal Drift.
- الـCanonical State سيقلل إعادة الشرح بين Agents ويخفض Token overhead.
- الأنظمة التي تنقل النص فقط ستفقد قرارات وقيودًا أكثر من الأنظمة ذات Schemas typed.
- يمكن اختبار Provider portability باستبدال النموذج مع تثبيت State وTools.

محكمة الهدم - شروط التكذيب:

- إذا لم يؤثر F_trans في Success أو constraint preservation.
- إذا كانت Canonical schema أكثر كلفة من إعادة التوليد دون تحسن.
- إذا تعذر تمثيل أنواع مهمة من reasoning/action دون فقد جوهري.

التنفيذ الهندسي:

تستخدم JSON Schema/Typed Dataclasses مع Versioning. يمر كل handoff عبر Constraint Diff وEvidence Diff وUnknowns Diff. تُخزن فقط المبررات الضرورية للتدقيق، لا أفكارًا داخلية مطولة.

علاقة النظرية ببقية الخماسية:

لا تعمل M-RST منفردة. تُقرأ نتائجها مع M-RWEL، M-SAP، M-CR، M-CAD. فالكثافة بلا ترجمة قد تكون وهمية، والتقارب بلا أثر عالمي قد يحذف قدرة نافعة، والانتشار المتناثر بلا محكمة قد يختار مسارًا رخيصًا لكنه خاطئ. لذا يكون قرار الطبقة متجهًا متعدد الشروط لا Score منفردًا.

تصميم تجربة أولية:

يُبنى Benchmark من ثلاث عائلات مهام: research مع software engineering، web evidence مع tests، وcomputer-use مع stateful workflow. تُقارن أربع سياسات: random sparse، full ensemble، single best model، وM-AGIL. تثبت الأدوات والصلاحيات، وتُقاس success وconstraint preservation وtokens وlatency وcost وfailures وtransfer. تُنشر النتائج السلبية وconfidence intervals.

4.4 نظرية مساعف للانتشار الوكيلي المتناثر

Moussaif Sparse Agency Propagation Theory (M-SAP)

الأطروحة:

العمومية العملية لا تتطلب تنشيط كل مهارة وذاكرة ووكيل؛ بل تتطلب قدرة دقيقة على تنشيط أصغر Subgraph كافٍ ثم توسيعه فقط عند وجود عدم يقين مسمى.

الآلية السببية:

يبدأ النظام بـMission Signature مضغوطة، يختار Primary HyperSkill و0-3 Supports، ويسترجع 3-12 Knowledge Cards. تتوسع Impact Cone حسب Information Gain وRisk وDependency، وتتوقف عند Plateau أو Budget أو حل قابل للتحقق.

الصياغة التشغيلية:

```
A_sparse = UsefulActiveNodes / (TotalActiveNodes × ContextCost × CoordinationRisk)
```

UsefulActiveNodes=العقد التي غيّرت القرار أو الفعل، TotalActiveNodes= كل ما تم تشغيله، ContextCost=تكلفة السياق، CoordinationRisk=خطر التعارض والتأخير.

التنبؤات القابلة للاختبار:

- Sparse routing سيحقق Success قريبًا من full activation بكلفة أقل بوضوح.
- تحديد uncertainty قبل التوسع سيقلل retrieval spam وtool overuse.
- الـbounded HyperLoops ستقلل oscillation وself-confirmation.
- المهام البسيطة ستستخدم Direct Path دون Multi-Agent.

محكمة الهدم - شروط التكذيب:

- إذا أدى sparse routing إلى فقد متكرر لمعارف حرجة مقارنة بـfull activation.
- إذا تعذر توقع UsefulActiveNodes أفضل من الاختيار العشوائي.
- إذا أصبحت كلفة Router أعلى من التوفير الناتج.

التنفيذ الهندسي:

يُبنى HyperGraph بعقد للمهارات والأدوات والذكريات والمقاييس. يستخدم Router scores من task fit وevidence need وcost وrisk. كل توسع يحتاج Named Gap وExpected Information Gain.

علاقة النظرية ببقية الخماسية:

لا تعمل M-SAP منفردة. تُقرأ نتائجها مع M-RWEL، M-RST، M-CR، M-CAD. فالكثافة بلا ترجمة قد تكون وهمية، والتقارب بلا أثر عالمي قد يحذف قدرة نافعة، والانتشار المتناثر بلا محكمة قد يختار مسارًا رخيصًا لكنه خاطئ. لذا يكون قرار الطبقة متجهًا متعدد الشروط لا Score منفردًا.

تصميم تجربة أولية:

يُبنى Benchmark من ثلاث عائلات مهام: research مع software engineering، web evidence مع tests، وcomputer-use مع stateful workflow. تُقارن أربع سياسات: random sparse، full ensemble، single best model، وM-AGIL. تثبت الأدوات والصلاحيات، وتُقاس success وconstraint preservation وtokens وlatency وcost وfailures وtransfer. تُنشر النتائج السلبية وconfidence intervals.

4.5 نظرية مساعف للتعلم العكوس من الأثر العالمي

Moussaif Reversible World-Effect Learning Theory (M-RWEL)

الأطروحة:

لا ينبغي أن تتحول Reflection أو نتيجة داخلية إلى Memory Canonical أو Post-Training data قبل أن ترتبط بأثر خارجي قابل للرصد، وبمصدر، وباختبار مضاد، وبمسار Rollback.

الآلية السببية:

بعد كل Action تُقارن World State المتوقعة بالملاحظة الفعلية، ثم يُستخرج Delta مع Confidence وCausal Attribution. تُرقّى الدروس عبر مراحل episodic → candidate rule → verified skill → training example، مع Holdout وSafety وForgetting checks.

الصياغة التشغيلية:

```
Promote(δ) = P_t ∧ P_h ∧ P_s ∧ P_f ∧ R_b ∧ Prov
```

P_t=نجاح target، P_h=نجاح holdout، P_s=safety، P_f=forgetting، R_b=rollback-ready، Prov=provenance complete.

التنبؤات القابلة للاختبار:

- فصل episodic memory عن canonical memory سيقلل تراكم الأخطاء.
- التعلم القائم على world effect سيحسن transfer أكثر من self-rated reflection وحدها.
- التحديثات العكوسة ستسمح بتجريب أسرع دون خسارة الاستقرار.
- اختيار RAG أو Skill قبل weight update سيخفض catastrophic forgetting.

محكمة الهدم - شروط التكذيب:

- إذا لم يتفوق world-effect promotion على reflection-only في held-out transfer.
- إذا كانت causal attribution شديدة الضوضاء بحيث لا يمكن إعادة الاختبار.

• إذا لم ينجح rollback عملياً أو بقيت آثار جانبية غير قابلة للاسترجاع.

التنفيذ الهندسي:

تسجل World-Effect Ledger الحالة قبل وبعد، الفعل، التوقع، الملاحظة، attribution، الاختبارات، القرار، والنسخة. يختار Mode Selector بين prompt patch وRAG وGraphRAG وHyperSkill وSFT/LoRA.

علاقة النظرية ببقية الخماسية:

لا تعمل M-RWEL منفردة. تُقرأ نتائجها مع M-SAP، M-RST، M-CR، M-CAD. فالكثافة بلا ترجمة قد تكون وهمية، والتقارب بلا أثر عالمي قد يحذف قدرة نافعة، والانتشار المتناثر بلا محكمة قد يختار مسارًا رخيصًا لكنه خاطئ. لذا يكون قرار الطبقة متجهًا متعدد الشروط لا Score منفردًا.

تصميم تجربة أولية:

يُبنى Benchmark من ثلاث عائلات مهام: research مع software engineering، web evidence مع tests، وcomputer-use مع stateful workflow. تُقارن أربع سياسات: random sparse، full ensemble، single best model، وM-AGIL. تثبت الأدوات والصلاحيات، وتُقاس success وconstraint preservation وtokens وlatency وcost وfailures وtransfer. تُنشر النتائج السلبية وconfidence intervals.

5.1 TL-01 - مترجم عقد طبقة AGI

Moussaif AGI Layer Contract Compiler

الوظيفة:

يحوّل طلب المستخدم إلى Task Contract typed يحفظ outcome والقيود والسلطة والميزانية ومعيار القبول.

المدخلات: طلب خام + سياق + صلاحيات

المخرجات: Mission Signature + JSON Contract + Non-goals + Acceptance tests

Failure criterion: يفشل عند إسقاط قيد صريح أو اختراع صلاحية.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.2 TL-02 - محلل كثافة القدرات عبر النماذج

Cross-Model Capability Density Profiler

الوظيفة:

يبني Capability Vectors من اختبارات فعلية ويحسب قيمة كل Provider/Model لكل Token وCost وLatency وRisk.

المدخلات: سجلات benchmark وtool traces وcost

المخرجات: Density matrix + confidence intervals + overlap map

Failure criterion: يفشل إذا اعتمد على marketing claims أو عينات غير قابلة للمقارنة.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.3 TL-03 - مراقب التقارب والتكرار

Convergence and Redundancy Monitor

الوظيفة:

يرصد marginal utility وsemantic/behavioral overlap ويصدر Keep/Merge/Distill/Retire.

المدخلات: نتائج دورات متتابعة + failure traces + embeddings

المخرجات: plateau report + redundancy graph + pruning plan

Failure criterion: يفشل إذا حذف مكونًا بلا genealogy أو rollback.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.4 TL-04 - مطبّع حالة الاستدلال

Reasoning-State Normalizer

الوظيفة:

يحوّل مخرجات النماذج إلى Canonical State من claims/evidence/decisions/actions/unknowns.

المدخلات: استجابة model + tool outputs + task contract

المخرجات: typed state capsule + loss ledger

Failure criterion: يفشل إذا ادعى نقل hidden chain-of-thought أو أخفى فقدًا دلاليًا.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهرًا للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.5 TL-05 - الموجّه الفائق المتناثر

Sparse HyperRouter

الوظيفة:

يختار أقل Model/Skill/Tool/KB stack كافٍ ويبرر الاستبعاد والتوسع.

المدخلات: mission signature + registry + budgets

المخرجات: route plan + scores + excluded routes + expansion triggers

Failure criterion: يفشل عند تفعيل stack كامل بلا named uncertainty.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.6 TL-06 - مُركّب الحلقات الفائقة المحدودة

Bounded HyperLoop Synthesizer

الوظيفة:

يحوّل الهدف إلى loop لها iteration/token/tool/cost/latency budgets وstop/rollback conditions.

المدخلات: route + task contract + verifier

المخرجات: loop graph + stop policy + recovery branches

Failure criterion: يفشل عند loop مفتوحة أو معيار توقف ذاتي غامض.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.7 TL-07 - بوابة الدليل والسلطة

Evidence-Authority Gate

الوظيفة:

تفصل صحة الادعاء عن حق التنفيذ وتمنع الأثر الخارجي غير المأذون.

المدخلات: claim ledger + permissions + risk class

المخرجات: allow/deny/review + evidence grade + required approval

Failure criterion: يفشل إذا عوضت الثقة العالية نقص الإذن أو المصدر.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.8 TL-08 - سجل فروق الأثر العالمي

World-Effect Delta Ledger

الوظيفة:

يسجل الحالة قبل/بعد والتوقع والملاحظة والسببية والنتائج السلبية.

المدخلات: action plan + telemetry + user feedback

المخرجات: auditable delta + causal confidence + memory candidate

Failure criterion: يفشل إذا سجل نجاحًا بلا observation أو تجاهل side effects.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.

- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.9 TL-09 - محدد نمط الذاكرة والتدريب اللاحق

Memory and Post-Training Mode Selector

الوظيفة:

يختار أقل طريقة تحديث تدخلاً: router patch، HyperSkill، GraphRAG، RAG، episodic memory، ephemeral state، LoRA، SFT.

المدخلات: verified delta + recurrence + transfer + rights + risk

المخرجات: mode decision + dataset spec + holdouts + rollback plan

Failure criterion: يفشل إذا اقترح weight update بينما retrieval كافٍ.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

5.10 TL-10 - محكمة Benchmark والتكذيب للطبقة

AGI Layer Benchmark and Falsification Court

الوظيفة:

تشغّل baselines وablations وadversarial tests وcost/safety/transfer evaluation ثم تصدر verdict.

المدخلات: artifact/version + benchmark suite + baselines

المخرجات: score vector + confidence + failed claims + promotion verdict

Failure criterion: يفشل إذا استخدم benchmark واحد لإعلان AGI أو أخفى negative results.

خوارزمية تشغيل مختصرة:

```
INPUT -> NORMALIZE -> SCORE -> CHECK AUTHORITY -> EXECUTE/DRY-RUN -> VERIFY -> FALSIFY ->
RECORD -> PROMOTE/ROLLBACK
```

تُسجل الأداة نسخة schema، ومصدر كل field، ودرجة الثقة، والقرار، والبدائل المستبعدة. لا يسمح لأي Tool بإخفاء عدم اليقين داخل نص طبيعي؛ يجب أن يبقى UNKNOWN أو ASSUMPTION ظاهراً للطبقات اللاحقة.

مقاييس القبول:

- Constraint preservation rate.
- Verified task success.
- Calibration error.
- Token/cost/latency delta against baseline.
- Rollback success and audit completeness.

6. بروتوكول التنفيذ الكامل

1. Parse: استخراج outcome والقيود والسلطة ومعيار القبول.
2. Question Governance: توليد 3-7 governing questions وترتيبها بـQ-MCD.
3. State Load: استرجاع 3-12 cards من KB ذات الصلة وحالة المشروع فقط.
4. Capability Profiling: قراءة نتائج فعلية حديثة لكل Model/Tool candidate.
5. Density Routing: حساب M-CAD واختيار primary route وsupports محدودة.
6. Translation Contract: إنشاء Canonical Reasoning State قبل أي handoff.
7. Bounded Planning: تجميع HyperLoop بميزانيات وstop conditions.
8. Authority and Safety: فصل صحة الخطة عن حق تنفيذها.
9. Action and Observation: تنفيذ الأداة أو dry-run مع telemetry.
10. Verification: مقارنة outcome بالacceptance tests وبالحالة المتوقعة.
11. Falsification: البحث عن rival explanation وmissing constraint وcheaper route.
12. Convergence Check: التوقف أو التقطير عند plateau/redundancy.
13. World-Effect Ledger: تسجيل before/after وside effects وcausal confidence.
14. Learning Mode Selection: episodic memory أو RAG أو Skill أو training.
15. Promotion/Rollback: بوابات holdout/safety/forgetting/provenance.

Pseudocode 6.1 مرجعي

```
def run_moussaif_agi_layer(task, state, registries, budgets):
    contract = compile_contract(task, state, budgets)
    questions = q_mcd_rank(contract)
    knowledge = sparse_retrieve(questions, registries.kb)
    candidates = profile_candidates(contract, registries.capabilities)
    route = select_by_density(candidates, contract, knowledge)
    capsule = normalize_reasoning_state(contract, knowledge, route)
    loop = synthesize_bounded_hyperloop(capsule, budgets)
    result = execute_with_authority(loop)
    verified = verify_world_effect(result, contract.acceptance_tests)
    verdict = falsification_court(verified, route, alternatives=candidates)
    decision = convergence_and_promotion(verdict, registries.memory)
    return persist_auditable_delta(decision)
```

7. تصميم قواعد المعرفة التسع عشرة

كل Knowledge Base تعامل كـretrieval-first operating volume، لا ككتلة تُحقن كاملة في Context. لكل ملف Namespace، Card IDs، metadata، primary tools، theories، hazards، gates، benchmarks، and update policy. العقد المستهدف للمستخدم هو 20,000,000 line addresses لكل KB؛ إلا أن النسخة المسلمة هنا Seed عملية مع مولد deterministic للتوسعة الدقيقة، لأن 380 مليون سطر كامل ستكون عشرات الغيغابايت ولا يمكن ادعاء تسليمها دون مادة فعلية ومساحة نقل مناسبة.

7.1 KB01 - الدستور والهوية والحدود الإبستمولوجية

Constitution, Identity, and Epistemic Boundaries

يحفظ تعريف AGI Layer، درجات الادعاء، حدود السلطة، وعدم مساواة runtime overlay بتغيير الأوزان.

- Virtual namespace: `KB01-L00000001` ... `KB01-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.2 KB02 - أطلس النظريات الخمس الجانبية

Five-Sided Theory Atlas

المتن النظري الكامل لM-CAD وM-CR وM-RST وM-SAP وM-RWEL وروابطها القابلة للاختبار.

- Virtual namespace: `KB02-L00000001` ... `KB02-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.3 KB03 - كثافة القدرات والتوجيه بين النماذج

Cross-Model Capability Density

Profiles وvectors وoverlap وcost-aware routing لنماذج الاستدلال المحلية والمدفوعة.

- Virtual namespace: `KB03-L00000001` ... `KB03-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.4 KB04 - حالة الاستدلال الموحدة والترجمة

Canonical Reasoning State and Translation

Schemas وadapters وhandoffs وloss ledgers للتبديل بين providers دون فقد الهدف.

- Virtual namespace: `KB04-L00000001` ... `KB04-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.5 KB05 - التوجيه المتناثر واقتصاد Token

Sparse HyperRouting and Token Economy

Minimal sufficient stack وcontext compaction وretrieval budgets وroute ablations.

- Virtual namespace: `KB05-L00000001` ... `KB05-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.6 KB06 - هندسة المهارات الفائقة

HyperSkills Engineering

تعريف retirement، versioning، transfer، evidence gates، outputs، inputs، activation، skill.

- Virtual namespace: `KB06-L00000001` ... `KB06-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

7.7 KB07 - الحلقات الفائقة المحدودة والتخطيط

Bounded HyperLoops and Planning

Loop synthesis، stop conditions، plateau detection، recovery، rollback، long-horizon control.

- Virtual namespace: `KB07-L00000001` ... `KB07-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

KB08 7.8 - المعرفة الفائقة والذاكرة وKB-as-OS

HyperKnowledge, Memory, and KB-as-OS

.Memory hierarchy، provenance، GraphRAG، canonical/episodic separation، retrieval cards

• Virtual namespace: `KB08-L00000001` ... `KB08-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

KB09 7.9 - الأدوات وMCP والمتصفح والكود واستخدام الحاسوب

Tools, MCP, Browser, Code, and Computer Use

.Tool contracts، sandboxing، observation، verification، GUI/browser/code execution patterns

• Virtual namespace: `KB09-L00000001` ... `KB09-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

KB10 7.10 - تنسيق الوكلاء والنماذج المتعددة

Multi-Agent and Cross-Model Orchestration

.Role allocation، diversity، debate limits، handoffs، shared state، correlated-error control

• Virtual namespace: `KB10-L00000001` ... `KB10-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

KB11 7.11 - الدليل والتحقق ومحاكم الهدم

Evidence, Verification, and Falsification Courts

.Claim ledger، source grading، defeaters، rival mechanisms، adjudication، negative results

• Virtual namespace: `KB11-L00000001` ... `KB11-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.12 KB12 - السلطة والسلامة والأمن والاسترداد

Authority, Safety, Security, and Recovery

.Least privilege، prompt injection، data rights، secrets، reversible execution، incident recovery

• Virtual namespace: `KB12-L00000001` … `KB12-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.13 KB13 - نماذج العالم وحالة المشروع

World Models and Project State

.State estimation، uncertainty، event sourcing، project maps، temporal consistency، hidden state

• Virtual namespace: `KB13-L00000001` … `KB13-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.14 KB14 - التعلم والتقطير والتدريب اللاحق

Learning, Distillation, and Post-Training

.Mode selection، dataset curation، SFT/LoRA، preference optimization، forgetting tests، adapters

• Virtual namespace: `KB14-L00000001` … `KB14-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.15 KB15 - النشر عبر بيئات التشغيل

Runtime Deployment Across Hermes, Codex, Claude, Cursor, and Ollama

.Provider-neutral adapters، registries، Windows/Hermes paths، Codex engineering، local workers

• Virtual namespace: `KB15-L00000001` … `KB15-L20000000`.

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.16 KB16 - المراقبة والكلفة والزمن واقتصاد RPM

Observability, Cost, Latency, and RPM Economy

.Tracing، token accounting، quotas، budget policies، cache، model fallback، SLOs

.KB16-L20000000 ... KB16-L00000001 :Virtual namespace •

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.17 KB17 - المعايير والمحاكاة والتقييم

Benchmarks, Simulations, and Evaluation

.GAIA/AgentBench/OSWorld-style tasks، custom layer benchmarks، ablation، confidence، reproducibility

.KB17-L20000000 ... KB17-L00000001 :Virtual namespace •

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

7.18 KB18 - مناهج البحث والبيانات وبروتوكولات التجربة

Research Methods, Datasets, and Experiment Protocols

.Hypotheses، preregistration، data cards، contamination checks، replication، statistical analysis

.KB18-L20000000 ... KB18-L00000001 :Virtual namespace •

.Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar •

.Retrieval policy: 3-12 cards first; expand only for a named uncertainty •

.Promotion policy: no card becomes canonical without provenance, verifier, and version history •

KB19 7.19 - الموجّه الرئيسي وسجل التكامل والتطور

Master Router, Integration, and Evolution Registry

يربط 18 migration، dependency graph، skill/loop registries، 5 theories، 10 tools، KB وversioning.

- Virtual namespace: `KB19-L00000001` ... `KB19-L20000000`.
- Physical seed: operational cards, schemas, tools, loops, benchmarks, and generation grammar.
- Retrieval policy: 3-12 cards first; expand only for a named uncertainty.
- Promotion policy: no card becomes canonical without provenance, verifier, and version history.

8. Benchmark Framework

لا يختبر M-AGIL بالأسئلة النصية فقط. يجب أن تغطي suite ست طبقات: constraint preservation، answer correctness، cross-model handoff، long-horizon state، tool execution، وlearning transfer. وتُستخدم بنية شبيهة بفلسفة GAIA وAgentBench وOSWorld، مع مهام خاصة بقياس الترجمة والكثافة والتقارب.

Metric	التعريف
Task Success	نسبة المهام التي اجتازت verifier خارجيًا.
Constraint Preservation	نسبة القيود الصريحة التي بقيت صحيحة حتى النهاية.
Density Gain	التحسن الموثق لكل Token/Cost/Latency/Risk.
Translation Fidelity	حفظ claims/evidence/goals/risks/actions خلال handoff.
Sparse Efficiency	أداء sparse route كنسبة من full activation مع كلفة أقل.
Redundancy Ratio	نسبة المكونات التي لم تغير القرار أو الأثر.
World-Effect Validity	تطابق التوقع مع الملاحظة الخارجية.
Rollback Success	استرجاع الحالة دون residue غير مأذون.
Held-out Transfer	تحسن على عائلات مهام لم تدخل في tuning.
Calibration	مطابقة confidence مع observed success.

8.1 Baselines وAblations

- Single best model without layer
- Full ensemble with all models/agents
- Random sparse routing
- Rules-only router
- M-AGIL without M-RST
- M-AGIL without convergence monitor
- M-AGIL without world-effect gating
- Full M-AGIL

8.2 بوابات النجاح

- لا تُقبل زيادة success إذا نتج عنها خرق Authority أو Safety.
- لا تُقبل كفاءة أعلى إذا هبط constraint preservation تحت الحد الأدنى.
- لا تُقبل ذاكرة جديدة إذا فشلت provenance أو rollback.
- لا يُعلن transfer من عائلة مهام واحدة.
- لا يُقارن نموذجان بميزانيات أو أدوات أو محاولات مختلفة دون تطبيع.

9. الأمن والحاكمية

تتعامل الطبقة مع prompt injection باعتباره تعارض سلطة لا مجرد نص ضار. كل datum مسترجع يحمل مستوى trust ولا يستطيع تعديل Constitution أو permissions. تعليمات المصادر الخارجية تُقرأ كمحتوى ما لم تكن موقعة ومأذونة ضمن control plane.

تستخدم الأدوات least privilege وshort-lived credentials وsandbox وdry-run. تُفصل read actions عن write actions، وتُطلب موافقة صريحة للأفعال غير العكوسة أو المالية أو النشر العام أو الاتصال الخارجي. نجاح reasoning لا يمنح حق التنفيذ.

يجب ألا تحفظ الذاكرة أسرارًا أو بيانات شخصية أو copyrighted corpora بلا حق. يقوم Memory Selector بتصنيف ما يجوز أن يكون ephemeral أو encrypted reference أو canonical rule. لا تُستخدم ملفات Markdown كادعاء أنها أعادت تدريب نموذج مغلق.

تُراقب الأخطاء المترابطة بين Agents. التعدد لا يعني الاستقلال؛ قد تكرر النماذج نفس bias أو training data. لذلك يقيس Capability Profiler diversity في traces وfailure modes، لا اختلاف الأسلوب اللغوي فقط.

10. اقتصاد Token وRPM

- Boot kernel صغير: Constitution + registries + mission signature، لا تحميل 19 KB.
- Primary HyperSkill واحدة و0-3 supports في المسار الافتراضي.
- 3-12 knowledge cards لكل retrieval wave.
- State Capsule من 300-900 tokens بين الوكلاء، مع hashes للمصادر الكاملة.
- Escalation إلى نموذج أعلى فقط عند uncertainty أو verifier failure أو risk threshold.
- Cache للنتائج الثابتة مع freshness policy، وعدم تكرار tool call بلا state change.
- Stop عند plateau لثلاث دورات أو عند استنفاد budget أو تحقق acceptance tests.
- تقدير Expected Value قبل كل Agent أو Tool إضافي.

11. النشر عبر Runtime متعدد

في Hermes تعمل الطبقة كـPersistent Orchestrator يحفظ registries وschedules وprovider quotas وmemory وtraces. في Codex تتحول القرارات إلى repository-native diffs وtests وmigrations. في Claude أو نماذج long-context تُستخدم synthesis وconstitutional review. في Ollama/local models تُسند classification وextraction وlow-risk generation عندما تكون كافية. هذه priors وليست احتكارات ثابتة.

كل Provider Adapter يطبق M-RST schema نفسها، ويعلن capabilities وlimits وpricing snapshot وtool permissions. عند تبديل النموذج، لا يُعاد بناء Task من الصفر؛ تُرسل State Capsule وretrieved evidence pointers وopen questions. ويُقاس فقد الترجمة قبل اعتماد adapter.

النشر يكون مرحليًا: bounded production، canary، dry-run، shadow، ثم promotion. لكل مرحلة acceptance gates وrollback pointer. لا تُرقى تغييرات governance أو memory constitution تلقائيًا من asset جديد.

12. برنامج البحث التجريبي

Experiment	التصميم
E1 Density	مقارنة single/full/random/M-AGIL عبر 300 مهمة من ثلاث عائلات وميزانيات ثابتة.
E2 Translation	تبديل Provider في منتصف المهمة وقياس حفظ القيود والدليل والقرارات.
E3 Convergence	زيادة عدد Agents/KB cards تدريجيًا ورصد plateau والتكرار.
E4 Sparse Propagation	ablation لعدد active nodes وretrieval waves وتأثيرها في success/cost.
E5 World-Effect Learning	مقارنة reflection-only وepisodic وworld-effect-gated promotion على held-out tasks.
E6 Security	حقن تعليمات خبيثة داخل documents/tools وقياس authority preservation.
E7 Post-Training Mode	مقارنة prompt/RAG/skill/router/SFT/LoRA لنفس السلوك من حيث transfer وforgetting.
E8 Long Horizon	مهام 100-500 خطوة مع state changes وmid-task messages وhidden dependencies.

12.1 متطلبات إعادة الإنتاج

- نشر task manifests وtool permissions وbudgets وmodel versions.
- تجميد prompts/schemas والاحتفاظ بالhashes.
- فصل development وvalidation وheld-out tests.
- تسجيل كل tool call وstate transition وverifier output.
- الإبلاغ عن failures وtimeouts وhuman interventions.
- استخدام confidence intervals وعدم الاكتفاء بمتوسط واحد.
- إعادة التجربة عبر Providerين على الأقل وRuntimeين عند اختبار portability.

13. حدود النظرية

- قد تكون معادلات الكثافة حساسة للأوزان، وتحتاج calibration حسب المجال والمخاطر.
- Canonical State قد لا يمثل كل أشكال cognition أو multimodal latent state.
- World-effect attribution صعب في البيئات ذات العوامل المتداخلة أو feedback المتأخر.
- Sparse routing قد يفقد rare capability إن كان registry ناقصًا أو benchmark منحازًا.
- Multi-model diversity قد تكون أقل من المتوقع بسبب تشابه بيانات التدريب.
- الكفاءة لا تعني الذكاء العام؛ قد ينجح النظام في suite محدودة فقط.
- الطبقة لا تخلق قدرات أساسية غير موجودة في أي model/tool/data متاح.
- Post-training يحتاج checkpoints وحقوق بيانات وcompute وevaluation؛ Markdown وحده لا ينفذه.
- الأسماء المساعفية تصف تركيبًا نظريًا مقترحًا ولا تعني معيارًا علميًا معتمدًا.
- لا بد من مراجعة بشرية مستقلة في المجالات عالية المخاطر.

14. الخاتمة

تعيد نظرية مساعف لطبقة AGI تعريف الانتقال نحو الأنظمة العامة باعتباره مشكلة تنسيق كثيف ومقتصد وقابل للتكذيب. لا يكون التقدم بإضافة Model أو Agent أو KB جديد تلقائيًا، بل بزيادة الفرق الموثق القابل للنقل لكل وحدة موارد، مع حفظ الهدف والدليل والسلطة، والتوقف عند convergence، وترجمة الحالة عبر providers، وتنشيط أقل subgraph كافٍ، وتحويل الأثر العالمي إلى تعلم عكوس. الخماسية الجديدة والعشر أدوات والتسع عشرة KB تشكل برنامجًا هندسيًا يمكن تنفيذه تدريجيًا وقياسه وهدمه وإعادة بنائه. النجاح الحقيقي ليس أن تسمى الطبقة AGI، بل أن تثبت في اختبارات مستقلة أنها أكثر قدرة وأقل تكرارًا وأوضح أثرًا وأسهل استردادًا من البدائل الأبسط.

المراجع وخريطة الإسناد

- **Yao et al [R1].** *ReAct: Synergizing Reasoning and Acting in Language Models*. arXiv:2210.03629. يدعم فصل/تداخل reasoning وacting؛ لا يثبت نظرية مساعف.
- **Schick et al [R2].** *Toolformer: Language Models Can Teach Themselves to Use Tools*. arXiv:2302.04761. يدعم تعلم اختيار الأدوات ودمج نتائجها.
- **Lewis et al [R3].** *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. arXiv:2005.11401. يدعم الجمع بين parametric وnon-parametric memory.
- **Shinn et al [R4].** *Reflexion: Language Agents with Verbal Reinforcement Learning*. arXiv:2303.11366. يدعم التعلم النصي من feedback دون تحديث weights.
- **Du et al [R5].** *Improving Factuality and Reasoning in Language Models through Multiagent Debate*. arXiv:2305.14325. يدعم إمكان فائدة التعدد مع ضرورة اختبار التكرار والأخطاء المترابطة.
- **Liu et al [R6].** *AgentBench: Evaluating LLMs as Agents*. arXiv:2308.03688. يدعم التقييم متعدد البيئات لوكلاء LLM.
- **Mialon et al [R7].** *GAIA: a benchmark for General AI Assistants*. arXiv:2311.12983. يدعم مهام عامة تجمع reasoning وweb وtools وmultimodality.
- **Xie et al [R8].** *OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments*. arXiv:2404.07972. يدعم execution-based evaluation في بيئات الحاسوب الحقيقية.
- **Yuan et al [R9].** *OSWorld 2.0: Benchmarking Computer Use Agents on Long-Horizon Real-World Tasks*. arXiv:2606.29537. يدعم الحاجة إلى long-horizon constraint tracking وmid-task state updates.
- **Moussaif, Zouhair [R10].** *Moussaif Agentic AGI Operating-System Theory, v*1.0. 2026-07-13. السلف النظري الداخلي الذي تُبنى فوقه AGI Layer دون ادعاء تحقق AGI.

ملحق A - Schema حالة الاستدلال الموحدة

```
{
  "schema_version": "1.0",
  "task_contract": {
    "objective": "",
    "constraints": [],
    "non_goals": [],
    "authority": [],
    "acceptance_tests": []
  },
  "claims": [
    {
      "id": "C-001",
      "type": "FACT|INFERENCE|HYPOTHESIS|PROPOSAL|UNKNOWN",
      "text": "",
      "confidence": 0.0,
      "sources": []
    }
  ],
  "evidence": [
    {
      "id": "E-001",
      "source_hash": "",
      "grade": "E0-E5",
      "supports": [],
      "contradicts": []
    }
  ],
  "decisions": [
    {
      "id": "D-001",
      "choice": "",
      "alternatives": [],
      "reason": "",
      "reversible": true
    }
  ],
  "actions": [
    {
      "id": "A-001",
      "tool": "",
      "permission": "read|write|external",
      "status": "planned"
    }
  ],
  "unknowns": [],
  "risks": [],
```

```
  "budgets": {
    "tokens": 0,
    "cost": 0,
    "latency_s": 0,
    "iterations": 0
  },
  "memory_delta": {
    "candidate": false,
    "mode": "none",
    "provenance": [],
    "rollback": ""
  }
}
```

ملحق B - بطاقة Knowledge Card معيارية

```
id: KBxx-C000001
topic: ""
task_phase: ""
governing_question: ""
mechanism: ""
inputs: []
outputs: []
evidence_requirements: []
tools: []
theories: []
hazards: []
gates: []
metrics: []
falsifiers: []
rollback: ""
provenance: []
version: "1.0"
```

ملحق C - محكمة الهدم ذات 26 نقطة

1. تعريف الادعاء بدقة
2. حدوده ومجاله
3. Baseline الأبسط
4. مصدر كل fact
5. فصل inference عن fact
6. التلوث والـleakage
7. البدائل السببية
8. حساسية الأوزان
9. حفظ القيود
10. حفظ السلطة
11. سلامة الأدوات
12. مخاطر الخصوصية
13. حقوق البيانات
14. التكرار الدلالي
15. التكرار السلوكي
16. كلفة التنسيق
17. Token waste
18. Latency and RPM
19. فشل rollback
20. فشل verifier
21. التحيز بين providers
22. negative transfer
23. catastrophic forgetting
24. world-effect mismatch
25. قابلية إعادة الإنتاج
26. قرار promote/quarantine/retire

ملحق D - مواصفات الحزمة ذات 19 KB

تتضمن الحزمة Seed files فعلية ومولدًا deterministic يمكنه إنشاء 20,000,000 سطر لكل KB أو تقسيمها إلى shards. لا تدعي النسخة المضغوطة أنها تحتوي فعليًا على 380,000,000 سطر معرفي؛ بل تقدم مادة تشغيلية، grammar ،line-address contract، وgenerator. المادة الكاملة عند التوليد تحتاج مساحة تخزين كبيرة جدًا، ويجب تقييم جدوى 20M lines مقابل Graph/Parquet/SQLite لأن line count ليس مقياسًا للجودة.