
Cassandra T2 Design Document

Architecture, training plan and pre-registered success criteria for a retrieval-native masked-diffusion language model, with every threshold and falsification condition fixed in advance.

Thomas Garren

SOPHIA XT LLC

19 September 2026 · v0.2

sophiaxt.com

CC BY 4.0. Code referenced herein is Apache 2.0.

Abstract

Cassandra T2 is a design for a masked-diffusion language model whose attention layer is structurally aware of its retrieval substrate, pre-registered here in full before the first training run. The design composes four proposals: an anchor-aware attention block with content, timestep and anchor paths; retrieval priors that let the anchor path consume per-anchor relevance scores and lattice coordinates from the LTMi-XT v0.1 format; lattice-compressed context, in which older context is crystallised into LTMi-XT loci and addressed through the same anchor path as user-supplied retrievals; and a preconditioning training stack aimed at cutting wall-clock time on one consumer GPU. Section 2 sets out the measured base the design rests on, all of it from the released Cassandra T1 line and the anchor-token masking study deposited under DOI 10.5281/zenodo.20496783: a 1.67x pooled held-out advantage with a 95 percent bootstrap interval of [1.51, 1.85], a mechanism located in the value and output projections by a four-arm ablation, and a continued-pretraining checkpoint that moved held-out overlap from 0.008 to 0.481. Section 5 fixes the seven-arm ablation grid, the metrics, the statistical procedure, the four success thresholds and the conditions that would falsify the design, every one of them written down before a result exists to fit them to. One of the four proposals already carries a published null: a three-way ablation from May 2026 found the lattice coordinate empirically content-free as a conditioning signal, and section 3.2 puts that result next to the proposal it constrains. Every T2 performance figure here is a target labelled at the point of use, and the status ledger in section 1.1 classes each quantitative claim as measured, target or estimate.

1. Status, and how to read this document

This document is a pre-registration. The architecture, the training stack, the seven-arm ablation grid, the metrics, the statistical procedure and the four thresholds that count as success are all fixed in this revision, and they were fixed before any T2 result existed to fit them to. No T2 training run has been made. Section 6 holds the results and is empty on purpose, so that a later revision fills it in place rather than replacing this document with a differently shaped one.

The reason section 6 is empty is not that the work stopped. It is that the work moved. The line ran from block, through fusion, to a compositional architecture, and each step was taken because evaluation said to take it rather than because the previous design failed to build. T2's framework was built out in full and the run was not made, because by the time it would have been made the compositional route was returning better numbers on the same benchmarks. That route shipped as Mnema V1, which is live. This document is published rather than withdrawn because the pre-registration is the record of what was committed to before the comparison existed, and a lab that only publishes the branch it took is a lab whose published branches cannot be checked.

The rule applied throughout is that a number either names the run or the artifact it came off, or it says in the same sentence that it is a target. A reader who cannot tell the two apart discounts both, which is the failure mode a design document has to design against, and the ledger below is how this one does it.

1.1 The status ledger

Every quantitative claim in this document falls into one of the rows below. A reader who reads nothing else should read this table.

Table 1. The status of every class of number in this document. Measured rows name a public artifact. Target rows are commitments about what will be tested, not descriptions of what happened.

Claim	Status	Where it comes from
Cassandra T1 architecture and released artifact identity	Measured	Public repository and the reproduction page; file hashes published
T1 training loss 3.78 to 2.2561 over five epochs	Measured	Published training trajectory for the released checkpoint
Anchor-token masking gives 1.67x pooled held-out advantage	Measured	Masking study, DOI 10.5281/zenodo.20496783
Value and output projections carry the effect	Measured	Masking study, ablation in section 5.7 of that paper
Continued pretraining moves held-out overlap 0.008 to 0.481	Measured	Masking study, section 6.4a; a checkpoint, not a released model
Forced-anchor decoding preserves locked positions at 1.0	Structural	Locked positions are excluded from the unmasking loop by construction
The lattice coordinate carries no usable conditioning signal	Measured null	Three-way coordinate ablation, 12 May 2026
T2 quality improvement over the T1 line	Target	Section 5.5 threshold, to be tested by arm V6 of the grid
T2 wall-clock reduction from the preconditioning stack	Target	Section 4.5 arithmetic over per-technique published claims
Parameter overhead of the three-path attention block	Estimate	Arithmetic over duplicated projections, pending a count off the built module
Addressable memory ceiling of lattice-compressed context	Arithmetic	Lattice cell count times an assumed per-cell capacity
Compute budget for the ablation grid	Estimate	Scaled from the continued-pretraining run. Not a quoted price

1.2 Relationship to the released line

Cassandra T1 is the released model. It is open under Apache 2.0, it ships a verified epoch-5 FP16 checkpoint, the tokenizer, the architecture and training utilities, and it publishes no standard benchmark results because none were measured. It is positioned as a research-stage proof of concept, and the repository says so.

Where the design needs a warm-start point it names a continued-pretraining checkpoint rather than a second released model: the released epoch-5 checkpoint plus 500 steps of anchor-token-masked full-model training, reported in section 6.4a of the masking study. That checkpoint is a research artifact produced inside that study. It sits outside the released artifact set and carries no version number in the model line, so this document gives the checkpoint and the recipe and the starting point gets rebuilt from the released weights rather than downloaded. The revision history at the end of this document records what changed from v0.1, including the removal of a model name that never referred to a release.

2. What is already measured

This section is the entire evidential base the T2 design rests on. Everything in it predates the design and was measured on the T1 line.

2.1 Cassandra T1

A 1.3 billion parameter masked-diffusion transformer: 28 layers, hidden size 2,048, 16 query heads against 4 key-value

heads in grouped-query attention, head dimension 128, SwiGLU feed-forward at intermediate size 5,632, rotary position encoding, RMSNorm, a 32,768-entry BPE vocabulary with the mask token at id 32,766 and the pad token at 32,767, 128K context with a 4,096-token sliding window and 256 global tokens, and 12 denoising steps at default inference. Multiplying out that configuration gives 1,329,711,104 parameters, and halving the byte size of the published FP16 checkpoint gives 1,329,750,332, the difference being the serialisation header.

Training cross-entropy over five epochs ran 3.78, 2.89, 2.67, 2.42 and 2.2561. The model is undertrained at the released checkpoint and produces incoherent unconditional output, which is a property of the run rather than a capability claim. The released file is 2,659,500,664 bytes with SHA-256 beginning D70C813C, and the reproduction page walks a reader through verifying it from the LFS pointer files without downloading 2.6 GB.

2.2 Anchor-token masking

In a training pair consisting of a question Q and an answer statement A, anchor-token masking never masks Q and applies the noise schedule to A at every step, with loss taken only over masked answer positions. The study compared it against random-position masking over both segments and against a reverse-asymmetric control that masks only Q. All three arms shared a base checkpoint, a corpus, an adapter rank and 7.0M trainable parameters. Only the mask differed.

Table 2. Pooled forced-anchor corpus overlap over 48 held-out queries in four domains the adapters were not fine-tuned on. Measured; DOI 10.5281/zenodo.20496783.

Arm	Pooled held-out overlap	What it is
base, no fine-tuning	0.0045	the floor
reverse-asymmetric	0.176	the direction control
random-position	0.2137	the control
anchor-token	0.3571	the arm under test

The pooled ratio of anchor to random is 1.67x with a 95 percent bootstrap interval of [1.51, 1.85] over 10,000 resamples, and $P(\text{anchor greater than random})$ of 1.000. The in-domain ratio is 1.26x with an interval of [1.18, 1.34]. The two intervals do not overlap, which is the finding that the advantage is larger out of distribution than in it. The reverse arm sits below the random arm at P equals 0.006, which establishes that the direction of the asymmetry is load-bearing rather than its presence. The effect replicates across astronomy, medical and legal held-out corpora and is weaker but still positive on a cooking corpus.

A four-arm ablation located the mechanism. Adapters wrapping only the value and output projections reached 0.416 held-out overlap on 2.98M trainable parameters, above the full adapter at 0.337 on 7.0M. Adapters wrapping only the query and key projections reached 0.136 on the same 2.98M, below the random-position control. Two arms with the same

trainable parameter count land a factor of three apart because of where the parameters sit. The account this supports is not that the model learns to look at anchors more, but that it learns to extract anchor-extending content from positions it was already attending to.

2.3 The continued-pretraining checkpoint

Section 6.4a of the same study probes whether the objective survives past adapter scale. Full-model continued pretraining of the released epoch-5 checkpoint under the anchor-token-masking objective, on the same 144 question and answer pairs derived from the three in-domain corpora, ran 1,000 steps with checkpoints at 500 and 1,000. The hyperparameters were a learning rate of $5e-5$ with 50-step warmup and cosine decay to $5e-6$, batch size 2 with gradient accumulation 2, sequence length 512 and bfloat16 throughout. Total wall clock was 7.6 minutes on one consumer GPU at a memory peak of 10.8 GB.

Table 3. Measured, on the 12-query held-out subset used in that section. The base figure of 0.008 is from that subset and is not the 0.0045 of Table 2, which pools 48 queries. The two are different eval sets and should not be compared across rows.

Checkpoint	Held-out pooled overlap	Against base
epoch-5 base, no continued training	0.008	reference
continued, step 500	0.481	58.7x
continued, step 1000	0.440	53.7x

Two things about this result matter for the T2 design. The first is that the step-500 value exceeds every adapter-scale result in the same study, which is why it is the warm-start candidate. The second is the trade the study publishes alongside it: unconditional generation degrades. Free-form prompts about subjects outside the training set produce output filled with training-corpus references rather than topical extension. The checkpoint has become a specialised forced-anchor extender. For a system where forced-anchor decoding is always applied that trade is acceptable, and for general use it is not. Any T2 run warm-started from this checkpoint inherits the trade and has to be evaluated with it in view.

2.4 LTMi-XT v0.1

The retrieval format underneath the design. It crystallises prose into atomic self-contained statements called loci, gives each a four-level breadcrumb of topic, subtopic, concept and claim, and derives a three-dimensional coordinate on a 64 by 64 by 64 lattice by hashing breadcrumb prefixes with BLAKE2b. Loci sharing k breadcrumb levels share k coordinates. Identifiers are content-addressed over canonical JSON. The lattice holds 262,144 cells. Over 10,000 trials a breadcrumb maps to the same cell 10,000 times, and 1,000 of 1,000 breadcrumbs sharing topic, subtopic and concept share

a cell. The specification deliberately publishes no benchmark numbers.

Determinism is a property of the address rather than of its meaning, and section 3.2 of this document is where that distinction becomes load-bearing.

2.5 Forced-anchor decoding

Given a query and a retrieved locus statement, the runtime tokenises the statement and locks its tokens at fixed positions of the answer slot. The unmasking loop excludes locked positions at every denoising step and the model fills the surrounding positions conditional on them. Preservation is structural rather than statistical: no decoding strategy and no failure mode can dislodge a position the loop never visits. The guarantee holds on masked-diffusion backends and drops to best effort on autoregressive APIs, where the published test suite asserts a rate below 1.0.

2.6 What is implemented but untrained

The three-path attention block of section 3.1 is implemented behind a configuration flag and unit-tested, so the first arm of the grid has code to run against. It has never been through a training run, and no claim in this document rests on its behaviour. The lattice-compressed context mechanism of

section 3.3 exists as an interface sketch rather than as working code.

3. Proposed architecture

Four changes are proposed against the T1 architecture. Each is stated with what it is supposed to buy and with the observation that would show it did not, and section 5 gives each of them its own arm so that the four can be told apart.

3.1 Anchor-aware triple attention

The sliding-window grouped-query attention block is replaced by a three-path block. All three paths read the same input hidden state and their outputs are mixed by a per-position softmax gate. The content path is bit-for-bit identical to the existing attention and reproduces single-path behaviour when the gate isolates it. The timestep path takes queries from the hidden state and keys and values from the diffusion timestep embedding broadcast across positions, which lets diffusion stage act as a routing signal rather than reaching the block only through normalisation conditioning. The anchor path takes queries from the hidden state and keys and values from the hidden state with attention scores masked so that each query position attends only to anchor positions. During training the anchors are the always-visible question tokens. During inference under forced-anchor decoding they are the locked positions.

The gate is a linear layer over the hidden state and an anchor-presence indicator, producing three logits. Its bias is initialised to $[2, 0, 0]$, which gives a softmax of approximately $[0.79, 0.105, 0.105]$ at step zero. The intent is that the two new paths start as small perturbations the optimiser can learn into rather than as a discontinuity at the start of training. An earlier draft of this document stated the initial gate as $[0.88, 0.06, 0.06]$, which does not follow from that bias; the arithmetic above is the correct reading and the bias is the thing to change if a different starting mixture is wanted.

Warm start matters more than the gate. Content-path weights initialise from the single-path continued-pretraining checkpoint and the two new paths initialise at random but

contribute about a tenth each through the gate, so the first training step of a T2 run behaves close to its starting checkpoint and drifts as the optimiser turns the new paths on. Whether a single-path checkpoint is a stable starting point for a three-path architecture is the first thing the ablation grid tests, and it is the most likely place for the design to fail early.

Parameter overhead is estimated at roughly 48 percent over the single-path baseline at this scale, almost all of it in the duplicated query, key, value and output projections of the two additional paths. That figure is arithmetic over the proposed module rather than a count taken off a built model, and it should be re-derived from the built module before it is quoted anywhere else.

3.2 Retrieval priors on the anchor path, and the null that constrains them

The anchor path can optionally consume two tensors supplied at the boundary between the retrieval runtime and the model. The first is a per-anchor relevance score in the unit interval, zero at non-anchor positions, projected to a per-head additive bias on the anchor-path attention scores through a zero-initialised linear layer and scaled by a scalar gate initialised to zero. The second is the three-dimensional lattice coordinate of the source locus for each anchor position, zero at non-anchors, embedded through three learned tables of 64 entries each and added into the anchor-path key projection, scaled by a second scalar gate also initialised to zero. At initialisation both contributions are exactly zero and the variant is functionally identical to the plain three-path block.

The second of those two priors has already been tested once, in a narrower form, and the result was null. A three-way ablation published on 12 May 2026 compared the BLAKE2b coordinate against a principal-component projection of frozen-encoder embeddings and against uniform-random assignment, with architecture, data, warm start, optimiser and the 500-step schedule held identical. All three arms landed at the same training loss of 0.2834. Unforced inference came out byte-identical between the random and hash arms with a zero-width bootstrap interval.

Table 4. The published coordinate ablation. Paired-by-query bootstrap, 2,000 resamples, 36 held-out corpus and query pairs across three domains. The principal-component scheme reaches 96.7 percent one-nearest-neighbour topic accuracy in coordinate space against 21.7 percent for the hash, and that geometric advantage transfers to no downstream metric.

Comparison	Metric	Point	95 percent interval	Significant
PCA against BLAKE2b	forced overlap	-0.002	$[-0.034, +0.029]$	no
PCA against BLAKE2b	unforced overlap	exactly 0	$[0.000, 0.000]$	no
random against BLAKE2b	forced overlap	-0.016	$[-0.054, +0.022]$	no
random against BLAKE2b	unforced overlap	exactly 0	$[0.000, 0.000]$	no

What that experiment settled is narrow and it is worth stating exactly. The lattice is an address rather than a semantic geometry. It says nothing for or against LTMi-XT as a retrieval format, which is in production use and has not been benchmarked against alternative retrieval approaches. It also does not by itself rule out the coordinate prior proposed here, because the ablation varied the coordinate scheme under a fixed architecture rather than giving the model a learned path

through which coordinates could be used. But it does mean the burden has moved. The coordinate prior is now a hypothesis with one failed test behind it, and the consequence for the design is that the ablation grid in section 5.2 separates the relevance prior from the coordinate prior, so a positive result on the pair is attributable to whichever one earned it.

The same findings document publishes the adversarial review of that null, including the criticism that 36 pairs is too small against

a noise floor of about 0.02 and that 200 would be needed for the interval the lab wants. The following round moved to 172. The document also records prior art that was missed the first time and names it.

3.3 Lattice-compressed context

The proposal that makes T2 more than a rearrangement of the T1 block. Older context tokens are crystallised into LTMi-XT loci at intervals by the same crystalliser the retrieval pipeline uses, placed at their canonical lattice coordinates and fed to the anchor path through the same interface as user-supplied retrieval anchors. The model is not told which anchors came from a user and which it wrote itself. Both are lattice-indexed addressable content.

The components are not new and the document does not claim they are. Compressing older context into a smaller bank is the approach of Rae et al. (2020), which compresses through learned summarisation where this proposal compresses deterministically through the crystalliser. Retrieving from past activations is the approach of Wu et al. (2022), which uses learned nearest-neighbour lookup where this proposal uses hash-based lattice addressing at constant cost. Learning a memory module at test time is the approach of Behrouz et al. (2024), where here the crystalliser is deterministic and the model learns only how to use the memory rather than how to construct it. Memory-token approaches place memory at sequence positions; this places it at lattice positions, so position encoding does not have to extrapolate. The combination is what has no precedent we can cite, and that is a statement about the search we ran rather than a statement about the literature. The search was ours, it was done in 2026, and it was not exhaustive.

The capacity bound is arithmetic and it is a hard ceiling rather than a soft one. With 262,144 cells and an assumed per-cell capacity of roughly one locus statement of about 400 characters, which is on the order of 100 tokens, the addressable memory ceiling is about 26 million tokens per context. Past that, newer crystallisations evict older ones at the same cell through the last-referenced timestamp. The assumed per-cell capacity is the load-bearing assumption in that sentence and it has not been tuned against anything.

3.4 Latent factorisation of the key and value projections

For T2 the grouped-query attention of T1 is replaced by a latent factorisation in which keys and values pass through a per-layer low-rank latent at rank about 128 and are decompressed in the block, a published technique for reducing attention parameter count and inference-time key-value cache. It composes with the three-path block by sharing the latent cache across paths, each with its own decompression matrices, with the anchor mask applied after decompression. The published claim for the technique in isolation is a four to five fold reduction in key-value cache memory. That is a citation rather than a measurement of this design, and the combination with the three-path block, where the duplicated projections push in the opposite direction, is one of the things arm V5 of the grid exists to settle.

4. Training plan

A stack of techniques chosen to make the seven-arm grid runnable on one consumer GPU inside a few days. Every efficiency figure in this section is a target, carried over from the published claim for that technique measured separately, on other models, by other people, and section 4.5 states what the stack as a whole is committed to.

4.1 Data

The corpus that exists today is on the order of 10,000 loci across seven bundles, which is the corpus the masking study used. Training a T2 ablation grid needs an order of magnitude more. The commitment is to build a 50,000-locus bundle from a public encyclopaedic source before any T2 run starts. The builder script is written and has not been executed at full scale.

The known risk is shape rather than size. Loci derived from encyclopaedic prose may not match the question and answer shape the anchor-token-masking pipeline trains on, and synthetic pair generation through a third-party generation API is the mitigation. The cost of that generation is not stated here because any figure would be a quote from a price list rather than a measurement, and price lists move.

4.2 Optimiser

A second-order optimiser using a diagonal Hessian estimate to precondition the gradient (Liu et al., 2024) replaces AdamW, adopted with the published hyperparameter template. The published claim is approximately a factor of two speedup on language-model pretraining at matched perplexity. Risk is low because the failure mode is that it behaves like AdamW.

4.3 Curriculum

Training data is drawn from declarative decoding specifications at four complexity tiers: a single literal lock with no retrieval, a single-locus retrieval at top-1, a multi-locus retrieval at top-3, and a reasoning scaffold with multi-stage locks. Batches shift linearly from the first tier to the fourth over the first 30 percent of training and sample uniformly after that. The target is roughly a 1.5x convergence speedup to a given loss, taken from the general curriculum-learning literature; section 5.3 is where it gets measured for this setup.

4.4 Quantisation and kernels

Selective int4 in the feed-forward layers with bfloat16 retained in attention and embeddings, targeting roughly a 2.5x memory reduction and a 1.3 to 1.5x wall-clock improvement. Fused normalisation, rotary and activation kernels target roughly 1.4x, and a current fused attention kernel targets 1.2x on the hardware available here. Ternary weight quantisation (Ma et al., 2024) is deliberately not adopted in the first T2 configuration, because stacking it with three architectural changes would make a negative result uninterpretable.

4.5 The efficiency target, stated as arithmetic

Multiplying the published per-technique claims gives about 6.6x. That number is the product of five optimistic figures

measured in isolation, and the three-path block and the latent factorisation both add overhead that eats into it. The target this document commits to is a 4 to 6x wall-clock improvement

over the pipeline that produced the continued-pretraining checkpoint, on the same hardware, measured as seconds per step at matched batch and sequence length.

published per-technique claims, multiplied	$2 \times 1.5 \times 1.4 \times 1.3 \times 1.2 = 6.6x$
architectural overhead of the new paths	partially offsets the gains
target, same hardware, same baseline	4 to 6x seconds per step
status	target; section 5.3 measures it

An earlier draft framed this as wall-clock parity between the hardware available here and a datacentre accelerator, with a hardware ratio quoted from vendor peak-throughput figures. That framing is removed. Peak sparse throughput ratios do not predict wall clock on a real pipeline, the comparison was to hardware this lab does not have and cannot measure, and the interesting claim was never about the other machine. What can be measured here is seconds per step against a baseline run on the same card, and that is what section 5.3 logs.

5. Experimental plan

5.1 Baseline

The baseline arm is the step-500 continued-pretraining checkpoint described in section 2.3, whose held-out pooled corpus overlap is 0.481 on the 12-query subset that section

used. That checkpoint is not in the released artifact set. Rebuilding it from the released epoch-5 weights takes the recipe in section 2.3 and about eight minutes on one consumer GPU, and the T2 grid should rebuild it rather than assume a stored copy, so that the baseline and the variants share a provenance.

Before the grid runs, the baseline is re-evaluated on the eval sets the grid will use, which are not yet built. The 0.481 figure is a measurement on a 12-query subset and is not a baseline for a metric or an eval set that does not exist yet.

5.2 The ablation grid

Seven arms, each adding one variable to the one before it. Every arm trains from the same warm start for the same number of additional steps on the same corpus.

Table 5. The pre-registered grid. V0 is the baseline, V6 is the full design. The relevance prior and the coordinate prior are separate arms rather than one, because the coordinate prior has a published null behind it and a combined arm would let a gain from one be credited to the other.

Arm	Attention	Relevance prior	Coordinate prior	Context	KV	Optimiser
V0	single path	off	off	off	grouped	AdamW
V1	three path	off	off	off	grouped	AdamW
V2	three path	on	off	off	grouped	AdamW
V3	three path	on	on	off	grouped	AdamW
V4	three path	on	on	on	grouped	AdamW
V5	three path	on	on	on	latent	AdamW
V6	three path	on	on	on	latent	second order

Target step counts are 2,000 additional steps for V1 through V5 and 5,000 for V6, on a target corpus of 50,000 loci. Those are plans, not commitments about what the runs will show.

5.3 Metrics

Quality metrics are the ones the masking study pre-registered, so that a T2 number and a T1-line number are the same measurement. Pooled corpus overlap on held-out bundles is the primary metric, computed on the generated text following the locked-anchor span. English ratio is a sanity metric and has to stay above 0.5. Anchor preservation has to stay at 1.0, because a design that buys quality by loosening a structural guarantee has not bought anything. Bootstrap intervals over 1,000 resamples must be non-overlapping with the baseline before any improvement is claimed.

Corpus overlap is a coarse proxy and the study that introduced it says so. It measures whether generated text uses words from the topical domain. It does not reward grammatical structure and it does not penalise correct topical tokens in an incoherent order. It is a necessary rather than a sufficient indicator, and it is used here because it is deterministic given fixed seeds,

needs no model-as-judge, and admits a straightforward bootstrap.

Efficiency metrics are seconds per step at matched batch and sequence length, peak memory, theoretical attention operation count from the profiler, and steps to a fixed loss target. Every one of them is recorded for V0 on the same card in the same session as the arm it is compared against, because a wall-clock comparison against a number measured on another day is not a comparison.

5.4 Statistical procedure

Bootstrap confidence intervals with 1,000 resamples on a held-out eval set of at least 50 queries, percentile method, reported as the point estimate and the interval together. Improvement claims require non-overlapping 95 percent intervals against the baseline arm. Permutation-test P values are reported alongside. Three seeds per arm, fixed in advance at 1234, 5678 and 9012.

5.5 Success criteria, fixed before any result is seen

These are thresholds, not predictions. Falling below one of them means the variant is not better than the baseline,

whatever the curves look like.

- **Quality.** The full design beats the baseline on pooled held-out corpus overlap with non-overlapping 95 percent bootstrap intervals.
- **Preservation.** The full design holds anchor preservation at or above 0.99 on a 100-query forced-anchor decode set.
- **Efficiency.** The full design reaches the baseline final loss in at most half the baseline wall-clock time on the same hardware.
- **Memory mechanism.** On a long-context eval requiring recall from more than 4,000 tokens back, the arm with lattice-compressed context beats the arm without it on non-overlapping intervals, on a recall metric specified before the eval set is built.

If any of the four fails, the failure is what gets published. The lab publishes its nulls, and section 3.2 of this document is one of them carried into a design it constrains.

5.6 Compute budget

Estimated at 10 to 15 GPU-hours for each of the five intermediate arms, 25 to 40 for the full design at 5,000 steps, and about 5 for the long-context evaluation, giving 80 to 120 GPU-hours in total, or three to five days of continuous run time on one consumer GPU. If the preconditioning stack delivers the target in section 4.5 that compresses to somewhere between one and one and a half days. These are estimates scaled from the one continued-pretraining run this line has actually measured, and the scaling is linear in steps and tokens, which is optimistic.

6. Results

Empty, and kept in the document so that a later revision fills it in place, with one subsection per arm carrying the training curve, held-out overlap against the baseline, bootstrap intervals, anchor preservation, and wall clock per step. Two arms carry an extra requirement. The arm that turns on the priors reports the trajectories of the two scalar gates, because if they stay near zero the model is declining to use the priors and that is the result. The arm that turns on lattice-compressed context reports the long-context recall benchmark against the arm below it, which is the only place the new mechanism can be seen on its own.

7. Scope, limitations and what this document does not establish

Every boundary on this design is collected here rather than distributed through the document, so that a reader takes the whole of it in one pass and weighs it against the whole of the rest.

7.1 Compute

One consumer GPU. A from-scratch T2 is not reachable in any useful timeframe, so every T2 arm is a warm start from a single-path checkpoint into a three-path architecture. That may simply not be a stable starting point. If the first arm degrades against the baseline, the warm-start strategy is

broken and the grid stops there rather than continuing to stack changes on a bad foundation.

7.2 Data

The corpus needed does not exist yet, and the corpus that does exist is about a fifth of the target size and was authored inside this lab. Held-out corpora authored by the researcher are weaker evidence than independently curated benchmark data, which the masking study also says about its own held-out sets. A T2 result on researcher-authored evals inherits that weakness in full.

7.3 Architecture

Three architectural extensions stacked on one another, plus an optimiser change and a quantisation change. Each extension has published precedent in some form; the composition has none that we can cite, and no stability precedent at all. This is why the grid is incremental and why each arm has to clear its own threshold. If the three-path block wins and the priors lose, the priors are dropped rather than carried along in the winning configuration.

7.4 The prior that has already failed once

The coordinate prior in section 3.2 proposes a use for the lattice coordinate after a published ablation found that coordinate empirically content-free as a conditioning signal under a fixed architecture. The design's answer is that the ablation did not give the model a learned path to use coordinates through, which is a real difference and also exactly the kind of reasoning that keeps a dead hypothesis alive. The mitigation is structural: the coordinate prior is its own arm, its gate trajectory is reported whatever the outcome, and a flat gate is reported as a second null rather than absorbed into a combined result.

7.5 Open questions that predate any run

- Whether the three coordinate embeddings should be added, which preserves head dimension, or concatenated, which needs a downstream projection.
- Whether the crystalliser should fire every fixed number of tokens or on an attention-cost trigger.
- What the long-context eval set should be, given that it has to be independent of the training data and does not exist yet.
- Whether the unconditional-generation degradation inherited from the warm-start checkpoint compounds under further anchor-heavy training, and what a curriculum that interleaves standard masked-diffusion phases would cost in anchor-extension quality.

7.6 What would falsify the design

Pre-registered. Any one of the following falsifies the thesis as written, and each is reported if it happens.

- The full design's pooled held-out corpus overlap sits within 5 percent of the baseline.
- The gates on the new paths stay near zero throughout training, meaning the model never learns to use them.

- Anchor preservation falls below 0.95, meaning a structural guarantee was traded for noise.
- Wall clock per step is worse than the baseline despite the whole preconditioning stack.

7.7 What this document does not establish

It establishes nothing about Cassandra T2, because Cassandra T2 does not exist. There are no weights, no checkpoints, no training curves and no evaluations. Every performance figure attached to T2 in this document is a target that names the threshold it has to clear, and a target that has not been tested is a statement about intent.

It does not establish that the composition is novel. The novelty claim is that we did not find a published system combining deterministic hash-based compression into a fixed lattice with an attention path that consumes both self-crystallised and user-supplied anchors. That is a claim about our search, which was ours, was done in 2026, and was not exhaustive. It should not be restated as T2 being the first or the only design of its kind. A reader who knows of published work that already does this is asked to write in.

It does not establish that the measured results in section 2 transfer to the proposed architecture. The 1.67x advantage is a fine-tuning result on one 1.3 billion parameter base checkpoint at one adapter rank, on seven corpora in one retrieval format, and nobody outside this lab has replicated it. The continued-pretraining result is one run, on 144 training pairs, evaluated on 12 held-out queries. Both were measured on the single-path architecture. A three-path architecture is a different model, and the argument that the inductive bias survives the change is an argument, not a result.

It does not establish that the efficiency stack works. The per-technique figures are published claims from other people's measurements on other models, and the product of five optimistic numbers is not an estimate. The only honest statement available today is the one in section 4.5: a target, a baseline, a metric and a piece of hardware to measure both on.

It does not establish anything about the retrieval format's quality relative to other retrieval approaches. LTMi-XT has not been benchmarked against alternatives. Its determinism is measured and its usefulness as a substrate for this design is assumed.

The author's prior, recorded before any result, is roughly a 30 percent chance that the full design clears all four success criteria. That is a stated expectation and not a measurement of anything. It is written down so that it cannot be revised upward after the fact.

8. Release plan

Released under Apache 2.0 regardless of the outcome: the three-path attention module, the priors implementation, the lattice-compressed context interface, all training and ablation scripts, and the runtime changes needed to pass relevance scores and coordinates to the model. Released under CC BY 4.0: this document, the specification updates covering the memory mechanism, and the decoding-specification extension covering self-crystallised memory.

Held internally: trained weights, if any run produces them, and the large derived training bundle, though the builder script for that bundle is open. Customer-specific adapters are not released, which is the same position the fine-tuning service takes.

Reproducibility commitments. Seeds are fixed in advance at 1234, 5678 and 9012. Eval sets used for any headline number are released under Apache 2.0 when that number is announced. Hyperparameters live in the released training scripts rather than in prose. Negative results are published with the same effort as positive ones, and section 3.2 of this revision is what that looks like in practice.

9. Conclusion

This document specifies a seven-arm ablation that tests whether retrieval-aware three-path attention, with lattice-compressed context on top of a preconditioning training stack, produces a measurable quality and efficiency improvement over the continued-pretraining checkpoint of the released Cassandra T1, on retrieval-grounded masked-diffusion language modelling.

What the design has behind it is a validated training objective with a causally tested mechanism, a decoding guarantee that holds by construction rather than by measurement, a deterministic retrieval substrate, and one published null about a component this design proposes to use again. Those four are the entire basis, and they are written down here alongside the thresholds that will judge the design against them. A reader should weigh the proposal against that basis rather than against the confidence of the prose.

References

- Behrouz, A., et al. (2024). Titans: Learning to Memorize at Test Time.
- Garren, T. (2026a). Anchor-Token Masking Produces an Out-of-Distribution Generalisation Advantage in Retrieval-Grounded Masked-Diffusion Language Models. SOPHIA XT LLC. DOI 10.5281/zenodo.20496783.
- Garren, T. (2026b). LTMi-XT v0.1: A Retrieval Format with Hash-Derived Topological Indexing. SOPHIA XT LLC specification.
- Garren, T. (2026c). Cassandra T1: open-released masked-diffusion language model. Apache 2.0. Architecture, checkpoint hashes and training trajectory published with the repository.
- Liu, H., et al. (2024). Sophia: A Scalable Stochastic Second-order Optimizer for Language Model Pre-training. ICLR 2024.
- Ma, S., et al. (2024). The Era of 1-bit LLMs: All Large Language Models are in 1.58 Bits. arXiv:2402.17764.
- Rae, J. W., et al. (2020). Compressive Transformers for Long-Range Sequence Modelling. ICLR 2020.
- Shah, J., et al. (2024). FlashAttention-3: Fast and Accurate Attention with Asynchrony and Low-precision. arXiv:2407.08608.
- Wu, Y., et al. (2022). Memorizing Transformers. ICLR 2022.
- Low-rank latent factorisation of key and value projections for attention, with on-the-fly decompression. arXiv:2405.04434 (2024) and arXiv:2412.19437 (2024).
- LTMi-XT empirical findings, 12 May 2026. Coordinate-scheme ablation and its adversarial review, published in the LTMi-XT repository.

Revision history

v0.1, 10 May 2026. Initial pre-registered design. Ablation grid and success criteria frozen. Results sections reserved.

v0.2, 19 September 2026. Corrections rather than new results. The intermediate model name v0.1 used throughout is removed; what that name referred to is written as what it is, a continued-pretraining checkpoint of the released Cassandra T1 reported in section 6.4a of the masking study, and the passages that read as announcements of a released model are rewritten. Every T2 performance figure is labelled a target at

the point of use and a status ledger is added at the front. The coordinate ablation published on 12 May 2026, two days after v0.1, is carried into section 3.2 as a constraint on the prior it tested, and the ablation grid is widened from six arms to seven so that the two priors are separable. The wall-clock parity framing against hardware this lab does not have is removed and replaced with a same-hardware target. The gate initialisation arithmetic is corrected. A checkpoint hash for an unreleased artifact, a vendor price estimate and a unit-test count that cannot be checked from a public surface are cut.