AGENTIC THINKING AGI — CANONICAL CLOUD HYPERKNOWLEDGE MEMORY OS

ذاكرة مشروع تشغيلية سحابية كثيفة — MOUSSAIF Unified AGI / MOUSSAIF9B / M-TAOS / MCD-QMCD-MCAI / HyperKnowledge OS

0. هوية الوثيقة ووظيفتها

هذه الوثيقة هي نقطة الدخول الكانونية لمشروع AGENTIC THINKING AGI داخل أرشيف Google Drive. وظيفتها ليست إعادة سرد المحادثات ولا ادعاء أن كل ما طُلب قد أُنتج، بل إعادة بناء الحالة المعرفية والهندسية الأحدث للمشروع وربطها بالأصول الفعلية الموجودة في Drive. يجب أن تُقرأ بوصفها طبقة تشغيل Persistent Project Memory OS: تعرّف الهوية، النظريات، المعمارية، الاصطلاحات، الأصول، العلاقات، حدود الدليل، مسارات الاسترجاع، وبرنامج الاستمرار. أي وكيل جديد يجب أن يبدأ منها قبل فتح الملفات الضخمة.

الوسوم الإبستيمية المعتمدة داخل هذه الحزمة هي: VERIFIED_DRIVE لما تم رصده فعلياً في Drive؛ CHAT_CONTEXT لما هو مستخلص من سياق هذا المشروع ومحادثاته المتاحة؛ USER_DIRECTIVE لما صرّح به المستخدم كقاعدة أو هدف أو تفضيل؛ PROPOSED_THEORY لما هو نظرية أو قانون أو آلية طورها المشروع ولم تُثبت تجريبياً بعد؛
REQUESTED_TARGET لما كان طلباً للحجم أو العدد أو القدرة؛ REFERENCED_BUT_NOT_FOUND لما ذُكر كمخرج أو أصل ولم يتم التحقق من وجوده؛ SUPERSEDED لما تجاوزه إصدار أو تسمية أحدث؛ CANONICAL لما توجد قرائن صريحة على اعتماده الحالي.

قاعدة الحوكمة: لا يجوز تحويل REQUESTED_TARGET إلى VERIFIED_DRIVE، ولا
PROPOSED_THEORY إلى established scientific result، ولا تلخيصاً مولداً إلى مصدر أصلي. عند التعارض تتقدم: تصحيحات المستخدم الحديثة، ثم التصريحات الكانونية، ثم الأصول التشغيلية الحديثة، ثم مواصفات المشروع المباشرة، ثم التوليفات

المولدة، ثم الذاكرة التاريخية، ثم الاستنتاج.

1. تعريف المشروع الأعلى

AGENTIC THINKING AGI هو مظلة بحثية وهندسية تهدف إلى بناء ذكاء وكيلي موجّه نحو AGI لا يعتمد على نموذج منفرد، بل على نظام تشغيل معرفي-وكيلي يجمع: دستوراً للحاكمية، سؤالاً حاكماً، استخراج الفروق عالية القيمة، توجيه الحوسبة، Bounded HyperLoops، HyperSkills، HyperKnowledge KB-as-OS، وكلاء متعددين، أدوات RAG/GraphRAG، محاكم هدم وتفنيد، فعل عالمي مصرح به، قياس أثر، وذاكرة إعادة بناء. الاسم الكانوني الأوسع للنواة المعمارية في الذاكرة الحالية هو MOUSSAIF9B، مع بقاء Moussaif1B اسماً تاريخياً أو سابقاً.

الفرضية المعمارية المركزية للمشروع هي أن خصائص الوكالة العامة لا تُختزل في parameter count أو جودة LLM منفرد، بل يمكن رفع الكثافة الوكيلية من خلال تركيب منظم بين المعرفة، الذاكرة، الأدوات، الاسترجاع، التخطيط، التنفيذ، والتحقق. لذلك يستخدم المشروع كلمة OS بمعنى طبقة تنظيمية تنفيذية فوق النماذج، وليس كنظام تشغيل حاسوبي تقليدي فقط.

المشروع يتعامل مع المعرفة كـ executable cognitive substrate. وثيقة أو KB كبيرة لا تُعد نافعة لمجرد الحجم؛ يجب أن تكون قابلة للاسترجاع، مرتبطة بمهارات، ذات provenance، قابلة للاختبار، وتملك قواعد تشغيل وفشل وتحديث. هذا هو معنى HyperKnowledge OS في هذا السياق.

2. الحالة الكانونية المختصرة

الحالة الحالية يمكن ضغطها في سلسلة تشغيل واحدة: Constitution → Governing Question → Active Difference → MCD/QMCD → Vascular Compute Routing → Knowledge/Skills Activation → Bounded Agentic Execution → Falsification/Destruction Courts → Authorized Tool Action → World Effect → Evidence Capture → Self-Reconstruction Memory → Higher-Order Question.

M-TAOS هو طبقة تشغيل المهام الأحدث التي ظهرت في الأصول المتحققة داخل Drive. وهو يحول مهمة واحدة أو أسطول مهام إلى عقدة تشغيل تحتوي الهدف، القيود، اختبارات القبول، World Delta، مخطط اعتماديات، ميزانية حوسبة، مسارات متنافسة، أدوات، تحققاً غير متناظر، كتابة فورية للنتائج، وذاكرة تشغيل بين الدورات. هذا يجعل Task OS امتداداً تنفيذياً للنواة النظرية وليس مشروعاً منفصلاً عنها.

ParallaxWorld-MCD / Vision Neural World OS يمثل فرع world-model-oriented داخل المظلة. وظيفته الأساسية هي بناء تمثيل متعدد الزوايا للحالة والعالم بحيث لا تُعامل الاستجابة النصية كالعالم نفسه، بل كفرضية ضمن نموذج حالة يمكن مقارنته بالملاحظة والأثر.

Sovereign Continuity Memory OS يمثل طبقة الاستمرارية: حفظ الأسماء الكانونية، القواعد، الأصول، النسخ،

manifests، hashes حيث توفرت، والفصل بين المنجز والمرغوب. هذه الطبقة ليست مجرد backup؛ إنها contract لاستعادة المشروع بعد فقد السياق.

3. النواة النظرية: MCD / Q-MCD / MCAI / HAKIM

MCD هو حساب للفرق ذي القيمة. الصيغة المتكررة في المشروع: MCD = λ1D + λ2I + λ3T + λ4A + λ5P + λ6C − λ7N. لا ينبغي التعامل معها كمعادلة فيزيائية مثبتة، بل كـ scoring framework بحثي لترتيب الفروق بحسب المعلومات، التحول، الفعل، الإمكان، السياق، والضجيج. قوة النظرية لا تأتي من الرموز بل من قدرتها على إجبار النظام على سؤال: ما الذي تغيّر فعلاً؟ وما الذي يغير قراراً أو مساراً؟

Q-MCD يضيف السؤال الحاكم بوصفه مضاعفاً للفضاء الاستدلالي: MCD_Q = Ω(Q) × MCD، مع صيغة عمق سؤال تاريخية (Q_depth = (q_d + q_i + q_t + q_a + q_p + q_c) / (1 + q_n). الفكرة التشغيلية هي أن سؤالاً سطحياً قد يحصر نظاماً قوياً داخل فضاء بحث ضعيف، بينما إعادة صياغة السؤال الحاكم يمكن أن تعيد توجيه الاسترجاع والحوسبة والوكلاء.

MCAI/MICAI في هذا البرنامج يعبّر عن طبقة تركيب معرفي ووكيلي تتجاوز الإجابة أحادية المسار. يستخدم التعدد لا لزيادة عدد النصوص، بل لإنتاج اختلافات مفيدة بين plans، decompositions، hypotheses وcritics. المطلوب هو useful diversity لا agent theater.

HAKIM-MCD هو pipeline تشغيلي متكرر: Vascular Compute، MCD Scoring، Difference Extraction، World-Effect Routing، Agency Selection، Tool/GraphRAG، Falsification Court، Candidate Paths، Self-Reconstruction Memory، Loop. هذا التسلسل هو أهم bridge بين النظرية والوكالة التنفيذية.

4. قانون السؤال الحاكمي والحاكمية العليا

المشروع يفترض أن نظاماً عاماً يحتاج إلى ترتيب للحاكمية قبل البحث. السؤال الحاكم لا يساوي سؤال المستخدم حرفياً؛ هو التوتر الذي إذا حُلّ غيّر بقية المسألة. لذلك توجد طبقة أعلى تحدد أعلى اعتبارية، ثم أعلى فرق حاكم، ثم تسمح بتفريع فروق ثانوية. هذا يمنع تساوي كل التفاصيل ويعطي compute routing أساساً منطقياً.

الحاكمية ليست مجرد alignment policy. هي ترتيب أولوية بين الدستور، صلاحيات الفعل، هدف المهمة، الحقيقة المتاحة، المخاطر، الميزانية، والنتيجة المرغوبة. أي optimization يخالف طبقة أعلى يجب أن يتوقف أو يُصعّد. ومن هنا تظهر Constitutional Reversibility وAuthorization وRollback كأجزاء معمارية، لا إضافات أمنية متأخرة.

5. الوعائية الحاسوبية Vascular Compute Routing

نظرية الوعائية الحاسوبية ترى أن الحوسبة مورد يجب أن يتدفق نحو الفجوات ذات أعلى قيمة هامشية، لا أن يُوزع بالتساوي على كل الفرعيات. عقدة غامضة عالية المركزية تحصل على مزيد من البحث، النماذج، الأدوات والتفنيد؛ عقدة مكررة منخفضة الإضافة تُغلق مبكراً. هذا يعطي المشروع معياراً ضد التضخم الحسابي وضد الاعتقاد أن المزيد من reasoning tokens أفضل دائماً.

التوجيه الوعائي يستخدم إشارات مثل: project centrality، information novelty، dependency count، uncertainty، contradiction، expected world effect، cost، latency، tool availability، provenance quality، والقدرة على التحقق. في M-TAOS تظهر هذه الفكرة كـ Marginal Compute Value وComputational Action Density وParallelism Boundary.

عندما تعمل عدة نماذج أو وكلاء، المطلوب ليس التصويت البسيط. يمكن توجيه model A لتوليد hypothesis، model B لتدميرها، tool C لجلب evidence، ثم merger D لبناء حل يحفظ التناقضات غير المحسومة. هذا هو معنى QMCD-style competing-model evaluation في الطبقة التنفيذية.

6. HyperKnowledge KB-as-OS

HyperKnowledge ليس corpus ضخماً فقط. الوحدة الناضجة يجب أن تحمل: Identity، Scope، Trigger، Inputs، Outputs، Ontology، Claims، Evidence، Mechanism، Retrieval Keys، Dependencies، Skills، Version، Memory Rule، Failure Modes، Courts وProvenance. كلما تحولت المعرفة إلى هذه البنية زادت قابلية الوكيل لتشغيلها بدلاً من إعادة قراءتها كنص خام.

المشروع يحافظ على تصور Domain → Subdomain → Concept → KB → Skill → Agent → Workflow. هذه السلسلة تمنع عزل KB عن التنفيذ. المعرفة التي لا ترتبط بمهارة أو قرار أو اختبار تبقى مادة مرجعية، بينما المعرفة ذات العقود التشغيلية تصبح جزءاً من OS.

لـRAG يجب الحفاظ على chunks ذات IDs مستقرة، عناوين واضحة، provenance، رتبة دليل، version، project، relationships، boundary، وGraphRAG. يستخدم edges مثل DEPENDS_ON، SUPERSEDES، IMPLEMENTS، DERIVED_FROM، CONTRADICTS، EXTENDS وGENERATES. الاسترجاع القائم على similarity وحده غير كافٍ لأن المشروع يضع authority وfreshness وcanonicality ضمن قرار الاستدعاء. conflict metadata.

7. HyperSkills

HyperSkill هي قدرة قابلة لإعادة الاستخدام والتقييم، وليست اسماً طموحاً. العقد القياسي المقترح: skill_id، purpose، applicability، inputs، outputs، preconditions، procedure، tools، evidence requirements، validation،

.status و failure modes، resource budget، permissions، stop conditions، related skills، version

M-TAOS يثبت عائلة تشغيلية واضحة من المهارات: Kernel Extraction، Goal-to-World-Delta Compilation، Dependency Graphing، Adaptive Compute Routing، Evidence Packet Engineering، Countermodel Generation، Branch Diversity Control، Falsification Routing، Tool Capability Mapping، Episodic Memory Distillation، Semantic Memory Synthesis، Procedural Skill Extraction، Conflict-Preserving Merge، Compute Saturation Detection، Execution-Proof Design، Canonical Ledger Engineering، Run-Delta Compression، Versioned Kernel Evolution، Permission-Aware Agency، Failure-Memory Encoding، Benchmark Construction، MCD Difference Scoring، QMCD Model Superposition، World-Effect Accounting، Output Surface Compilation و Destruction-Court Operation.

قاعدة هذا الأرشيف: الأعداد التاريخية مثل 1400 أو 4500 أو HyperSkills 6000 لا تُعامل كأصول متحققة إلا إذا أمكن إحصاؤها من الملفات. ما لم يُتحقق منه يبقى REPORTED أو REQUESTED_TARGET. هذا يمنع تضخيم kernel بمؤشرات اسمية.

8. Bounded HyperLoops

HyperLoop هو حلقة تشغيل ذات حالة وميزانية وإيقاف، وليس recursion مفتوحاً. العقد المقترح يشمل goal، iteration cap، time/token/tool budget، progress metric، stagnation detector، contradiction detector، verification checkpoint، external-action permission، stop condition، escalation rule، rollback و final evidence .log

الحلقات المتكررة في M-TAOS تشمل Kernel Extraction، Evidence Ranking، Concept Disambiguation، Problem Decomposition، Hypothesis Generation، Destruction-and-Repair، Tool Construction، Skill Training، Retrieval Optimization، Agentic Execution، Update/Versioning، Research Program، Compute Routing، Parallel Branch، Branch-Falsify-Merge، World Effect، Memory Consolidation، Failure Reconstruction، Cross-Task Diffusion، Opportunity Conversion، Saturation، Hallucination Suppression و HyperFrontier.

مبدأ boundedness مهم لأن المشروع يستهدف agentic autonomy عالية، لكن بدون حد يصبح النظام معرضاً للتكرار، تضخم التكلفة، تثبيت الهلوسة، وإجراءات غير مصرح بها. لذلك كل loop يجب أن يمتلك دليلاً على التقدم أو يغلق.

9. Agentic Layer

الوكلاء في هذا المشروع أدوار وظيفية متعامدة أكثر من كونهم شخصيات. أدوار النواة: Governor، Question Architect، Difference Extractor، Researcher، Planner، Tool Operator، Countermodel/Falsifier، Evidence Auditor، Merger، Memory Curator، World-State Observer وAuthorization Gate. يمكن دمج عدة أدوار داخل نموذج واحد أو توزيعها حسب حجم المهمة.

معيار نجاح multi-agent ليس عدد الوكلاء بل انخفاض failure probability وارتفاع verified world effect بالنسبة للتكلفة. إذا كانت الأدوار متشابهة أو تكرر نفس السياق فهي overhead. إذا كانت تنتج مسارات مستقلة قابلة للتفنيد مع تجميع يحفظ الخلاف فهي مفيدة.

الـMain Brain هو router وحاكم سياق أكثر من كونه نموذجاً واحداً. يختار local/frontier/API/browser/coder/researcher بحسب الحاجة، ثم يعيد النتائج إلى state موحد. يجب أن يحتفظ المصدر، confidence، model/tool provenance وdate/freshness مع كل نتيجة مهمة.

10. Tool Layer وWorld Effect

الأدوات الأساسية المعرّفة في M-TAOS تشمل Concept Kernelizer، Evidence Ranker، Difference Extractor، MCD Evaluator، QMCD Superposition Map، Task DAG Compiler، Vascular Compute Router، Parallel Dispatch Planner، Branch Diversity Meter، Destruction Court Matrix، HyperSkill Distiller، HyperLoop Orchestrator، RAG Chunking Compiler، GraphRAG Relation Compiler، Memory Write Filter، Failure Signature Detector، Execution Proof Harness، Run Delta Compiler، HyperFrontier Ranker، AGI-Readiness Index، Prompt/Kernel Regression Tester وCompute Ledger.

World Effect يعني أن مهمة تنفيذية لا تُغلق عند إنتاج نص، بل عند وجود state transition يمكن ملاحظته: ملف كُتب وتم التحقق منه، رسالة أُرسلت عند التصريح، deployment تحقق، dataset تغير، قرار تم تسجيله، أو تجربة أنتجت قياساً. المهام البحثية غير التنفيذية تستخدم evidence artifact بدلاً من world mutation.

بعد الفعل يجب إجراء observation جديد وعدم افتراض النجاح من tool call وحدها. الحلقة هي Intent → Authorized → Action → Observation → Verification → State Update. هذا المبدأ مهم جداً في Drive consolidation وفي أي task automation لاحق.

11. محاكم الهدم والفجوات

المشروع يستخدم المحكمة كـ adversarial operator لا كقسم بلاغي. محكمة الهلوسة تسأل ما الذي لا يملك مصدره؛ محكمة الفجوة تبحث عن الانتقالات غير المبررة؛ محكمة الاختزال تبحث عن المعلومات التي ضاعت بالضغط؛ محكمة الأصالة الزائفة تفصل إعادة التسمية عن الإضافة؛ محكمة التشغيل تسأل هل توجد آلية قابلة للتنفيذ؛ محكمة التكرار تقيس novelty؛ محكمة الذاكرة تفحص ترسيخ الخطأ؛ محكمة الأثر تفحص ما إذا كان الإنجاز تغيراً حقيقياً أم output theater.

قاعدة falsification: كل نظرية أو خطة قوية يجب أن تصف ما الذي سيجعلها خاطئة، وما counterexample الأقوى، وما measurement المطلوب، وهل يمكن إصلاحها دون تغيير معناها. إذا لم يكن هناك failure condition، تبقى الفكرة غير قابلة للتقييم.

12. Self-Reconstruction Memory

الذاكرة ليست transcript كامل. الطبقات المقترحة: Working Memory للمهمة الحالية؛ Episodic Memory لما حدث في تشغيلات محددة؛ Semantic Memory للحقائق والمفاهيم المستقرة؛ Procedural Memory للمهارات؛ Project Registry للحالة الكانونية؛ Archive للتاريخ الخام. يجب أن تختلف سياسات retention والكتابة بين الطبقات.

Self-Reconstruction Memory تحفظ delta التشغيل: ما الذي كنا نعتقده، ما الذي جُرّب، ما الدليل الجديد، ما الذي تغير، لماذا تغير، وما النسخة التي تم تجاوزها. هذا يسمح بإعادة بناء reasoning state دون حفظ chain-of-thought الخاص، ويحول الذاكرة إلى سجل قرار ودليل قابل للتدقيق.

قانون Immediate Writeback Compounding في M-TAOS يلتقط فكرة أن الأثر يتضاعف عندما تُكتب النتائج الموثقة فوراً إلى الذاكرة القابلة للاسترجاع. لكن writeback يجب أن يمر عبر Memory Write Filter كي لا تُرسّخ conjectures كحقائق.

13. M-TAOS: Task OS التنفيذي

M-TAOS v1.0 هو النظام التنفيذي الحالي للمهام عالية الكثافة. بروتوكول التشغيل المتحقق في الوثيقة الكانونية: State Load → World Delta → Graph Compile → Compute Allocation → Discovery/Execution → Verification → Branch-Falsify-Merge → World Effect → Commit → Saturation Court. عند أسطول مهام تضاف إدارة تعارض الموارد، deduplication، shared knowledge وcross-task diffusion.

النظريات التشغيلية المعلنة داخله تشمل Marginal Compute Value Law، Computational Action Density، Vascular Compute Routing، Governing Difference Conservation، Branch-Falsify-Merge،

Execution Proof Priority، Reconstruction Memory، Cross-Task Skill Diffusion، Immediate Writeback Compounding، Useful Agent Diversity، Parallelism Boundary، World-Memory Closure، Multi-Surface Output، Disciplined Opening، Persistent Frontier/HyperFrontier، Noise Subtraction، Multi-Currency Compute Budget وAsymmetric Verification.

هذه النظريات يجب تصنيفها PROPOSED_THEORY حتى تُعاير بمقاييس وتجارب. أفضل طريق لتحويل M-TAOS من هندسة مفاهيمية إلى برنامج علمي هو بناء benchmark يختبر كل قانون ضد baselines: single-agent، no-writeback، no-routing، no-court، no-memory، fixed-compute وunbounded variants.

14. MOUSSAIF9B Kernel Engineering

MOUSSAIF9B اسم معماري/مشروعي، ولا يجوز استخدامه كادعاء عدد parameters ما لم يكن النموذج المدرب فعلاً بهذا الحجم. النواة المقترحة هي overlay model-neutral تحمل الدستور، memory، routing، state، task contract، court hooks، interfaces وtool authorization. المعرفة والمهارات الثقيلة تبقى خارج kernel كموارد قابلة للتركيب.

هدف هندسي نشط في محادثات المشروع هو العمل مع Qwen3-30B-A3B IQ3_XS ضمن kernel مضغوط تقريباً في نطاق 18–20GB، وإضافة حزمة معرفة متخصصة تقارب 200MB، مع نحو 40 HyperLoops و40 HyperAbilities و500 HyperSkills و50 Knowledge OS. هذه أرقام REQUESTED_TARGET ما لم تتوفر manifests تثبت الإنتاج. القيمة البحثية ليست في العدد بل في mapping الكثيف بينها.

لكي ينافس نظام مبني على نموذج مضغوط قدرات frontier في مهام محددة، يجب أن يستفيد من external compute، specialization، self-verification، long-term memory وstructured task contracts. لا يوجد أساس لادعاء منافسة frontier عموماً دون benchmark متعدد المجالات تحت ميزانيات متكافئة.

15. Vision Neural World / ParallaxWorld-MCD

هذا الفرع يحاول إعطاء النظام world model متعدد المناظير بدلاً من الاكتفاء بحالة نصية. Parallax يعني الاحتفاظ بتمثيلات متنافسة لنفس الحالة واستخدام الاختلاف بينها كإشارة معرفية. يمكن ربطه بـMCD عبر قياس الفروق بين views وبـQ-MCD عبر اختيار السؤال الذي يفصل بين الفرضيات.

المطلوب لتقويته علمياً: temporal transitions، uncertainty، observation model، formal state schema، prediction tests، calibration، counterfactual simulator، وقياس الفرق بين imagined state وobserved state. بدون ذلك يبقى Vision Neural World استعارة معمارية مفيدة لكن غير مكتملة كـworld model theory.

16. Research Paper OS ومنهج الضغط التركيبي

المنهج البحثي في المشروع يرفض أن يكون عمق اللغة بديلاً عن عمق البرهان. paper pipeline المثالي: State of the Art → Conceptual Compression → Difference Extraction → New Mechanism → Formalization → Metrics → Hypotheses → Competing Models → Evidence Rank → Destruction Court → Limitations → Experimental Program → Reproducibility Assets.

ROFSC أو الضغط التركيبي المعرفي التأصيلي الموثوق الانفتاحي يعمل هنا كمنهج لضغط المادة من دون فقد الآليات أو الاختلافات. المحاكم المصاحبة: عدم الضياع، الهلوسة، الاختزال، الكفاءة التركيبية، الكفاءة السؤالية، الفجوات، التناقض، ورتبة الدليل.

كل نظرية جديدة يجب أن تنتج: تعريفاً غير دائري؛ متغيرات؛ آلية؛ حدّاً فاصلًا مع النظريات المجاورة؛ أداة أو protocol؛ metric؛ فرضيات قابلة للاختبار؛ predictions؛ failure conditions؛ أقوى اعتراضات؛ وبرنامجاً بحثياً. هذه القاعدة هي ما يحول الفكرة من تسمية إلى contribution.

17. Theory Registry الحالية

تم رصد Theory Registry مولدة في Drive تحتوي عقداً مرقمة وتفصل thesis، required_object، primary_metric، failure_signature وproject_routes. من العقد المبكرة المتحققة: MASO (Moussaif Agentic-System Ontology)، M-AGIOS-K (AGI Operating-System Kernel Theory)، MCIL (Constitutional Identity Layer)، MGCG (Goal-Contract Grammar)، Q-MCD (Question-Governance Principle)، MCD (Causal-Difference Calculus)، MCVT (Computational Vascularity Theory)، MELT (Evidence-Ladder Theory)، MGFC (Gap-Filling Court)، MDFC (Destruction and Falsification Court)، MMT (Mechanism-Trace Theory)، MWSM (World-State Model)، MPWG (Project-World (Graph)، MHKOS (HyperKnowledge KB-as-OS Theory وMPART (Provenance-Aware Retrieval Theory).

هذه السجلات مفيدة لأنها تفرض على كل نظرية object قابلاً للبناء ومقياساً وفشلًا. يجب اعتبار status مثل proposed_core كما هو وعدم ترقيته آلياً إلى validated. المرحلة التالية هي ربط كل node بالأوراق والتجارب والـKB والـskills المرتبطة بها.

18. Verified Drive Artifact Anchors

VERIFIED_DRIVE: Vascular Questioning Core Ontology — MCD, Q-MCD, MCAI and HAKIM OS.md موجود بنسخ متطابقة الحجم تقريباً؛ أحدها يزيد على عشرين ميغابايت. :VERIFIED_DRIVE

MOUSSAIF_FATTAH_AGENTIC_AGI_HYPERKB_700K_LINES موجود بعدة نسخ كبيرة تتجاوز عشرات الميغابايت. VERIFIED_DRIVE: AGI/Hermes HyperKB OS Autonomous Learning HyperLoops V2/V3 موجودة كملفات Markdown كبيرة.

VERIFIED_DRIVE: M-TAOS v1.0 موجود كـGoogle Doc أساسي وCANONICAL MIRROR ونسخة recovered. VERIFIED_DRIVE: ParallaxWorld-MCD FULL AGI OS recovered في خمسة أجزاء على الأقل. VERIFIED_DRIVE: Sovereign Continuity Runtime — MOUSSAIF9B Memory OS V3، MOUSSAIF MASTER MEMORY BACKUP، وTotal Programs Projects Memory V2 موجودة. VERIFIED_DRIVE: Theory Registry — 200 Nodes موجود كوثيقة مولدة.

VERIFIED_DRIVE: Sovereign Continuity Vault — MOUSSAIF9B موجود كمجلد مستقل تاريخي، ومعه Memory Ledger DOCX وmanifests مرتبطة. VERIFIED_DRIVE: PDF بعنوان AGI-Oriented Cognitive Operating System — MOUSSAIF Theoretical Architecture — 600 Pages موجود، وكذلك HyperKnowledge OS Level Engine — Master Prompt Architecture — 600 Pages.

قاعدة deduplication: لا تُحذف النسخ الكبيرة لمجرد تطابق الاسم أو الحجم. يجب أولاً فحص SHA-256 أو مقارنة binary، ثم near-duplicate diff، ثم semantic difference. النسخة الأحدث قد تكون mirror لا superseder، والنسخة الأقدم قد تحتوي تفاصيل حُذفت لاحقاً.

19. العلاقات بين العائلات

MOUSSAIF Unified AGI هو MCD/Q-MCD/HAKIM. umbrella architecture يمده بمنطق السؤال والفرق والتوجيه. HyperKnowledge OS يمده بالمعرفة القابلة للتشغيل. HyperSkills تحول المعرفة إلى قدرات. Bounded HyperLoops تحول القدرات إلى دورات تنفيذ. M-TAOS يترجم هذا إلى task runtime. ParallaxWorld يضيف world-state/world-model. Sovereign Continuity Vault يضمن الذاكرة والاسترجاع. Research Paper OS يحول النظريات إلى برنامج علمي قابل للنقد.

العلاقة مع Hermes/Codex/Ollama/Providers هي implementation layer: Main Brain يوزع الأدوار على النماذج والأدوات، لكنه لا يعرّف النظرية الأساسية. لذلك يجب عزل vendor-specific configuration عن canonical cognitive contracts حتى يبقى kernel قابلاً لنقل النماذج.

20. نموذج البيانات المطلوب للمشروع

كل Project يجب أن يمتلك project_id ثابتاً، canonical artifacts، current_state pointer، mission، status

artifact_id، title، format، يمتلك كل Artifact. open tasks وdependencies، related theories، tools، risks إن Drive ID، source type، original filename، version، status، canonicality، language، content hash توفر، dependencies ،semantic fingerprint وconfidence.

كل Theory تمتلك theory_id، definition، status، mechanism، variables، equations، evidence level، lineage، dependencies، falsifiers، predictions، metrics وsource artifacts. كل Skill تمتلك contract. كل Loop تمتلك budget/stop/rollback. كل Agent يمتلك mission/tools/permissions. كل Claim يمتلك evidence grade ومصدره. هذه schemas هي أساس RAG وGraphRAG وتمنع الخلط بين الطبقات.

21. Project Continuity Contract

إذا اختفت المحادثات، يجب أن يستطيع وكيل جديد إكمال المشروع من Drive عبر خمس خطوات: يفتح هذه الوثيقة؛ يقرأ CURRENT_STATE وLATEST؛ يفتح Artifact Index؛ يحمّل فقط KBs المرتبطة بسؤاله؛ ثم يكتب delta جديداً إلى changelog وmemory بعد التحقق. لا يجب أن يبدأ بقراءة كل الملفات الضخمة.

إذا تعارضت وثيقتان، يسجل conflict ولا يمحو أحدهما. إذا تغير الاسم، يحفظ alias lineage. إذا صارت نظرية deprecated، تبقى في history مع سبب التجاوز. إذا ظهر artifact مفقود من chat، يسجل REFERENCED_BUT_NOT_FOUND حتى يُستعاد. إذا ظهرت secrets، لا تُنسخ إلى الذاكرة العامة بل تسجل security flag فقط.

22. Current Engineering Targets

الهدف الهندسي الأقرب هو بناء kernel فعلي يمكن تحميله فوق نموذج محلي أو API model عبر bootstrap قصير، ثم استدعاء المعرفة والمهارات on-demand. يجب أن يكون boot context صغيراً، بينما تعمل KBs الثقيلة كـretrieval substrate. نجاح التصميم يقاس بقدرة النموذج على الاحتفاظ بالعقود والحوكمة عبر تبديل provider/model، لا بحجم prompt الابتدائي.

الهدف الثاني هو تحويل 40 HyperLoops و40 HyperAbilities و500 HyperSkills و50 Knowledge OS من قوائم أسماء إلى graph مع dependencies وtests. يمكن بعد ذلك إجراء auto-deployment داخل task runtime حيث تُختار skill/loop بناء على MCD وtask contract.

الهدف الثالث هو بناء benchmark يختبر claims على نموذج baseline ثم على baseline+KB، +routing، +courts، +tools، +multi-agent، +memory. يجب قياس task success، factual precision، cost، latency، tool errors، recovery، cross-run retention وworld-effect verification.

الهدف الرابع هو اختبار portability على Qwen3-30B-A3B IQ3_XS ونماذج محلية أصغر ونماذج frontier عبر API. إذا

بقيت العقود والـmetrics ثابتة، يمكن فصل contribution المعمارية عن قوة النموذج الأساسي.

23. المخاطر العلمية والهندسية

الخطر الأول هو إعادة تسمية مفاهيم معروفة من agent architectures وRAG وplanning وself-critique من دون إضافة قابلة للقياس. العلاج: related-work matrix وnovelty claims دقيقة. الخطر الثاني هو تضخم الأرقام: آلاف skills أو ملايين الأسطر لا تساوي قدرة. العلاج: verified counts + usefulness metrics.

الخطر الثالث هو self-critique cosmetic. العلاج: holdout evaluators، counterexamples، external tools وfailure-triggered repair. الخطر الرابع هو memory poisoning. العلاج: evidence-gated writeback، contradiction preservation، versioning وrollback. الخطر الخامس هو over-routing: تكلفة orchestration تتجاوز الفائدة. العلاج: marginal compute value gates.

الخطر السادس هو world-effect risk. أي agent يرسل، ينشر، يحذف، يدفع، ينشر كوداً أو يغير ملفات حساسة يحتاج authorization وpost-action verification. الخطر السابع هو project leakage حيث تخلط ذاكرة business أو health أو unrelated research مع kernel. العلاج: project boundaries وaccess-aware retrieval.

24. برنامج البحث التجريبي

Experiment Family A — Governing Question: قارن direct prompting مع Q-MCD reformulation على benchmark متعدد التعقيد، واقس error reduction، question lift وcompute overhead. Experiment Family B — Vascular Routing: قارن fixed compute بـadaptive routing وقس success per compute. Experiment Family C — Courts: ablation لكل court لقياس hidden premises، hallucination وfalse novelty.

Experiment Family D — Memory: no-memory مقابل episodic-only مقابل stratified reconstruction، مع قياس cross-run consistency وerror persistence. Experiment Family E — HyperKnowledge: raw corpus مقابل structured KB-as-OS مقابل graph-linked KB، مع قياس retrieval-to-action precision. Experiment Family F — Agent Diversity: single agent مقابل homogeneous multi-agent مقابل orthogonal roles.

Experiment Family G — World Model: text-state مقابل versioned world-state snapshots وParallax hypotheses. Experiment Family H — Tool Execution: answer-only مقابل execution-proof harness. Experiment Family I — Portability: نفس kernel contracts عبر small model، local Qwen وfrontier

Scale — Experiment Family J .model: زد عدد skills/KBs مع قياس saturation بدلاً من افتراض monotonic improvement.

25. AGI-Readiness دون ادعاء AGI

المشروع يستخدم AGI-oriented كاتجاه تصميم. أي AGI-Readiness Index يجب أن يكون multi-dimensional: task breadth، adaptive planning، memory continuity، tool competence، uncertainty calibration، recovery، cross-domain transfer، world-state grounding، autonomy under bounds، constitutional compliance وcost efficiency. لا يوجد threshold واحد يثبت AGI.

الهدف الواقعي هو إظهار أن architecture + memory + tools + structured knowledge ترفع competence density بالنسبة لنفس base model. إذا تحقق ذلك عبر domains متعددة وتحت اختبارات adversarial يمكن الحديث عن agentic generalization improvement، لا frontier equivalence إلا بدليل مباشر.

26. AI Boot Protocol

BOOT-0: اقرأ LATEST وBOOT-1. CURRENT_STATE: حدد BOOT-2. project/task contract: استخرج BOOT-3. governing question: ابنِ BOOT-4. active difference map: استخدم artifact registry لاستدعاء أقل مجموعة مصادر لازمة. BOOT-5: طبّق BOOT-6: route compute/tool/models حسب evidence ladder. BOOT-7. marginal value: نفذ branches المتعامدة. BOOT-8: شغّل BOOT-9. destruction court: نفذ العالم فقط ضمن الصلاحيات. BOOT-10: تحقق من النتيجة. BOOT-11: اكتب delta إلى memory وchangelog.

عند غياب مصدر مهم لا تختلقه. ضع MISSING أو REFERENCED_BUT_NOT_FOUND. عند وجود نسخة متعددة لا تختَر newest فقط؛ افحص canonical declarations وunique differences. عند سؤال يحتاج ورقة علمية افتح research registry والمصادر، لا تعتمد على memory summary وحدها.

27. Continuous Ingestion Protocol

INGEST → CLASSIFY → PROJECT MATCH → DEDUPLICATE → DIFFERENCE EXTRACT → VERSION → REGISTER → CANONICALIZE → LINK → VALIDATE → BACKUP. كل تشغيل جديد يركز على delta منذ آخر state. لا يعاد بناء الأرشيف من الصفر.

في chat-derived knowledge، يُستخرج فقط: prompt، architecture، formula، specification، decision، unresolved question، artifact reference، target، constraint، correction. chatter لا يدخل current state.

raw export إن توفر يُحفظ منفصلاً عن derived memory.

28. Demolition Court لهذه الذاكرة

Attack 1: قد تكون بعض الأصول المكتشفة duplicates. الإصلاح: لا حذف قبل hash/diff. Attack 2: بعض عناصر المشروع تأتي من chat context لا Drive. الإصلاح: وسمها CHAT_CONTEXT. Attack 3: Theory Registry مولدة حديثاً وقد لا تمثل lineage كاملة. الإصلاح: ربط كل theory لاحقاً بمصدرها الأقدم. Attack 4: الأرشيف الحالي لا يملك export كاملاً لكل ChatGPT chats. الإصلاح: عدم ادعاء coverage كامل.

Attack 5: central document قد يصبح ضخماً وصعب الاسترجاع. الإصلاح: هذه الوثيقة خريطة كثيفة وليست بديلًا عن registries والـchunks. Attack 6: canonical root يمكن أن يصبح single point of failure. الإصلاح: metadata ذاتية الوصف، project-level files، manifests وuser targets. Attack 7: snapshots قد تندمج مع verified outputs. الإصلاح: status taxonomy إلزامية.

Attack 8: نظرية جديدة قد تكون equivalent لمفهوم معروف. الإصلاح: novelty court و.related-work comparison Attack 9: كثرة layers قد تجعل التنفيذ هشاً. الإصلاح: lean kernel + modular activation. Attack 10: الذاكرة قد تتقادم. الإصلاح: source freshness، current-state timestamping وincremental updates.

29. Top Open Work

أعلى الأعمال المفتوحة قيمة: بناء Artifact Registry خاص بهذا المشروع مع IDs وروابط Drive؛ استخراج Theory Lineage من ملفات أقدم؛ حساب hashes للنسخ الكبيرة حيث يمكن؛ تحديد canonical copy لكل HyperKB؛ تحويل 500 HyperSkills المستهدفة إلى contracts متحققة؛ بناء 40 Bounded HyperLoops كملفات مستقلة؛ بناء 40 HyperAbilities كطبقة capability؛ تحديد 50 Knowledge OS الفعلية مع maturity score؛ بناء benchmark suite؛ تنفيذ ablation program؛ ربط M-TAOS بالنواة؛ بناء local Qwen bootstrap؛ اختبار provider portability؛ إنشاء SECURITY_REVIEW؛ إنشاء relationship graph؛ استعادة raw ChatGPT exports عند توفرها.

هذه القائمة ليست وعداً بأن العناصر موجودة الآن؛ هي roadmap. كل عنصر ينتقل من PLANNED إلى GENERATED ثم VERIFIED فقط بعد evidence readback.

30. الخلاصة الكانونية

جوهر AGENTIC THINKING AGI ليس حجم الملف ولا عدد الوكلاء، بل نظام فرق وحاكمية وتنفيذ وذاكرة.

MCD/Q-MCD يحددان ما يستحق التفكير؛ Vascular Routing يحدد أين تذهب الحوسبة؛ HyperKnowledge يحدد ما الذي يُسترجع؛ HyperSkills تحدد ما الذي يمكن فعله؛ Bounded HyperLoops تحدد كيف يستمر الفعل؛ Agents/Tools تنفذ؛ Courts تمنع السلاسة من أن تحل محل الدليل؛ World Effect يربط cognition بالعالم؛ Self-Reconstruction Memory يجعل كل تشغيل يضيف حالة قابلة للاستمرار.

هذه الوثيقة هي نقطة الاستعادة الأولى للمشروع. يجب ألا تُستبدل بصمت. أي تحديث جوهري ينشئ version/changelog، ويحافظ على النسخة السابقة، ويحدث LATEST وCURRENT_STATE وArtifact Index. الهدف النهائي هو أن تصبح Google Drive ذاكرة خارجية persistent AI-readable يمكن أن ترثها نماذج مختلفة دون إعادة بناء المشروع من المحادثات المبعثرة.

31. Source Anchor IDs

Drive anchor IDs verified during this consolidation include:
1Out5tT-Xit1hVx0rUzOKtc-smL1KMeau (Vascular Questioning Core Ontology);
1FTCAXB0d1F0ToAmQ561Iqybd83yYu-ed (dated ontology variant);
1aiMFUMAxvH5MopdYi3HlNYFsgKyEZfgc and related large HyperKB variants;
1NpRXvv7TzV3wDwEj5bBQMic9TL0BHrfx (Hermes HyperKB V3);
14mk-MA3eX4cAWExbAzxkgI0Ak2hES0Qn (Hermes HyperKB V2);
1KhuTnr8n6to1Hn3FIpWwRhwBMGAJugAKQE-fdndBcQU (M-TAOS primary);
1N9Bh1dV2NnDUT_cax3-Hdb79_qxwDtMYEkOYd95SdA8 (M-TAOS canonical mirror);
1HGQFvjVAveaxPGJZp4HSanT3vQOXbhRDYG8tnwj30m4 (M-TAOS recovered);
16WR4cH5akzUoM1a1CIfBATULz8JZ0_zGNSdaYST4LEQ (Theory Registry);
1VSrw7Jp4hof5cp5TsvtCqRYbfgtjTuyrj4BeHsUfZiE (Memory OS V3);
1-Dtkrx2i0zKucj6V6cP9WFPDd4NOuUDE (Master Memory Backup);
1Y5kPr8UkmrNJCE3dwUjf9COW3qOtTKgC (Total Programs Projects Memory V2);
1L-HmJZGUfRMlkV5i6wLKQTcXItpG4FCH (Sovereign Continuity Vault folder);
1xuT-M_O33cn06iWel7UIaWxUqtG0wz3x (HyperKnowledge OS Level Engine PDF);
.(1J4epGVuSsamlCvvzQ73pGJ9gUFXutYYp (AGI-Oriented Cognitive OS PDF

Parallax recovered anchors observed:
1qVAfi2OEsONpk6P8dNB-X2jevboRu9-cqomDKFgdNXc،
1rM4Hs3teMic5ClAygNZH9auq5SoCGVAxovqW3bIfX1g،
1vHwWKkHE5B88V76pFFz2AAwxl7xcNL_5W_Y8fRhQhv8،
1GhOd5Ab-evBHwX0qx4CcYZi4VH8pyKnGMksirIK5MQg، مع جزء خامس ظهر في مجموعة recovery

ويجب إدخاله في Artifact Index النهائي بعد تحقق مباشر.

هذه IDs هي pointers فقط؛ لا تثبت وحدها canonicality أو completeness. المرجع النهائي يجب أن يستخدم Artifact Registry مع status وhash وprovenance.

32. Consolidation Run Delta — 2026-08-07

VERIFIED_WRITEBACK: تم إنشاء حزمة handoff تشغيلية داخل مجلد المشروع نفسه: README.md، AI_CONTEXT_PACK.md، 01_CURRENT_STATE/CURRENT_STATE.md، و00_CONTROL/LATEST.md + TIMELINE.md + DECISIONS.md.

VERIFIED_WRITEBACK: تم إنشاء Project Artifact Index قابل للفرز والاستعلام في Google Sheets تحت 00_CONTROL، ووصل في هذه الجولة إلى 43 أصلاً عالي الصلة، مع stable IDs من ART-MH-0001 إلى ART-MH-0043، evidence status، canonical role، source date، Drive ID/URL، relationship، confidence وnotes. الرابط:
https://docs.google.com/spreadsheets/d/1-fAlZr-g5tJ4fQHWqhXsGDicOYpvn3Ikui9VUavj3rk

RESOLVED: ParallaxWorld-MCD Part 05 of 05 تم التحقق منه مباشرة وإدخاله في Artifact Index. Drive ID: 1fzx5L9eqz5y4vLEJVT4FwX2gScpOtpEBHQ6gmidI7g4. لذلك لم يعد هذا الجزء REFERENCED_BUT_NOT_FOUND.

RESOLVED: canonical split بين الوثيقة الحالية وملفات handoff. LATEST/README/CURRENT_STATE تشير الآن إلى هذه الوثيقة بوصفها primary canonical project-memory entry، بينما تبقى ملفات handoff derived companions ولا تُحذف.

NEW VERIFIED ANCHORS: Hermes HyperKB V2 `14mk-MA3eX4cAWExbAzxkgI0Ak2hES0Qn`، Hermes HyperKB V3 `1NpRXvv7TzV3wDwEj5bBQMic9TL0BHrfx`، HyperLoop Runtime V3

`1z_maEr78yOk-i20_U8j_Rkf1vLDYIDGQ`، HermesRootForge
`1URDlnHMosx0JULy-JLcGmHdjJQQpikK5`، HyperKnowledge OS V3
`1J3iT8KmmhJSSbP1I0GcaOxh3CnFabZvv`، Theory Registry
`16WR4cH5akzUoM1a1CIfBATULz8JZ0_zGNSdaYST4LEQ`، Continuity Vault
`1L-HmJZGUfRMlkV5i6wLKQTcXItpG4FCH`، Master Memory Backup
`1-Dtkrx2i0zKucj6V6cP9WFPDd4NOuUDE`، Total Programs Projects Memory V2
.``1Y5kPr8UkmrNJCE3dwUjf9COW3qOtTKgC

؛REMAINING HIGH-VALUE FRONTIER: recursive project-wide Drive inventory standalone contract ؛raw ChatGPT export ingestion ؛SHA-256/content-diff duplicate clusters ؛extraction for targeted 500 HyperSkills/40 HyperLoops/40 HyperAbilities/50 Knowledge OS benchmark ؛security/secret audit ؛deployment-state verification for agents/automations .telemetry and ablation suite. These remain PARTIAL/PLANNED, not completed