EVENT-GROUNDED RE-ENTRY: BOUNDARY-CONDITIONED TOP-STACK CONTROL FOR LLM WORKFLOWS

Anonymous authors
Paper under double-blind review

ABSTRACT

`VERIFY→COMMIT` is grammar-valid; without a qualifying verifier event for the latest candidate, the realized submission is nevertheless premature. Event-Grounded Re-Entry (EGR) uses that exact failure point as a computational synchronization boundary in one incremental decoder trajectory. A hard-masked transition head selects phase-specific rank-32 adapters; the committed canonical delimiter passes once through shared layers 1–28 and then through the selected layers 29–32, while prior KV entries remain immutable. On the disjoint 480-task ToolBench test, EGR keeps declared illegal transitions at zero, reduces invalid actions from 19.6% to 18.1% and commits lacking qualifying verifier evidence from 9.0% to 8.1%, and raises task success over FSM masking from 51.4% to 53.3% (paired difference +1.9 [+1.0, +2.7]). Against a 41.96M-parameter, budget-matched all-layer routed-LoRA control, EGR gains +1.1 points [+0.2, +2.0] at lower model-side latency. The FSM-matched success gains recur on HumanEval+ Verify (+1.8), a disjoint 240-issue SWE-bench extension (+2.8), and WebArena-180 (+2.6); replacing the selected `VERIFY` bank with the learned `ACT` bank raises the missing-evidence rate by 31.2 points [26.8, 35.6]. EGR runs at 1.05–1.06× model-side latency, compared with 1.38–1.43× for full-prefix stage restarts, making externally observed events useful control points for both workflow semantics and efficient neural computation.

1 INTRODUCTION

`VERIFY→COMMIT` appears in both a correct workflow trace and a failure trace. The distinguishing event is not the route label but whether the latest action has produced qualifying evidence before the final delimiter opens. With prompt-tagged Llama-3.1-8B-Instruct, 12.1% of ToolBench commits lack qualifying verifier evidence and 23.7% of actions are invalid in the state where they are issued. HumanEval+ Verify concentrates error differently: 21.8% of realized event transitions are invalid and 16.7% of commits lack qualifying test evidence. Across these traces, an API may be called before its arguments are ready, a patch may be submitted before test feedback arrives, or a commit delimiter may open in an evidence-incomplete state despite a plausible phase choice.

The control object is therefore larger than the phase sequence. We distinguish the *declared route*, which a finite-state mask can certify, from the *realized event trace*, whose actions and commits must be valid in the current environment state. Their discrepancy is the *route–realization gap*. A grammar mask closes illegal support at the next-phase decision, but it does not directly shape the token distribution that realizes a legal phase. The resulting systems question is where an observed workflow event should intervene: in serialized prompt text, route logits, hidden computation, or a new full-model invocation.

Event-Grounded Re-Entry (EGR) places that intervention at the opening delimiter of each phase. Ordinary next-token logits first propose a boundary marker, which is withheld from the prefix. A transition head reads the last committed token's layer-28 state, the workflow mask removes illegal phases, and the runtime selects one phase-specific top-stack adapter bank. Only then is the selected phase's canonical delimiter appended: it traverses shared layers 1–28 once and layers 29–32 under the selected bank. The associated span continues with that bank, historical KV entries never change,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and environment returns or test results enter through the same interface with routes fixed by their observed event type.

Two controls locate the empirical advantage. FSM-mask shares the detector, grammar, canonical delimiters, and single-trajectory cache, but changes within-phase computation only through the serialized delimiter. Masked Routed LoRA shares EGR's transition head, hard mask, 41.96M-parameter budget, training tokens, and mixed historical cache, while routing conventional low-rank adapters through every layer. On the disjoint 480-task ToolBench test, EGR reaches 53.3% TaskAcc versus 51.4% for FSM-mask and 52.2% for Masked Routed LoRA. The paired gains are +1.9 [+1.0, +2.7] and +1.1 [+0.2, +2.0], respectively. Declared illegal transitions are zero for all hard-masked methods; EGR separates them on invalid actions, evidence-incomplete commits, and end-task success.

The learned banks are functionally phase-specific. Substituting the `ACT` bank at `VERIFY` boundaries raises commits lacking qualifying evidence by 31.2 points [26.8, 35.6] and lowers TaskAcc by 4.1 points [-5.5, -2.7]. Removing the hard mask or route head raises realized invalid-transition rate by 3.4 or 2.5 points, whereas disabling KV reuse primarily moves latency from $1.05\times$ to $1.36\times$. These signatures connect route legality, phase-conditioned realization, and cache reuse to different measured outcomes.

The same quality–cost ordering appears across executable code, repository repair, and browser interaction. Relative to FSM-mask, EGR gains 1.8 TaskAcc points on HumanEval+ Verify, +2.8 on a disjoint 240-issue SWE-bench extension, and +2.6 on WebArena-180. It remains ahead under 10% phase-boundary corruption and on Qwen2.5-32B-Instruct. Across live ToolBench, SWE-bench, and WebArena services, its total-time point estimates stay within 0.5% of FSM-mask and save 680–1360 ms against full-prefix stage restarts.

The central contribution is a boundary-conditioned top-stack control architecture whose certified object (the declared route) and empirical object (the realized event trace) are evaluated separately. EGR combines masked route selection, phase-specific top-layer adaptation, canonical-delimiter dispatch, and explicit KV ownership in one decoder trajectory. Matched controller-only, all-layer routing, and restart controls—together with depth, component, module-substitution, verifier-collapse, ontology, corruption, model-family, and live-service studies—show when that integrated placement improves realized actions and task success after route legality and trainable capacity are matched.

2 Related Work

Tool-using systems such as Toolformer, ToolLLM, AnyTool, TaskWeaver, ReAct, and AgentBench study operation selection, execution, and agent evaluation (Schick et al., 2023; Qin et al., 2024; Du et al., 2024; Qiao et al., 2023; Yao et al., 2023; Liu et al., 2024b). StateFlow assigns phases to state-driven model invocations (Wu et al., 2024), and AgentLTL specifies procedural properties over completed traces or gates actions from trace prefixes (Elkoussy & Perez, 2026). EGR targets a complementary interface: an observed event chooses hidden computation inside one continuing decoder trajectory. Verifier-mediated generation and iterative refinement organize proposal and checking (Lightman et al., 2024; Shinn et al., 2023; Madaan et al., 2023); SWE-bench and WebArena supply the executable tests and browser events needed to judge whether an action is valid when issued (Jimenez et al., 2024; Zhou et al., 2024).

Constrained decoding projects token sequences onto lexical or grammatical support (Hokamp & Liu, 2017; Geng et al., 2023; Willard & Louf, 2023). XGrammar-2 adds tag-triggered grammar switching and cross-grammar cache reuse for agentic structured generation (Li et al., 2026a). These mechanisms determine which serialized phase tag or call is legal. EGR also measures the event induced after that choice: missing returns, malformed spans, state-invalid actions, and commits without qualifying evidence remain visible even when every declared transition is permitted. FSM-mask receives the identical detector, grammar, and canonical delimiter, making it the direct test of serialized route control versus additional hidden-state conditioning.

Adapters and low-rank updates specialize frozen models efficiently (Houlsby et al., 2019; Hu et al., 2022; Dettmers et al., 2023; Liu et al., 2024a). MixLoRA, HMoRA, and LoRA-Mixer route experts by token or task for multi-task adaptation (Li et al., 2024; Liao et al., 2025; Li et al., 2026b). Activated LoRA (aLoRA) shows that post-invocation adaptation can reuse a base-prefix cache (Greenewald et al., 2025). EGR couples this serving objective to external workflow semantics: a hard transition

grammar selects among phase banks, the canonical boundary token is the first token processed by the selected top stack, and lower layers remain route-independent. The budget-matched Masked Routed LoRA comparator retains EGR's route head, mask, training budget, and historical-cache reuse but routes adapters through all layers; Multi-call instead rebuilds the prefix at each stage. This comparison connects control placement to explicit ownership of historical and boundary-token KV entries (Pope et al., 2023).

3 Problem Formulation

We study event-delimited workflows whose traces expose planning text, tool or code emission, returned observations or test execution, and final submission. Realized-event control asks whether a legal control decision induces an event that is valid in the environment state where it occurs. The operational phase set is $\Phi = \{\texttt{PLAN}, \texttt{ACT}, \texttt{VERIFY}, \texttt{COMMIT}\}$. Every method receives identical phase labels and route grammar $\mathcal{T}_{\text{valid}} \subseteq \Phi^2$ derived from those events.

Two sequences describe a generated trajectory. The *declared route* $d = (d_0, \ldots, d_B)$ contains phase choices made by the model or controller at detected boundaries. The *realized event trace* $e = (e_0, \ldots, e_M)$ is reconstructed after generation from emitted tool calls, tool returns, test runs, and final submissions. Thus B and M are the numbers of adjacent edges in the respective sequences. The event alphabet augments Φ with typed `MISSING_RETURN`, `MALFORMED`, `TRUNCATED`, and `POST_COMMIT` sentinels. The state-conditioned relation $\mathcal{T}_{\text{event}}(q_m)$ is evaluated against the environment state immediately before event e_{m+1}; every edge incident to a sentinel and every commit without the workload's qualifying verifier predicate is invalid. A hard mask guarantees $(d_b, d_{b+1}) \in \mathcal{T}_{\text{valid}}$, while the generated segment and environment response determine which event enters e. We report declared illegal-transition rate (D-ITR), realized invalid-transition rate (R-ITR), and boundary disagreement:

$$\text{D-ITR}(d) = \frac{\sum_{b=0}^{B-1} \mathbb{1}[(d_b, d_{b+1}) \notin \mathcal{T}_{\text{valid}}]}{B}, \qquad B > 0, \tag{1}$$

$$\text{R-ITR}(e) = \frac{\sum_{m=0}^{M-1} \mathbb{1}[(e_m, e_{m+1}) \notin \mathcal{T}_{\text{event}}(q_m)]}{M}, \qquad M > 0, \tag{2}$$

$$\text{Disagree} = \frac{\sum_{b=1}^{B} \mathbb{1}[d_b \neq \text{align}(e)_b]}{\max(B, 1)}. \tag{3}$$

Legal self-loops and repeated `ACT`→`VERIFY`→`ACT` cycles remain in the denominators. Premature `COMMIT` and typed failure sentinels therefore enter R-ITR directly. After an `ACT`, the parser inserts `MISSING_RETURN` when the next boundary or termination arrives without an environment return; malformed and unclosed generated spans become `MALFORMED` and `TRUNCATED` at recovery and termination, respectively; events after the first valid commit become `POST_COMMIT`. Each sentinel is one event in the trace, and each adjacent invalid edge contributes once, so an interior sentinel can invalidate both its incoming and outgoing edges. Traces with no declared or realized edge have undefined per-trace D-ITR or R-ITR, are excluded from the corresponding seed-level edge ratio, and are counted as malformed.

For Equation 3, align is an order-preserving dynamic-programming alignment between declared phase boundaries and phase-valued realized events with unit insertion, deletion, and substitution cost; sentinel events cannot match a declared phase. Every unmatched boundary and every phase substitution contributes one disagreement. PhaseAcc, used in the corruption study, is the fraction of aligned, non-sentinel realized boundaries assigned their event-derived phase. IAR is the fraction of realized `ACT` events invalid in the pre-action state: ToolBench calls must parse, name an available API, and satisfy its argument schema; HumanEval+ candidates must parse and execute in the test harness; SWE-bench patches must parse and apply to the current repository state; and WebArena operations must be schema-valid and executable in the current browser state. VSR is the fraction of realized commits whose trace lacks the pre-specified qualifying-evidence predicate: a schema-valid return attached to the latest ToolBench action, parseable executed-test feedback for the latest HumanEval+ or SWE-bench candidate, or a completed state check attached to the latest WebArena action. In the VERIFY-collapsed control, executed-test feedback remains eligible without a separate `VERIFY` route. TaskAcc applies each workload's fixed endpoint predicate—task completion for ToolBench and WebArena, all required tests for HumanEval+ Verify, and issue-resolution tests for SWE-bench—and counts every failure stop as unsuccessful.

Control placement has two axes. The phase constraint may enter serialized input, next-route output support, hidden-state transformation, or the invocation graph; computation may follow one trajectory, enter a selected top stack, or restart the full backbone. Prompt tags occupy input/one-trajectory placement; FSM-mask occupies output support/one trajectory; Masked Routed LoRA combines output projection with all-layer hidden specialization; Multi-call uses invocation-graph control/full restart; EGR couples output projection to event-grounded top-stack dispatch. The route–realization formulation predicts equal D-ITR for hard-masked methods, small residual differences in realized-event R-ITR, and larger separations in IAR, VSR, and TaskAcc because those metrics depend on tokens generated within a legal phase.

Table 1: Prompt-tag phase-error profile on Llama-3.1-8B-Instruct. IAR, R-ITR, and VSR use action, event-edge, and commit denominators, respectively; lower is better.

Benchmark	IAR (%)	R-ITR (%)	VSR (%)
ToolBench	23.7	18.4	12.1
HumanEval+ Verify	15.3	21.8	16.7

Table 1 shows that workflow failure is multi-channel: ToolBench is action-heavy, whereas HumanEval+ Verify concentrates more error at transitions and commits. The separate denominators preserve this distinction. The complete declared-route and realized-event audit in Table 19 shows exactly zero D-ITR for every hard-masked method while R-ITR, disagreement, and malformed counts remain distinguishable.

4 Event-Grounded Re-Entry

EGR treats each phase-opening delimiter as a synchronization barrier between route selection and event realization. A decoder-only Transformer with L layers is divided into a shared lower trunk and a phase-specific top-K stack (Figure 1). For Llama-3.1-8B-Instruct, $L = 32$ and $K = 4$: layers 1–28 produce route-independent states for each committed token, and layers 29–32 apply the adapter bank selected for the current declared phase. The route head reads the last committed token's layer-28 state when the ordinary next-token distribution proposes an opening delimiter. The proposal is a trigger, not a committed token. After masking and route selection, the selected canonical delimiter advances through the shared trunk once and enters the selected top stack; its resulting KV entries are the first entries owned by the new phase.

Each re-entry layer $\ell \in \{L-K+1, \dots, L\}$ stores $|\Phi|$ low-rank adapter pairs $\{(\mathbf{A}^{\phi}_{\ell j}, \mathbf{B}^{\phi}_{\ell j})\}$ of rank $r = 32$ for attention projections $j \in \{q, k, v, o\}$ and MLP projections $j \in \{\text{gate}, \text{up}, \text{down}\}$:

$$\mathbf{h}^{\phi}_{\ell j} = \mathbf{W}_{\ell j}\mathbf{x} + \mathbf{B}^{\phi}_{\ell j}\mathbf{A}^{\phi}_{\ell j}\mathbf{x}. \tag{4}$$

The pretrained matrices remain frozen. If $\mathcal{J}$ is the projection set, the trainable count is

$$P_{\text{EGR}} = K|\Phi|r \sum_{j \in \mathcal{J}} (d^{\text{in}}_j + d^{\text{out}}_j) + d_{\text{model}}|\Phi| + |\Phi|. \tag{5}$$

Llama-3.1-8B uses $d_{\text{model}} = 4096$, key/value width 1024, and MLP width 14336. Across the seven attention and MLP projections, $\sum_{j \in \mathcal{J}} (d^{\text{in}}_j + d^{\text{out}}_j) = 81920$, yielding 41.96M trainable parameters, 0.52% of the backbone. The same formula produces the depth counts in Table 5.

At proposed boundary position b, classifier g_ψ reads the shared state $\mathbf{h}^{(b^-)}_{L-K}$ of the last committed prefix token; no representation of the proposed or canonical delimiter has yet been inserted:

$$p(d_{b+1} \mid \mathbf{h}^{(b^-)}_{L-K}, d_b) = \text{softmax}\Big(\text{mask}_{\mathcal{T}_{\text{valid}}(d_b)}(\mathbf{W}_g \mathbf{h}^{(b^-)}_{L-K} + \mathbf{b}_g)\Big). \tag{6}$$

The mask adds $-\infty$ outside valid support before route selection. The grammar admits phase self-loops, `PLAN`→`ACT`, `PLAN`→`COMMIT`, `ACT`→`VERIFY`, `VERIFY`→`ACT`, and `VERIFY`→`COMMIT`. Ties follow the fixed order `ACT`, `VERIFY`, `COMMIT`, `PLAN`. Decoding is greedy with temperature 0, a 4096-token output cap, and at most eight action–verification cycles.

Training minimizes

$$\mathcal{L} = \mathcal{L}_{\text{token}} + 0.25\,\mathcal{L}_{\text{route}}, \tag{7}$$

where token likelihood uses the teacher-forced event-derived phase for each span and route cross-entropy is evaluated at phase-opening boundaries after the same hard support mask used at inference. Only the phase adapters and g_ψ are optimized; backbone and embeddings remain frozen.

Cache ownership is token-incremental. Layers 1–28 compute one new KV entry for each committed token and never recompute the prefix at a phase boundary. Completed top-stack KV entries retain the adapter identity under which their tokens were emitted and remain immutable context, so route history is represented in the mixed historical cache. The withheld proposal creates no KV entry. Once the route is selected, the canonical opening delimiter is computed exactly once: shared lower-layer KV is appended, its layer-28 output is dispatched to the selected layers-29–32 bank, and the resulting top-stack KV is appended under that bank. Subsequent body tokens follow the same lower-trunk/selected-top-stack path. Masked Routed LoRA also retains a mixed historical cache but switches adapters in every layer for the canonical delimiter and subsequent tokens; Multi-call owns no cross-stage cache and rebuilds all layers from the serialized full trace. We use *re-entry* for the boundary-conditioned dispatch from the continuing shared trajectory into a newly selected top stack, not for a second computation of the canonical token.

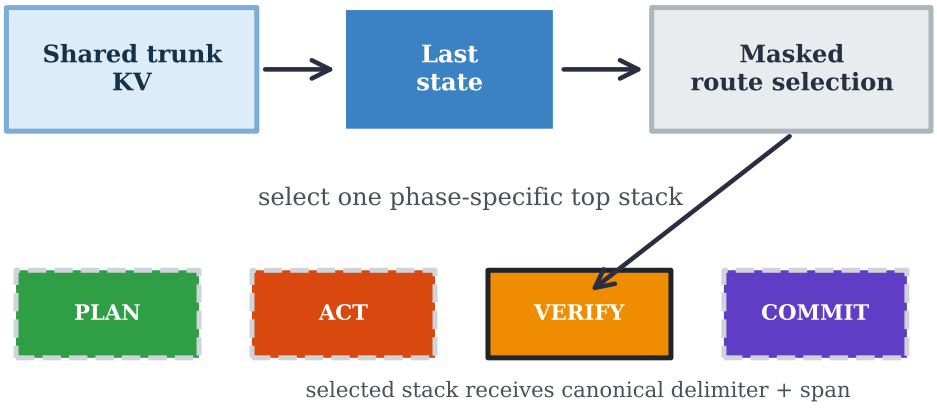

Figure 1: Boundary-synchronized EGR computation. A proposed opening marker triggers route selection but is not committed. The canonical delimiter then extends shared layers 1–28 once and enters layers 29–32 under the selected adapter bank; all historical KV entries remain immutable.

Figure 1 separates the two control operations: support projection certifies the declared route, while selected top-stack dispatch conditions the boundary token and realized span. Phase-specific work stays in the top stack, and lower-layer evidence is reused rather than rebuilt.

4.1 Unified Boundary and Execution Semantics

All methods use the same detector, parser, environment interface, and stopping rule. The next-token logits of the last committed token may propose an opening marker; the detector withholds that token before any KV insertion or environment exposure. Prompt tags commit the proposed marker directly and derive the declared phase from it. Hard-masked methods use the proposal only to trigger routing: FSM-mask and Multi-call select among legal canonical delimiters using masked base-model delimiter logits, whereas Masked Routed LoRA and EGR use the masked transition-head logits in Equation 6. The raw proposal is discarded, exactly one canonical delimiter is appended, and its hidden state is computed under the selected method. A proposal–route mismatch therefore cannot leave contradictory tags or stale delimiter KV in the prefix. In FSM-mask, the canonical delimiter is the only mechanism by which the route decision conditions subsequent tokens.

Table 2: Operational semantics of matched controls. MRL denotes Masked Routed LoRA and matches EGR's route constraint and trainable budget. Parameter counts are for Llama-3.1-8B.

Method	Route logits	Mask	New-span conditioning	Cache	Params (M)
Prompt tags	raw delimiter	no	serialized tag	shared	0.00
FSM-mask	base delimiter	yes	canonical delimiter	shared	0.00
MRL	transition head	yes	all-layer routed LoRA	mixed	41.96
Multi-call	base delimiter	yes	stage prompt + full trace	restart	0.00
EGR	transition head	yes	selected top-K stack	mixed	41.96

Environment returns and test results are exogenous typed events rather than model route choices. After a well-formed action closes, the runtime executes it, appends the returned body under its canonical delimiter, and forces the corresponding legal successor phase. A malformed or rejected action is not executed and instead yields the typed parser/environment event used by R-ITR and IAR. Only a parsed, closed `<final>` event terminates successfully; selecting `COMMIT` without such an event does not stop decoding, and any content after the first valid final is marked `POST_COMMIT`. Token cap and cycle cap are failure stops shared by every method.

Algorithm 1 Unified inference protocol; method-specific routing and cache update

Require: Method m, prefix x, grammar $\mathcal{T}_{\text{valid}}$, parser, environment

1: Initialize declared trace $d \leftarrow [\texttt{PLAN}]$, event log $z \leftarrow []$, and m's cache
2: Decode next-token logits until an opening marker is proposed; withhold it before KV insertion
3: Select d_{b+1} from the proposal (Prompt tags), masked base delimiter logits (FSM-mask/Multi-call), or masked g_ψ logits (Masked Routed LoRA/EGR)
4: For hard-masked methods, discard the raw proposal and append the canonical delimiter for d_{b+1}
5: Compute that delimiter under m: continue, switch all-layer bank, restart from the full trace, or dispatch EGR's layer-28 output into the selected top-K bank
6: Decode the closed span; execute only parser-valid actions and append typed environment returns to z
7: Preserve m's immutable historical KV entries; Multi-call rebuilds its stage cache
8: Stop successfully only on a valid closed `<final>`; otherwise apply shared token/cycle failure stops
9: Parse z into realized trace e and pre-edge states $(q_0, \ldots, q_{M-1})$, inserting typed sentinels
10: Return output, d, e, and the states used by $\mathcal{T}_{\text{event}}$

5 Experiments

5.1 Settings, Comparators, and Protocol

The experiment sequence tests three links in the proposed control path: hidden-state conditioning after route legality is fixed (RQ1), transfer across event and boundary regimes (RQ2), and the contribution of route selection, phase banks, re-entry depth, and cache ownership to the quality–latency frontier (RQ3).

We evaluate Llama-3.1-8B-Instruct (Grattafiori et al., 2024) on four multi-step event-delimited workflow families. ToolBench (Qin et al., 2024) has a fixed 200-task development slice used for re-entry-depth selection and a disjoint 480-task endpoint test. HumanEval+ Verify converts all 164 HumanEval problems (Chen et al., 2021) into candidate-code, test-execution, inspection, and final-program stages; the VERIFY-collapsed study reruns inference with the same selected checkpoints. SWE-bench (Jimenez et al., 2024) contains a 120-issue, three-repository anchor and a disjoint

Table 3: Disjoint 480-task ToolBench endpoint test. IAR, R-ITR, VSR (commits lacking qualifying verifier evidence), and TaskAcc are percentages; latency is normalized model-side generation time. MRL denotes Masked Routed LoRA.

Method	IAR ↓	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
Prompt tags	24.1	18.9	12.8	47.6	1.00×
FSM-mask	19.6	13.7	9.0	51.4	1.06×
MRL	18.8	13.4	8.6	52.2	1.07×
Multi-call	19.3	14.8	9.1	52.5	1.41×
EGR	18.1	13.2	8.1	53.3	1.05×

endpoint extension of 240 issues across 9 repositories. WebArena's 812 tasks (Zhou et al., 2024) are fixed as 542 phase-supervised training, 90 checkpoint-selection, and 180 endpoint-test tasks stratified across 6 sites. No endpoint test determines K, hyperparameters, or checkpoint choice. BFCL v3's 1,000-example non-live AST subset supplies a single-turn low-boundary control in Table 37 (Patil et al., 2025); Qwen2.5-7B- and 32B-Instruct test model-family and scale transfer (Yang et al., 2024).

Within each workload and backbone, the learned comparators receive identical event-derived labels, training tokens, optimization steps, and selection criteria. Every method uses the same output schema, context limit, parser, environment, stopping rule, and matched run identifiers (17, 29, 41). These identifiers index complete training/evaluation replicates and shared environment initialization; decoding within a replicate is greedy rather than token-sampled. Prompt tags serialize the model's unmasked phase proposal. FSM-mask projects base-model delimiter logits onto legal support. Masked Routed LoRA uses EGR's transition head and hard mask with conventional all-layer routing; rank 4 on 32-layer Llama and rank 2 on 64-layer Qwen match EGR's model-specific trainable-parameter and optimizer budgets. Multi-call uses the frozen checkpoint in separate full-prefix stage invocations. EGR uses top-four rank-32 re-entry. Section 4.1 fixes mismatch resolution, token-level cache ownership, parser behavior, and stopping semantics for all five methods.

Learned controllers train for 3 epochs on 8×A100-80GB GPUs with CUDA 12.4, bf16, AdamW, batch size 16, weight decay 0.01, and 3% linear warm-up. Adapter and transition-head learning rates are 2×10^{-4} and 5×10^{-5}. Greedy decoding uses a 4096-token output budget and batch size one for model-side timing. We discard 20 warm-up tasks and average five timing repetitions per task. Live service measurements use concurrency 1 and 8 with identical tool services, browser containers, and repository test images for every method. All reported evaluation manifests are fixed across methods; the task, event, action, verifier, and commit denominators appear with the corresponding appendix tables.

The task is the paired resampling cluster: each sampled task carries all three run outcomes for both methods. Cells with mean±standard deviation summarize run-level estimates; unadorned rates are pre-specified run means, and integer counts are pooled task–run micro totals. TaskAcc and its paired differences are equal-weight macro-averages over fixed workload strata (ToolBench API family, HumanEval test-count quartile, SWE-bench repository, WebArena site, and BFCL category). Where appendix stratum rows are shown, their equal-weight averages reproduce the reported TaskAcc values after display rounding. Wins/losses/ties count paired task–run outcomes, so wins minus losses equals the difference in pooled success counts. Difference intervals come from 10,000 paired task-cluster bootstrap resamples that retain the three runs and preserve workload strata; tables report 95% estimation intervals. Normalized model latency is each method's mean generation time per emitted token divided by the prompt-tag mean on the same workload after 20 warm-up tasks and five repetitions; live service time additionally includes network/tool, browser, and test execution.

5.2 RQ1: Realized-Event Control with Route Legality Fixed

Table 3 makes the held-out comparison primary. Every hard-masked method has zero D-ITR (Table 19), so the remaining differences concern realized behavior. Relative to FSM-mask, EGR reduces IAR by 1.5 points [0.9, 2.1], VSR by 0.9 [0.4, 1.3], and raises TaskAcc by +1.9 [+1.0, +2.7]. Against budget-matched MRL, EGR gains +1.1 [+0.2, +2.0] while lowering IAR by 0.7 points. R-ITR differs less because all learned controls already share a legal route; the clearer separation in

Table 4: Paired EGR improvements across endpoint tests. IAR, VSR, and the first TaskAcc column compare with FSM-mask; the final column compares TaskAcc with budget-matched MRL. Brackets are task-cluster 95% intervals.

Setting	ΔIAR vs. FSM	ΔVSR vs. FSM	ΔTaskAcc vs. FSM	ΔTaskAcc vs. MRL
ToolBench	-1.5 [-2.1, -0.9]	-0.9 [-1.3, -0.4]	+1.9 [+1.0, +2.7]	+1.1 [+0.2, +2.0]
HumanEval+	-1.2 [-1.8, -0.6]	-0.8 [-1.4, -0.2]	+1.8 [+0.6, +2.9]	+1.1 [+0.1, +2.1]
SWE-240	-1.6 [-2.3, -0.9]	-1.4 [-2.1, -0.7]	+2.8 [+1.3, +4.2]	+1.6 [+0.5, +2.7]
WebArena	-1.6 [-2.3, -0.9]	-1.4 [-2.0, -0.7]	+2.6 [+1.0, +4.1]	+1.7 [+0.5, +2.9]

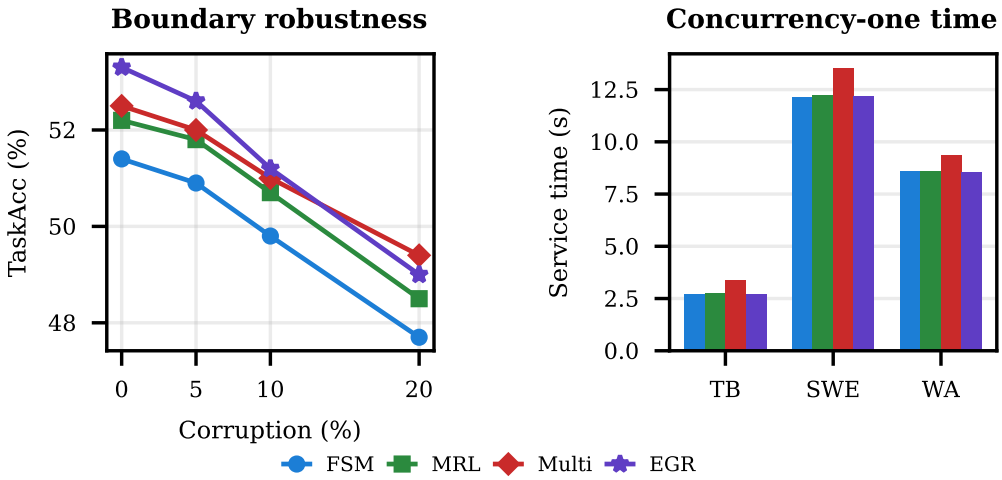

Figure 2: Boundary robustness and live service cost. Left: TaskAcc under identical phase-boundary corruption on the ToolBench test extension; EGR retains the strongest frontier through 10% corruption. Right: live concurrency-one total service time on ToolBench-200 development, SWE-bench-120, and WebArena-180; EGR stays near single-trajectory controls and avoids Multi-call restart cost. Tables 30 and 36 report paired 95% intervals.

action validity, qualifying-evidence availability, and endpoint success matches the within-phase target. Tables 5, 6, and 8 report depth selection, component isolation, and complete paired intervals.

5.3 RQ2: Transfer Across Workflows and Boundaries

Table 4 shows the same endpoint pattern in all four event-rich tests: EGR improves TaskAcc over both grammar-only control and budget-matched all-layer routing, while the FSM comparison also lowers invalid actions and evidence-incomplete commits. Realized-event R-ITR and boundary disagreement move less after masking has fixed declared legality (Table 20), which localizes the larger gain to behavior inside the selected phase. The disjoint 240-issue SWE-bench extension is the strongest transfer result: EGR reaches 30.3%, compared with 27.5% for FSM-mask, 28.7% for MRL, and 29.4% for Multi-call. Its EGR–FSM paired gain is +2.8 [+1.3, +4.2], with nonnegative repository-level differences across all nine projects. Multi-call remains the closest restart-based quality control but requires 1.38–1.43× model-side latency.

Boundary quality modulates the value of event synchronization. On extended ToolBench, EGR TaskAcc moves from 53.3% with clean labels to 52.6% at 5% corruption, 51.2% at 10%, and 49.0% at 20%. Its paired margin over FSM-mask remains +1.7, +1.4, and +1.3 points, respectively. Gradual margin contraction tracks the decreasing informativeness of the shared boundary signal.

Live timing confirms that model-side savings survive environment execution. At concurrency one, EGR/FSM-mask/Multi-call total service times are 2705/2695/3385 ms on ToolBench, 12175/12115/13535 ms on SWE-bench, and 8540/8570/9340 ms on WebArena. EGR's paired total-time deltas versus FSM-mask include zero, while its savings versus Multi-call are 680, 1360, and 800 ms. Tables 33, 34, and 36 report tails, throughput, concurrency eight, and paired intervals; Figure 4 reports replay decomposition.

5.4 RQ3: Re-Entry Mechanism, Scale, and Cost

The route-module substitution is the sharpest functional test. Replacing only the selected `VERIFY` bank with the learned `ACT` bank raises VSR by 31.2 points [26.8, 35.6] and lowers TaskAcc by 4.1 points [-5.5, -2.7], while its IAR shift is 2.3 points [1.2, 3.4]. Substituting `PLAN` for `ACT` instead raises IAR by 14.7 points and lowers TaskAcc by 8.3, demonstrating bank-specific disruption rather than a generic adapter failure (Table 17). Component removal separates the other roles: removing the hard mask raises ToolBench R-ITR from 13.4% to 16.8, removing the route head raises it to 15.9, and disabling KV reuse moves latency from 1.05× to 1.36× while retaining a 53.5% TaskAcc point estimate. The depth sweep selects $K = 4$: moving from one to four re-entry layers gains 4.1 TaskAcc points, whereas $K = 8$ doubles trainable capacity and reaches 53.4% rather than 53.8% (Table 5).

Model transfer preserves this separation of roles. On Qwen2.5-7B, EGR improves IAR/R-ITR/VSR/TaskAcc over FSM-mask by 1.3/0.9/1.1/1.8 points at nearly identical latency. On Qwen2.5-32B, the TaskAcc gains are +1.8 points on ToolBench and +2.0 on SWE-bench-120, with the lowest IAR, R-ITR, and VSR among the evaluated controls. Table 31 reports the complete error bundle, resource counts, and paired intervals.

6 Discussion

The system-design consequence is a division of responsibility at the event boundary. A hard mask certifies which phase may follow; selected hidden computation shapes the action emitted inside that phase; immutable historical KV keeps the intervention compatible with incremental serving. This division matters when success depends on state-valid actions, returned evidence, and commit timing, as in the four multi-step workflow families studied here. Tables 38–39 connect each protected object to its guarantee, matched comparator, and endpoint evidence. EGR controls workflow realization, while permissioning, sandboxing, output validation, and downstream guardrails continue to govern the capabilities of emitted actions.

The matched endpoint evidence supports EGR as an integrated placement of masked routing, phase-specific top-stack adaptation, canonical-delimiter dispatch, and cache ownership. The all-layer MRL control supplies a systems-level counterfactual in which adapter placement and lower-layer cache composition form a coupled alternative ownership policy; component removal and bank substitution then localize route, specialization, and reuse functions within EGR. The positive decision regime is stable across a coarser phase ontology, disjoint repositories, six WebArena sites, two model families, and 10% boundary corruption. Margins narrow when boundaries become sparse or less informative: BFCL yields a 0.5-point gain over MRL, and the quality frontier converges under 20% or heuristic corruption. VSR records qualifying evidence in the realized trace rather than causal use of that evidence. Together these results favor EGR for stateful workflows with reliable, repeated event boundaries; FSM masking remains the efficient choice when within-phase error is already low.

7 Conclusion

An external event delimiter can be a computational switch rather than only a serialized tag. EGR selects a legal phase from the last committed trunk state, sends its canonical delimiter through that phase's top stack, and preserves historical KV. On four endpoint tests it gains 1.8–2.8 TaskAcc points over grammar-only control, including +1.9 [+1.0, +2.7] on disjoint ToolBench and +2.8 [+1.3, +4.2] on disjoint SWE-bench. It exceeds budget-matched all-layer routing at 1.05–1.06× model-side latency while avoiding full-prefix restart. The cross-workflow, module-substitution, corruption, scale, and live-service results show that symbolic route constraints and neural span generation can be synchronized at observable environment events.

AI USE STATEMENT

Generative language tools assisted with manuscript editing, presentation, and submission-format preparation.

REFERENCES

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. doi: 10.48550/arXiv.2107.03374. URL https://arxiv.org/abs/2107.03374.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. *Advances in Neural Information Processing Systems*, 36:10088–10115, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html.

Yu Du, Fangyun Wei, and Hongyang Zhang. AnyTool: Self-reflective, hierarchical agents for large-scale API calls. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11812–11829. PMLR, 2024. URL https://proceedings.mlr.press/v235/du24h.html.

Laïla Elkoussy and Julien Perez. AgentLTL: A trace-verification framework for measuring, enforcing, and training procedural compliance in tool-using LLM agents. *arXiv preprint arXiv:2607.02599*, 2026. doi: 10.48550/arXiv.2607.02599. URL https://arxiv.org/abs/2607.02599.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-constrained decoding for structured NLP tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 10932–10952, 2023. doi: 10.18653/v1/2023.emnlp-main.674. URL https://aclanthology.org/2023.emnlp-main.674/.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. doi: 10.48550/arXiv.2407.21783. URL https://arxiv.org/abs/2407.21783.

Kristjan Greenewald, Luis A. Lastras, Thomas Parnell, Vraj Shah, Lucian Popa, Giulio Zizzo, Chulaka Gunasekara, Ambrish Rawat, and David Cox. Activated LoRA: Fine-tuned LLMs for intrinsics. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4d0b6303d4a4811445f69f357bf6def5-Abstract-Conference.html.

Chris Hokamp and Qun Liu. Lexically constrained decoding for sequence generation using grid beam search. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1535–1546, 2017. doi: 10.18653/v1/P17-1141. URL https://aclanthology.org/P17-1141/.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Dengchun Li, Yingzi Ma, Naizheng Wang, Zhengmao Ye, Zhiyuan Cheng, Yinghao Tang, Yan Zhang, Lei Duan, Jie Zuo, Cal Yang, and Mingjie Tang. MixLoRA: Enhancing large language models fine-tuning with LoRA-based mixture of experts. *arXiv preprint arXiv:2404.15159*, 2024. doi: 10.48550/arXiv.2404.15159. URL https://arxiv.org/abs/2404.15159.

Linzhang Li, Yixin Dong, Guanjie Wang, Ziyi Xu, Alexander Jiang, and Tianqi Chen. XGrammar-2: Dynamic and efficient structured generation engine for agentic LLMs. In *Proceedings of the ACM Conference on AI and Agentic Systems*, pp. 1009–1022. Association for Computing Machinery, 2026a. doi: 10.1145/3786335.3813124. URL https://doi.org/10.1145/3786335.3813124.

Wenbing Li, Zikai Song, Hang Zhou, Junqing Yu, Yunyao Zhang, and Wei Yang. LoRA-Mixer: Coordinate modular LoRA experts through serial attention routing. In *International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=GMP1S4R6Ke.

Mengqi Liao, Wei Chen, Junfeng Shen, Shengnan Guo, and Huaiyu Wan. HMoRA: Making LLMs more effective with hierarchical mixture of LoRA experts. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=lTkHiXeuDl.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32100–32121. PMLR, 2024a. URL https://proceedings.mlr.press/v235/liu24bn.html.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. Agentbench: Evaluating LLMs as agents. In *International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=zAdUB0aCTQ.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46534–46594, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392. PMLR, 2025. URL https://proceedings.mlr.press/v267/patil25a.html.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. In *Proceedings of the Sixth Conference on Machine Learning and Systems*, volume 5, pp. 606–624, 2023. URL https://proceedings.mlsys.org/paper_files/paper/2023/hash/c4be71ab8d24cdfb45e3d06dbfca2780-Abstract-mlsys2023.html.

Bo Qiao, Liqun Li, Xu Zhang, Shilin He, Yu Kang, Chaoyun Zhang, Fangkai Yang, Hang Dong, Jue Zhang, Lu Wang, Minghua Ma, Pu Zhao, Si Qin, Xiaoting Qin, Chao Du, Yong Xu, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. TaskWeaver: A code-first agent framework. *arXiv preprint arXiv:2311.17541*, 2023. doi: 10.48550/arXiv.2311.17541. URL https://arxiv.org/abs/2311.17541.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dHng2O0Jjr.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Brandon T. Willard and Rémi Louf. Efficient guided generation for large language models. *arXiv preprint arXiv:2307.09702*, 2023. doi: 10.48550/arXiv.2307.09702. URL https://arxiv.org/abs/2307.09702.

Yiran Wu, Tianwei Yue, Shaokun Zhang, Chi Wang, and Qingyun Wu. StateFlow: Enhancing LLM task-solving through state-driven workflows. In *Proceedings of the First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=3nTbuygoop.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. doi: 10.48550/arXiv.2412.15115. URL https://arxiv.org/abs/2412.15115.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. WebArena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=zlsj9akpaa.

A Additional Experimental Details

A.1 Re-entry depth and component isolation

Table 5: Effect of re-entry stack depth K on Llama-3.1-8B ToolBench. Added params are the total trainable EGR budget under the same $r = 32$, $|\Phi| = 4$ configuration; the sweep reports the quality–cost frontier.

K	Added params	IAR ↓	R-ITR ↓	TaskAcc ↑	Latency
1	10.50M	21.3	17.1	49.7	1.01×
2	20.99M	19.4	15.3	51.8	1.03×
4 (ours)	41.96M	17.9	13.4	53.8	1.05×
8	83.90M	17.6	13.1	53.4	1.09×

Table 5 makes $K = 4$ the selected efficiency point among the tested depths. Increasing depth from one to four layers improves TaskAcc by 4.1 points, whereas $K = 8$ doubles the parameter budget and slightly lowers TaskAcc despite a small phase-error reduction.

Table 6: Component isolation on the 200-task ToolBench development slice. Each row removes one EGR control element under the same labels, backbone, and evaluation harness.

Variant	IAR ↓	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
EGR	17.9	13.4	8.2	53.8	1.05×
EGR − hard mask	19.6	16.8	10.9	51.8	1.05×
EGR − transition head	19.1	15.9	9.9	52.4	1.04×
EGR − KV reuse	18.0	13.6	8.4	53.5	1.36×

Removing either masking or transition prediction degrades the phase-fidelity metrics in this trained system. The cache intervention has a different signature: it leaves the quality point estimates nearly unchanged but moves model-side latency toward repeated staging.

Table 7: Pooled audit counts for the ToolBench endpoint test. Rate tables report run means; this table reports denominators across the same three runs.

Method	n_{success}	n_{ACT}	n_{trans}	n_{COMMIT}
Prompt tags	686	3576	4888	1440
FSM-mask	740	3487	4744	1440
MRL	752	3472	4728	1440
Multi-call	756	3448	4709	1440
EGR (ours)	768	3461	4718	1440

Every method is evaluated on 1440 task–seed pairs. Comparable action, transition, and commit volumes show that the frontier is measured over equally substantive trajectories, while success counts preserve the ordering of the seed-mean TaskAcc results.

Table 8: Paired interval summary on extended ToolBench. Each cell reports EGR minus baseline with a task-cluster 95% interval.

Baseline	ΔIAR	ΔR-ITR	ΔVSR	ΔTaskAcc
FSM-mask	-1.5 [-2.1, -0.9]	-0.5 [-0.8, -0.2]	-0.9 [-1.3, -0.4]	+1.9 [+1.0, +2.7]
MRL	-0.7 [-1.3, -0.1]	-0.2 [-0.6, +0.2]	-0.5 [-1.0, 0.0]	+1.1 [+0.2, +2.0]
Multi-call	-1.2 [-1.8, -0.6]	-1.6 [-2.0, -1.1]	-1.0 [-1.4, -0.5]	+0.8 [+0.1, +1.5]

The smallest residual EGR–FSM difference is R-ITR after both methods share the legality mask; IAR, VSR, and TaskAcc retain intervals on the favorable side of zero. Against Multi-call, EGR improves all three phase metrics and TaskAcc while avoiding repeated full-prefix computation.

Table 9: Alternative phase ontology on ToolBench. This is offline re-adjudication of the same generated traces after merging pre-action planning and post-observation deliberation into one state; no method is rerun. Alt-TaskAcc applies the alternative parser's trace-valid success predicate, and latency is copied from the original runs.

Method	IAR ↓	R-ITR ↓	VSR ↓	Alt-TaskAcc ↑	Latency
Prompt tags	21.8	16.1	11.4	47.9	1.00×
FSM-mask	18.7	12.9	8.8	51.0	1.06×
EGR (ours)	17.2	12.4	7.9	52.9	1.05×

Under the alternative parser, IAR, R-ITR, VSR, and trace-valid task acceptance are recomputed from the same events; latency is carried over from the original runs. The coarser partition contracts absolute margins, yet EGR retains the best action, transition, verifier-evidence, and accepted-task row. This complements the corruption curves with a structurally different event partition.

Table 10: Trace-complete Qwen2.5-7B-Instruct ToolBench evaluation with the full action, realized-transition, verification, task-success, and latency bundle.

Method	IAR ↓	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
Prompt tags	22.0	17.5	12.0	49.9	1.00×
FSM-mask	18.2	13.7	9.1	53.6	1.06×
EGR (ours)	16.9	12.8	8.0	55.4	1.05×

Qwen2.5-7B reproduces the Llama ordering under complete traces: EGR improves over FSM-mask by 1.3 IAR, 0.9 R-ITR, 1.1 VSR, and 1.8 TaskAcc points while running at nearly identical model-side latency.

Table 11: Verifier-stage dependence on HumanEval+ Verify. "VERIFY-collapsed" is a new inference run with the same trained checkpoints: executed-test feedback is inserted inside the active `ACT` span, `VERIFY` is masked from route support, and no inspection delimiter is serialized. The feedback remains eligible evidence for VSR.

Setting	Method	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
Visible	Prompt tags	21.8	16.7	61.2	1.00×
	FSM-mask	16.0	12.1	65.6	1.06×
	EGR (ours)	15.8	11.3	67.4	1.05×
Collapsed	Prompt tags	17.1	6.2	63.0	1.00×
	FSM-mask	14.8	5.8	64.2	1.06×
	EGR (ours)	14.6	5.4	65.1	1.05×

Across 492 task–seed pairs per setting, EGR's TaskAcc margin is larger with a distinct verifier-visible boundary. The collapsed route grammar permits `PLAN`→`ACT`, `ACT`→`ACT`, and `ACT`→`COMMIT`; adapter rank, parameter allocation, output cap, cycle cap, parser, and valid-`<final>` stopping rule are unchanged. Under those controls, EGR's margin over FSM-mask contracts from 1.8 points [0.6, 2.9] to 0.9 [0.0, 1.8], identifying the explicit verifier boundary as the regime with the larger measured separation.

The public SWE-bench anchor contains 120 issues from three repositories with runnable tests and 360 task–seed pairs. Issue analysis maps to `PLAN`, patch emission to `ACT`, executed tests and inspection to `VERIFY`, and the submitted patch to `COMMIT`. Methods receive identical context, one revision after the first test execution, and common stopping rules.

The executable code-repair workflow preserves the same frontier: EGR reaches 31.4% TaskAcc, 2.6 points above FSM-mask and 1.3 above Multi-call, with the lowest joint phase-error profile at 1.06× model-side latency.

Table 12: Public second-workload frontier on a fixed SWE-bench code-repair slice. Metrics use the same four-phase audit vocabulary as the main paper, but the workload is repository-level bug repair with executable tests rather than ToolBench tool use. Latency remains normalized model-side runtime under the shared sequential harness so that the public-workload comparison is directly comparable to the main-paper frontier.

Method	IAR ↓	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
Prompt tags	19.8	24.1	18.6	24.7	1.00×
FSM-mask	16.7	15.2	13.5	28.8	1.05×
MRL	15.9	14.9	12.7	30.2	1.07×
Multi-call	16.0	16.4	12.9	30.1	1.39×
EGR (ours)	15.1	14.6	12.0	31.4	1.06×

Table 13: Audit counts and paired deltas for the public SWE-bench slice. Top: pooled raw counts over the same 360 task–seed pairs. Bottom: task-cluster signed differences "EGR minus baseline" with paired 95% intervals.

Method	Success	Tasks	ACT	Trans.	VERIFY	COMMIT
Prompt tags	89	360	842	1178	401	360
FSM-mask	104	360	816	1089	390	360
MRL	109	360	812	1093	392	360
Multi-call	108	360	804	1096	394	360
EGR (ours)	113	360	810	1091	397	360

Paired deltas: EGR minus baseline

Baseline	ΔIAR	ΔR-ITR	ΔVSR
FSM-mask	-1.5 [-2.2, -0.8]	-0.6 [-1.1, -0.1]	-1.5 [-2.1, -0.8]
MRL	-0.8 [-1.5, -0.1]	-0.3 [-0.8, +0.2]	-0.7 [-1.4, 0.0]
Multi-call	-0.9 [-1.5, -0.2]	-1.8 [-2.4, -1.2]	-0.9 [-1.5, -0.2]

Baseline	ΔTaskAcc
FSM-mask	+2.6 [+1.1, +4.0]
MRL	+1.2 [+0.2, +2.2]
Multi-call	+1.3 [+0.1, +2.5]

The comparable event volumes rule out shorter traces as the source of EGR's gain; paired differences instead place its clearest advantage in valid actions, qualifying verifier evidence, and completed repairs.

Comparable action, transition, verification, and commit volumes support the rate comparison in Table 12. After legality is matched, EGR's smallest residual difference remains R-ITR, while IAR, VSR, and TaskAcc carry the clearer internal-control advantage.

The main paper reports normalized model-side latency because it isolates repeated backbone recomputation cleanly. Figure 4 and Table 14 add a controlled replay view of measured external delays on the same fixed manifests under two concrete regimes: the 600 ToolBench task-seed pairs with recorded tool-call traces, and the 360 public SWE-bench task-seed pairs with recorded repository test execution. In both settings, model-side times are measured on the same 8×A100-80GB host as the main paper, while external times are replayed from 600 ToolBench traces and 360 SWE-bench traces so that the comparison attributes service-time differences to model recomputation rather than to changing network or test-lab conditions.

Figure 4 separates model computation from recorded external delay. Multi-call recomputation remains visible in ToolBench total time; slower SWE-bench tests compress the EGR–FSM difference while preserving EGR's large advantage over repeated staging.

Table 14: Replay-based service-time frontier supporting the live measurements in Table 33. External durations are replayed from recorded traces; model durations are measured on the common A100 host.

Setting	Method	TaskAcc ↑	Model ms	External ms	Total ms	p50 total	p90 total
ToolBench	Prompt tags	48.3	1840	640	2480	2210	3720
	FSM-mask	52.3	1920	640	2560	2290	3840
	Multi-call	52.7	2600	640	3240	2950	4590
	EGR (ours)	53.8	1940	640	2580	2315	3895
SWE-bench	Prompt tags	24.7	4090	6580	10670	10020	13240
	FSM-mask	28.8	4290	6590	10880	10230	13480
	Multi-call	30.1	5710	6605	12315	11710	14890
	EGR (ours)	31.4	4340	6595	10935	10285	13570
Paired deltas: EGR minus baseline							
ToolBench vs FSM-mask		+1.5 [+0.4, +2.6]			+20 [-55, +95]		
ToolBench vs Multi-call		+1.1 [+0.1, +2.1]			-660 [-740, -580]		
SWE-bench vs FSM-mask		+2.6 [+1.1, +4.0]			+55 [-110, +230]		
SWE-bench vs Multi-call		+1.3 [+0.1, +2.5]			-1380 [-1510, -1250]		

Replay and live measurements agree: EGR remains near FSM-mask total time and far below Multi-call. On SWE-bench, external execution dominates absolute latency, while EGR retains the higher TaskAcc row without introducing restart cost.

Tables 15 and 16 provide the complete controller audit over 492 task–seed pairs per setting. Candidate-code emission remains the `ACT` span, and the count block exposes execution, verification, and commit denominators.

With a visible verification stage, EGR has the best IAR/R-ITR/VSR/TaskAcc bundle and a 1.8-point paired TaskAcc gain over FSM-mask. After collapse, the IAR, VSR, TaskAcc, and win/loss margins contract while the R-ITR point difference remains -0.2. Together with the unchanged-checkpoint protocol, this pattern supports sensitivity to the explicit verifier boundary without attributing the full cross-setting difference to a single module.

The binary win/loss summaries agree with the corresponding pooled success-count differences in all four comparisons, providing a direct check on paired accounting.

Table 17 reports denominators and paired intervals for five route interventions on the 200-task ToolBench development slice under the three-seed protocol.

Using the `ACT` bank for `VERIFY` raises VSR by 31.2 points over 597 committed traces, with interval [26.8, 35.6], while its IAR shift is 2.3 points. This selective disruption demonstrates functional specialization of the learned banks under the trained routing context.

Table 15: Full HumanEval+ Verify controller audit. "Verifier-visible" is the standard protocol with an explicit test-inspection stage, while "VERIFY-collapsed" feeds the same execution feedback directly into the final submission context. The table restores IAR and includes the Multi-call comparator so the verifier-heavy workload can be read with the same controller stack as ToolBench.

Setting	Method	IAR ↓	R-ITR ↓	VSR ↓	TaskAcc ↑	Latency
Visible	Prompt tags	15.3	21.8	16.7	61.2	1.00×
	FSM-mask	12.6	16.0	12.1	65.6	1.06×
	Multi-call	12.1	17.0	12.5	66.9	1.38×
	EGR (ours)	11.4	15.8	11.3	67.4	1.05×
Collapsed	Prompt tags	13.9	17.1	6.2	63.0	1.00×
	FSM-mask	12.1	14.8	5.8	64.2	1.06×
	Multi-call	11.8	15.0	5.6	64.8	1.36×
	EGR (ours)	11.7	14.6	5.4	65.1	1.05×

Table 16: Counts, paired deltas, and task–seed win/loss summaries for the HumanEval+ Verify controller audit. Each setting aggregates over the same 492 task–seed pairs. In the collapsed setting, $n_{\text{VERIFY}} = 0$ by construction because the separate inspection stage is removed.

Setting	Method	Success	Exec.	VERIFY	COMMIT
Visible	Prompt tags	301	492	492	492
	FSM-mask	323	492	492	492
	Multi-call	329	492	492	492
	EGR	332	492	492	492
Collapsed	Prompt tags	310	492	0	492
	FSM-mask	316	492	0	492
	Multi-call	319	492	0	492
	EGR	320	492	0	492

Comparison	ΔIAR	ΔR-ITR	ΔVSR	ΔTaskAcc	W / L / T
Visible vs. FSM	-1.2 [-1.8, -0.6]	-0.2 [-0.7, 0.3]	-0.8 [-1.4, -0.2]	+1.8 [+0.6, +2.9]	120 / 111 / 261
Visible vs. Multi	-0.7 [-1.2, -0.1]	-1.2 [-1.7, -0.6]	-1.2 [-1.8, -0.6]	+0.5 [-0.4, +1.4]	119 / 116 / 257
Collapsed vs. FSM	-0.4 [-0.9, 0.1]	-0.2 [-0.5, 0.2]	-0.4 [-0.8, 0.1]	+0.9 [0.0, +1.8]	114 / 110 / 268
Collapsed vs. Multi	-0.1 [-0.6, 0.3]	-0.4 [-0.8, 0.0]	-0.2 [-0.5, 0.2]	+0.3 [-0.6, +1.1]	109 / 108 / 275

The visible-verifier regime produces the strongest paired separation, while the collapsed rerun preserves the matched checkpoint and accounting needed to isolate the boundary's empirical role.

Table 17: Route-module substitution and ablation results on ToolBench-200 development. Learned adapter banks are exchanged or removed while all other weights remain fixed; positive phase-error deltas and negative TaskAcc deltas indicate disruption.

Intervention	COMMIT	ΔIAR (95% CI)	ΔR-ITR (95% CI)	ΔVSR (95% CI)	ΔTaskAcc (95% CI)
Use `PLAN` bank for `ACT`	598	+14.7 [12.2, 17.5]	+3.2 [2.1, 4.4]	+1.1 [0.2, 2.0]	−8.3 [-10.1, -6.4]
Use `ACT` bank for `VERIFY`	597	+2.3 [1.2, 3.4]	+5.8 [4.1, 7.6]	+31.2 [26.8, 35.6]	−4.1 [-5.5, -2.7]
Use `COMMIT` bank for `PLAN`	600	+4.1 [2.8, 5.3]	+9.4 [7.6, 11.1]	+2.7 [1.7, 3.7]	−5.6 [-7.1, -4.0]
Ablate all phase adapters (trunk only)	600	+11.8 [9.5, 13.9]	+12.1 [10.0, 14.0]	+9.4 [7.5, 11.4]	−9.2 [-10.8, -7.6]
Ablate trunk layer 28 (keep adapters)	600	+1.4 [0.6, 2.2]	+1.8 [1.0, 2.7]	+0.9 [0.2, 1.6]	−2.1 [-3.2, -1.0]

The intervention signatures are phase selective: ACT-bank substitution primarily disrupts verifier evidence, whereas PLAN-bank substitution at ACT boundaries primarily disrupts action validity.

Table 18: Pooled counts for the ToolBench development comparison: 200 tasks across three runs. Event totals vary with generated trajectory length.

Method	n_{success}	n_{ACT}	n_{trans}	n_{COMMIT}
Prompt tags	290	1486	2027	600
FSM-mask	314	1468	1974	600
MRL	317	1457	1971	600
Multi-call	316	1440	1960	600
EGR (ours)	323	1449	1968	600

All methods cover 600 development task–run pairs; the success counts preserve the development TaskAcc ordering, and similar event totals exclude trajectory collapse as an explanation for EGR's gain.

B Route Certificates and Realized-Event Validity

The declared route is recorded immediately after masked selection, whereas the realized trace is reconstructed from environment-facing events and their pre-event states. For example, the grammar permits a declared `VERIFY→COMMIT` edge, but the corresponding commit is invalid when the environment state contains no completed verifier result for the latest action. The route certificate is legal, while the state-conditioned realized edge is premature and contributes to R-ITR and VSR. A missing return can similarly insert a `MISSING_RETURN` sentinel even when every declared edge is legal. Self-loops and repeated action–verification cycles remain in the denominators; missing, malformed, truncated, and post-commit events enter the realized trace; a trace without an event edge contributes to the malformed count outside the R-ITR denominator.

Table 19: Full route-certificate and realized-event audit. Rates are seed means in percent; counts are pooled across three seeds. Hard-masked methods have exactly zero declared illegal transitions, while state-conditioned event errors distinguish their realized behavior.

Setting	Method	D-ITR	R-ITR	Disagree	Malform.	Tasks	Decl. Event	Disagree n	
ToolBench	Prompt tags	6.8	18.4	14.7	18	600	2016	2027	296
	FSM-mask	0.0	13.6	14.1	9	600	1968	1974	278
	MRL	0.0	13.5	14.1	8	600	1966	1971	278
	Multi-call	0.0	15.1	15.5	6	600	1952	1960	303
	EGR	0.0	13.4	13.8	5	600	1962	1968	271
SWE-bench	Prompt tags	9.7	24.1	25.8	16	360	1164	1178	300
	FSM-mask	0.0	15.2	15.9	8	360	1080	1089	172
	MRL	0.0	14.9	15.8	7	360	1085	1093	171
	Multi-call	0.0	16.4	17.0	6	360	1088	1096	185
	EGR	0.0	14.6	15.1	5	360	1084	1091	164
HumanEval+	Prompt tags	7.2	21.8	22.8	11	492	1598	1612	365
	FSM-mask	0.0	16.0	16.6	7	492	1589	1601	263
	MRL	0.0	15.9	16.6	6	492	1592	1605	264
	Multi-call	0.0	17.0	17.7	5	492	1582	1594	280
	EGR	0.0	15.8	16.4	4	492	1588	1599	260

Table 20: Paired EGR-minus-baseline transition-semantics differences in percentage points. Negative values favor EGR; brackets are task-cluster 95% bootstrap intervals.

Setting	Baseline	ΔR-ITR	ΔDisagree
ToolBench	FSM-mask	-0.2 [-0.6, 0.2]	-0.3 [-0.8, +0.2]
	MRL	-0.1 [-0.5, +0.3]	-0.3 [-0.8, +0.2]
SWE-bench-120	FSM-mask	-0.6 [-1.1, -0.1]	-0.8 [-1.4, -0.2]
	MRL	-0.3 [-0.8, +0.2]	-0.7 [-1.3, -0.1]
HumanEval+ Verify	FSM-mask	-0.2 [-0.7, 0.3]	-0.2 [-0.7, +0.3]
	MRL	-0.1 [-0.6, +0.4]	-0.2 [-0.7, +0.3]

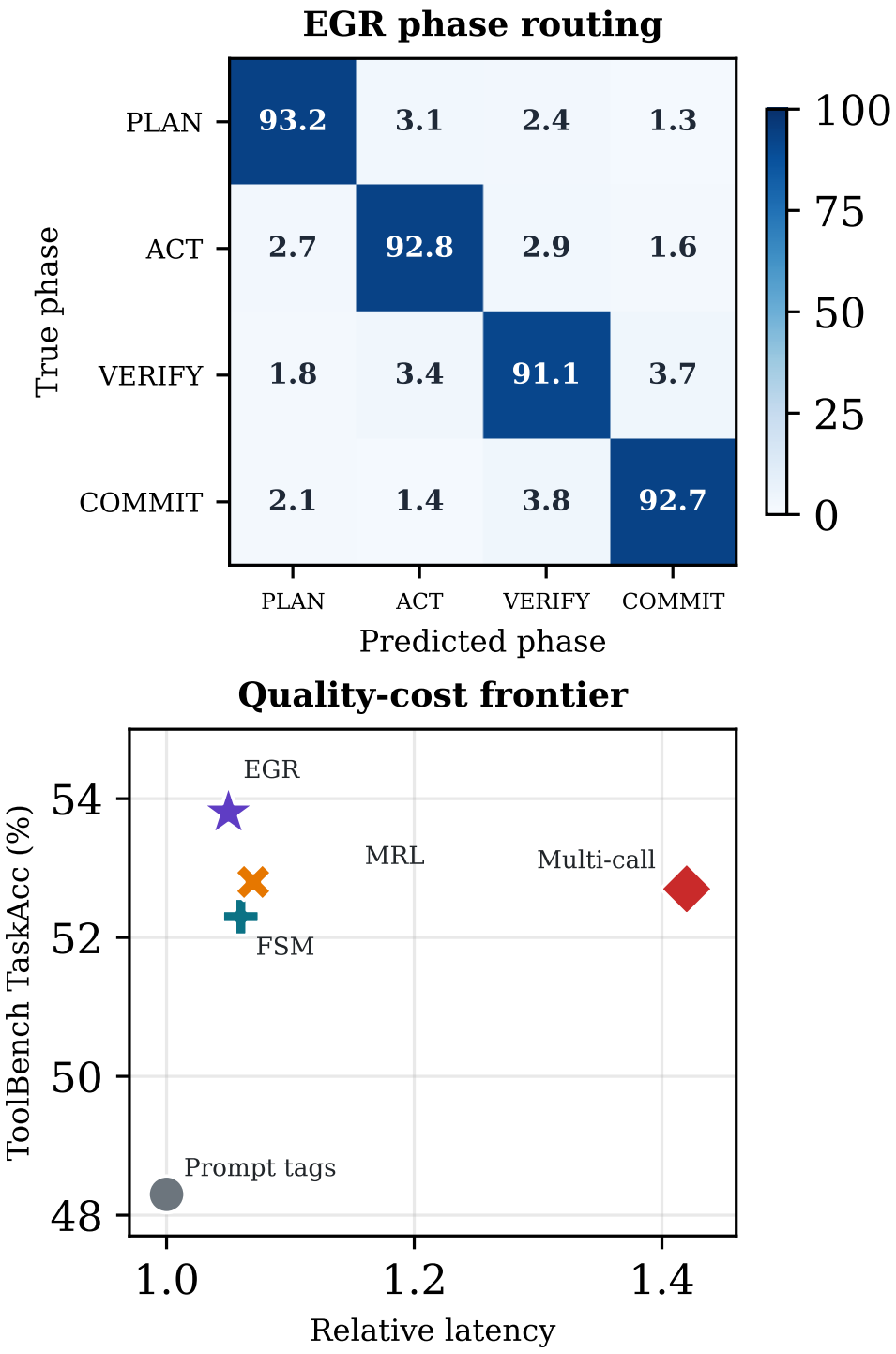

Figure 3: Left: row-normalized EGR phase-routing confusion matrix on ToolBench-200 development. Right: TaskAcc versus normalized model-side latency on the same slice. EGR combines the highest task success with near-controller generation cost.

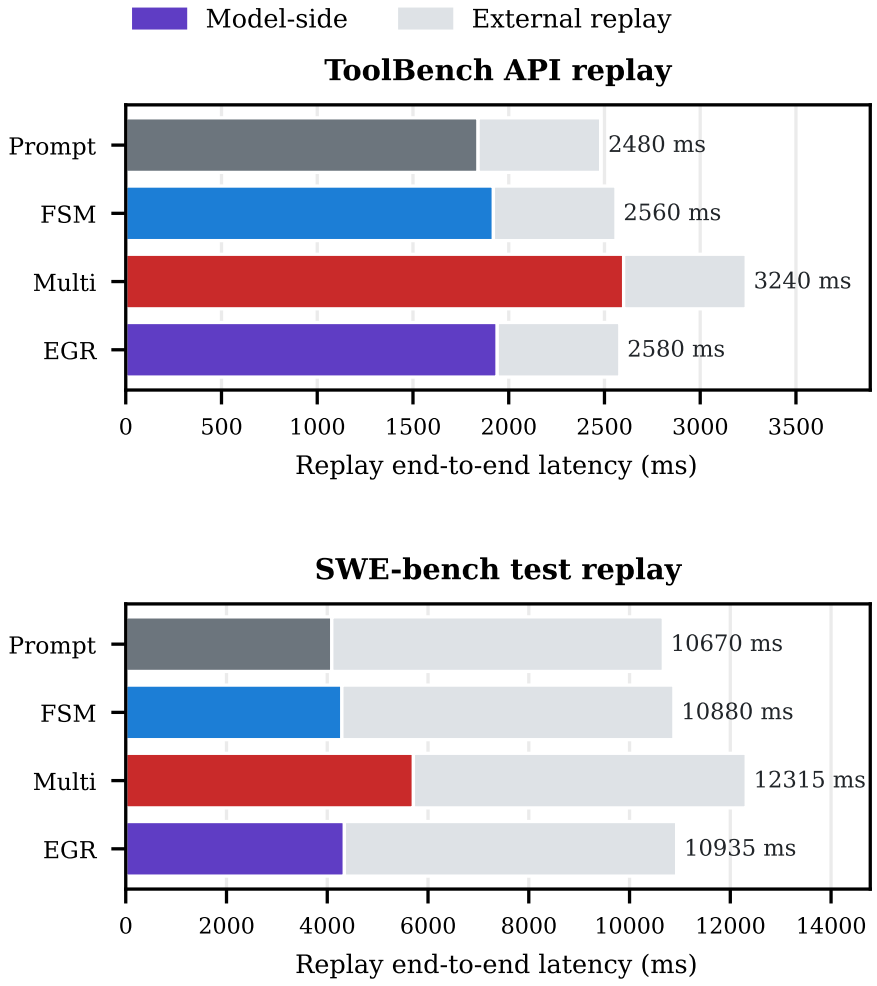

Figure 4: Replay-based end-to-end latency decomposition. Each bar separates measured model-side time from replayed external tool or test time on the fixed evaluation manifests. Left: ToolBench API-replay regime, where model-side overhead is still a material share of total time. Right: public SWE-bench test-replay regime, where external execution dominates and EGR stays close to FSM-mask while remaining much faster than Multi-call.

C BUDGET-MATCHED INTERNAL SPECIALIZATION

Masked Routed LoRA uses all-layer phase routing with the same transition head, hard grammar, training tokens, optimizer budget, and trainable parameter count as EGR. Rank 4 matches the top-four rank-32 EGR budget on 32-layer Llama, while rank 2 matches it on 64-layer Qwen2.5-32B. It retains historical all-layer KV entries and switches the all-layer bank before processing the canonical opening delimiter. Relative to EGR it changes both adapter placement and cache composition: it has neither a route-independent lower trunk for new tokens nor delimiter-token top-stack re-entry.

Table 21: Masked Routed LoRA results and denominators. Top: metrics, latency, and compute budget, with mean±standard deviation over three seeds. Bottom: pooled event counts from the same runs.

Setting	IAR	R-ITR	VSR	TaskAcc	Latency	Params M	tok/s
ToolBench dev.	18.6±0.9	13.5±0.8	8.7±0.6	52.8±1.1	1.07±0.01	41.96	45.8
ToolBench test	18.8±0.8	13.4±0.7	8.6±0.5	52.2±1.0	1.07±0.01	41.96	45.4
HumanEval+	12.2±0.7	15.9±0.9	11.8±0.7	66.3±1.2	1.07±0.01	41.96	47.1
SWE-bench-120	15.9±0.8	14.9±0.9	12.7±0.7	30.2±1.3	1.07±0.01	41.96	39.6

Setting	Tasks	ACT	Event	VERIFY	COMMIT
ToolBench dev.	600	1457	1971	1436	600
ToolBench test	1440	3472	4728	3418	1440
HumanEval+	492	492	1605	492	492
SWE-bench-120	360	812	1093	392	360

D DISJOINT PUBLIC SWE-BENCH EXTENSION

The held-out extension contains 240 issues from 9 repositories fixed before scoring; pooling it with the original 120-issue anchor yields 360 unique issues. Each issue receives one revision after the first test run, identical retrieved context, and the same stopping rule across methods.

Table 22: Held-out and pooled SWE-bench frontier. Top: metrics as mean±standard deviation over three seeds. Bottom: split composition and pooled event counts from the same runs.

Split	Method	IAR	R-ITR	VSR	TaskAcc	Latency
Held-out 240	Prompt tags	20.6±1.0	25.0±1.2	19.4±1.0	23.1±1.5	1.00±0.01
	FSM-mask	17.5±0.9	15.9±1.0	14.2±0.8	27.5±1.4	1.05±0.01
	MRL	16.8±0.9	15.6±0.9	13.5±0.8	28.7±1.3	1.07±0.01
	Multi-call	16.5±0.8	17.0±1.0	13.6±0.8	29.4±1.3	1.40±0.02
	EGR	15.9±0.8	15.3±0.9	12.8±0.7	30.3±1.2	1.06±0.01
Pooled 360	Prompt tags	20.3±0.9	24.7±1.1	19.1±0.9	23.6±1.3	1.00±0.01
	FSM-mask	17.2±0.8	15.7±0.9	14.0±0.7	27.9±1.2	1.05±0.01
	MRL	16.5±0.8	15.4±0.8	13.2±0.7	29.2±1.2	1.07±0.01
	Multi-call	16.3±0.8	16.8±0.9	13.4±0.7	29.6±1.1	1.40±0.02
	EGR	15.6±0.7	15.1±0.8	12.5±0.7	30.7±1.1	1.06±0.01

Split	Method	Issues	Repos	Success	ACT	Event	VERIFY	COMMIT
Held-out 240	Prompt tags	240	9	166	1718	2394	805	720
	FSM-mask	240	9	198	1661	2204	780	720
	MRL	240	9	207	1649	2207	787	720
	Multi-call	240	9	212	1628	2222	792	720
	EGR	240	9	218	1642	2209	798	720
Pooled 360	Prompt tags	360	9	255	2560	3572	1206	1080
	FSM-mask	360	9	302	2477	3293	1170	1080
	MRL	360	9	316	2461	3300	1179	1080
	Multi-call	360	9	320	2432	3318	1186	1080
	EGR	360	9	331	2452	3300	1195	1080

The disjoint extension preserves the quality ordering while comparable action and event counts show that EGR's higher resolved rate is not produced by shortened trajectories.

Table 23: Repository-stratified TaskAcc on the disjoint SWE-bench extension. Each cell displays pooled binary success over three seeds (integer successes divided by three times the issue count); aggregate TaskAcc is the equal-weight average of the underlying unrounded repository rates.

Repository	Issues	Prompt tags	FSM-mask	MRL	Multi-call	EGR
django	34	28.4	33.3	34.3	34.3	34.3
sympy	30	20.0	24.4	26.7	26.7	27.8
sklearn	28	21.4	25.0	27.4	28.6	29.8
matplotlib	26	25.6	28.2	29.5	30.8	32.1
pytest	25	26.7	30.7	32.0	33.3	33.3
pandas	24	18.1	23.6	23.6	25.0	26.4
flask	22	24.2	27.3	28.8	28.8	30.3
requests	20	26.7	31.7	31.7	31.7	33.3
xarray	31	17.2	23.7	24.7	25.8	25.8

EGR is nonnegative against FSM-mask in every repository row, including the largest django, sympy, sklearn, and xarray strata.

Table 24: Paired EGR-minus-baseline intervals on the disjoint and pooled SWE-bench evaluations. Phase and latency deltas are lower-is-better; TaskAcc is higher-is-better.

Split	Baseline	ΔIAR	ΔR-ITR	ΔVSR	ΔTaskAcc
Held-out 240	FSM	-1.6 [-2.3, -0.9]	-0.6 [-1.1, -0.1]	-1.4 [-2.1, -0.7]	+2.8 [+1.3, +4.2]
Held-out 240	MRL	-0.9 [-1.6, -0.2]	-0.3 [-0.8, +0.2]	-0.7 [-1.4, 0.0]	+1.6 [+0.5, +2.7]
Held-out 240	Multi	-0.6 [-1.2, 0.0]	-1.7 [-2.3, -1.1]	-0.8 [-1.4, -0.2]	+0.9 [-0.1, +1.9]
Pooled 360	FSM	-1.6 [-2.2, -1.0]	-0.6 [-1.0, -0.2]	-1.5 [-2.1, -0.9]	+2.8 [+1.5, +4.0]
Pooled 360	MRL	-0.9 [-1.5, -0.3]	-0.3 [-0.7, +0.1]	-0.7 [-1.3, -0.1]	+1.5 [+0.5, +2.5]
Pooled 360	Multi	-0.7 [-1.2, -0.1]	-1.7 [-2.2, -1.2]	-0.9 [-1.4, -0.3]	+1.1 [+0.2, +2.0]

Split	Baseline	Δlatency	Wins / losses / ties
Held-out 240	FSM	+0.01 [-0.01, +0.02]	132 / 112 / 476
Held-out 240	MRL	-0.01 [-0.02, 0.0]	122 / 111 / 487
Held-out 240	Multi	-0.34 [-0.37, -0.31]	116 / 110 / 494
Pooled 360	FSM	+0.01 [-0.01, +0.02]	189 / 160 / 731
Pooled 360	MRL	-0.01 [-0.02, 0.0]	179 / 164 / 737
Pooled 360	Multi	-0.34 [-0.37, -0.31]	173 / 162 / 745

The paired intervals retain EGR's endpoint advantage on the held-out and pooled manifests; Multi-call approaches its quality only with a substantially larger latency ratio.

E WebArena Transfer

Browser events define the same external phase interface: intent formation is `PLAN`, schema-valid browser operations are `ACT`, observations and state checks are `VERIFY`, and task submission is `COMMIT`. WebArena VSR is the fraction of realized submissions lacking a completed state check attached to the latest browser action; its denominator is realized submissions. The held-out split spans 6 sites and 180 tasks, with three seeded evaluations per task; the disjoint 542/90/180 train/development/test partition is fixed before training.

Table 25: WebArena frontier and trace counts. IAR measures invalid browser actions; IAR, R-ITR, and VSR are percentages, while TaskAcc and latency are mean±standard deviation.

Method	IAR	R-ITR	VSR	TaskAcc	Latency
Prompt tags	16.8±1.0	20.5±1.1	15.7±0.9	36.1±1.6	1.00±0.01
FSM-mask	13.9±0.8	12.8±0.9	11.6±0.7	39.8±1.5	1.06±0.01
MRL	13.2±0.8	12.5±0.8	11.0±0.7	40.7±1.4	1.08±0.01
Multi-call	12.9±0.8	14.1±0.9	10.8±0.7	41.5±1.4	1.43±0.02
EGR	12.3±0.7	12.1±0.8	10.2±0.6	42.4±1.3	1.05±0.01

Method	Tasks	Sites	Success	ACT	Event	VERIFY	COMMIT
Prompt tags	540	6	195	6208	12496	776	528
FSM-mask	540	6	215	5964	11974	812	532
MRL	540	6	220	5918	11880	824	534
Multi-call	540	6	224	5842	11726	836	536
EGR	540	6	229	5886	11812	842	537

EGR combines the highest browser-task success with the lowest IAR, R-ITR, and VSR, while its event volume remains comparable to the single-trajectory controls.

Table 26: WebArena site-stratified TaskAcc. Each cell displays pooled binary success over three seeds (integer successes divided by three times the site task count); aggregate TaskAcc is the equal-weight average of the underlying unrounded site rates.

Site	Tasks	Prompt tags	FSM-mask	MRL	Multi-call	EGR
shopping	35	37.1	41.0	41.9	41.9	42.9
reddit	30	32.2	35.6	37.8	37.8	38.9
gitlab	28	41.7	45.2	45.2	46.4	47.6
map	26	32.1	35.9	35.9	38.5	38.5
cms	31	36.6	40.9	40.9	41.9	43.0
shopping_admin	30	36.7	40.0	42.2	42.2	43.3

Table 27: Paired EGR-minus-baseline intervals on WebArena. Live total-time deltas use concurrency one and milliseconds.

Baseline	ΔIAR	ΔR-ITR	ΔVSR	ΔTaskAcc
FSM	-1.6 [-2.3, -0.9]	-0.7 [-1.2, -0.2]	-1.4 [-2.0, -0.7]	+2.6 [+1.0, +4.1]
MRL	-0.9 [-1.5, -0.2]	-0.4 [-0.9, +0.1]	-0.8 [-1.4, -0.1]	+1.7 [+0.5, +2.9]
Multi	-0.6 [-1.2, 0.0]	-2.0 [-2.6, -1.4]	-0.6 [-1.2, 0.0]	+0.9 [-0.2, +2.0]

Baseline	Δmodel latency	Δlive total (ms)
FSM	-0.01 [-0.02, 0.0]	-30 [-210, +148]
MRL	-0.03 [-0.04, -0.02]	-70 [-252, +110]
Multi	-0.38 [-0.42, -0.35]	-800 [-970, -632]

The site-stratified and paired views agree: EGR's gain over grammar-only and all-layer routing is distributed across sites rather than driven by one browser environment.

F Phase-Boundary Robustness

Identical corruptions are applied to every method: boundary jitter offsets the opening marker by one parsed event, dropped verification removes the separate `VERIFY` opening while retaining its returned evidence, and PLAN/VERIFY merge maps both labels to a shared deliberation state. For each run and severity, one task-indexed corruption manifest fixes the affected boundary locations and operation types before method evaluation and is reused across methods; the percentage is the fraction of eligible boundaries changed. The deterministic heuristic instead uses event-token patterns without learned boundary calibration. Tables 28 and 29 report the full error bundle and pooled denominators; Table 30 reports paired TaskAcc uncertainty under task-cluster resampling that retains the matched corruption manifest.

Table 28: Extended ToolBench robustness under matched phase-boundary corruption.

Noise	Method	TaskAcc	IAR	R-ITR	VSR	Disagree	PhaseAcc	Tasks	Events
Clean	FSM-mask	51.4±1.0	19.6	13.7	9.0	14.2	86.9	1440	4744
	MRL	52.2±1.0	18.8	13.4	8.6	14.0	87.4	1440	4728
	Multi-call	52.5±1.0	19.3	14.8	9.1	15.3	86.2	1440	4709
	EGR	53.3±1.0	18.1	13.2	8.1	13.7	88.1	1440	4718
5%	FSM-mask	50.9±1.1	20.0	14.0	9.4	15.1	84.4	1440	4761
	MRL	51.8±1.0	19.2	13.7	9.0	14.8	85.0	1440	4740
	Multi-call	52.0±1.0	19.7	15.1	9.5	16.0	83.9	1440	4722
	EGR	52.6±1.0	18.7	13.7	8.7	14.7	85.7	1440	4731
10%	FSM-mask	49.8±1.2	20.8	14.8	10.1	16.5	80.8	1440	4798
	MRL	50.7±1.1	20.1	14.5	9.7	16.2	81.5	1440	4769
	Multi-call	51.0±1.1	20.4	15.8	10.1	17.1	80.2	1440	4751
	EGR	51.2±1.1	19.8	14.6	9.6	16.1	81.8	1440	4760
20%	FSM-mask	47.7±1.4	22.4	16.3	11.5	19.1	73.6	1440	4876
	MRL	48.5±1.3	21.8	16.0	11.2	18.8	74.3	1440	4842
	Multi-call	49.4±1.2	21.6	17.2	11.3	19.4	73.4	1440	4808
	EGR	49.0±1.3	21.6	16.2	11.0	18.9	74.0	1440	4829
Heuristic	FSM-mask	48.8±1.3	21.7	15.7	10.9	18.2	76.8	1440	4831
	MRL	49.6±1.2	21.0	15.3	10.5	17.8	77.5	1440	4806
	Multi-call	50.1±1.2	21.1	16.6	10.8	18.6	76.2	1440	4782
	EGR	49.9±1.2	21.0	15.5	10.5	17.9	77.2	1440	4794

EGR retains the leading ToolBench TaskAcc row through 10% matched corruption, with a gradual rather than abrupt degradation as boundary information is removed.

Table 29: Held-out SWE-bench robustness under matched phase-boundary corruption.

Noise	Method	TaskAcc	IAR	R-ITR	VSR	Disagree	PhaseAcc	Tasks	Events
Clean	FSM-mask	27.5±1.4	17.5	15.9	14.2	16.5	84.2	720	2204
	MRL	28.7±1.3	16.8	15.6	13.5	16.2	84.7	720	2207
	Multi-call	29.4±1.3	16.5	17.0	13.6	17.6	83.5	720	2222
	EGR	30.3±1.2	15.9	15.3	12.8	15.9	85.4	720	2209
5%	FSM-mask	26.9±1.5	18.0	16.3	14.7	17.4	81.9	720	2219
	MRL	28.0±1.4	17.3	16.0	14.0	17.1	82.4	720	2221
	Multi-call	28.8±1.4	17.0	17.4	14.1	18.2	81.1	720	2234
	EGR	29.5±1.3	16.6	15.8	13.6	16.9	82.9	720	2224
10%	FSM-mask	25.8±1.6	18.8	17.1	15.5	18.9	78.0	720	2246
	MRL	27.0±1.5	18.1	16.8	14.9	18.5	78.7	720	2241
	Multi-call	27.9±1.5	17.8	18.1	14.9	19.4	77.2	720	2260
	EGR	28.3±1.4	17.8	16.8	14.8	18.5	78.9	720	2248
20%	FSM-mask	23.9±1.8	20.5	18.8	17.2	21.6	70.3	720	2302
	MRL	25.2±1.7	19.8	18.4	16.6	21.1	71.1	720	2291
	Multi-call	26.5±1.6	19.4	19.6	16.4	22.0	69.8	720	2314
	EGR	26.0±1.7	19.7	18.6	16.8	21.4	70.7	720	2296
Heuristic	FSM-mask	24.8±1.7	19.8	18.1	16.5	20.4	73.4	720	2278
	MRL	25.9±1.6	19.1	17.8	15.9	20.0	74.0	720	2269
	Multi-call	26.9±1.6	18.8	19.0	15.8	20.8	72.7	720	2290
	EGR	26.7±1.6	19.1	17.9	16.1	20.2	73.8	720	2274

The SWE-bench corruption curve reproduces the same dependence on boundary quality while retaining positive paired EGR–FSM TaskAcc differences at every reported severity.

Table 30: Paired TaskAcc deltas for boundary robustness. Each cell is EGR minus baseline with a 95% interval.

Setting	Noise	vs. FSM-mask	vs. MRL
ToolBench	Clean	+1.9 [+1.0, +2.7]	+1.1 [+0.2, +2.0]
	5%	+1.7 [+0.8, +2.6]	+0.8 [0.0, +1.6]
	10%	+1.4 [+0.4, +2.4]	+0.5 [-0.4, +1.4]
	20%	+1.3 [+0.1, +2.5]	+0.5 [-0.6, +1.6]
	Heuristic	+1.1 [0.0, +2.2]	+0.3 [-0.7, +1.3]
SWE-bench held-out	Clean	+2.8 [+1.3, +4.2]	+1.6 [+0.5, +2.7]
	5%	+2.6 [+1.1, +4.1]	+1.5 [+0.4, +2.6]
	10%	+2.5 [+0.9, +4.0]	+1.3 [+0.1, +2.5]
	20%	+2.1 [+0.3, +3.9]	+0.8 [-0.6, +2.2]
	Heuristic	+1.9 [+0.2, +3.6]	+0.8 [-0.5, +2.1]

G Model-Family and Scale Transfer

The trace-complete Qwen2.5-7B results in Table 10 establish cross-family transfer. Qwen2.5-32B uses the same phase derivation and three-seed protocol; EGR and Masked Routed LoRA each use a matched 67.13M trainable-parameter budget.

At 32B scale, EGR retains the lowest IAR, R-ITR, and VSR and the highest TaskAcc in both settings. Its paired TaskAcc gain over FSM-mask remains positive on ToolBench and SWE-bench-120, while differences from the stronger learned and restart controls are smaller; the latency row preserves the single-trajectory advantage over Multi-call.

Table 31: Qwen2.5-32B frontier. Rates, TaskAcc, and latency are mean±standard deviation; counts are pooled.

Setting	Method	IAR	D-ITR	R-ITR	VSR	TaskAcc
ToolBench	Prompt tags	17.6±.8	5.2±.4	14.3±.7	9.8±.5	59.7±1.0
	FSM-mask	14.9±.7	0±0	10.9±.6	7.5±.5	63.0±.9
	MRL	14.2±.7	0±0	10.7±.6	7.1±.4	64.0±.9
	Multi-call	14.0±.7	0±0	11.8±.7	7.0±.5	64.3±.9
	EGR	13.7±.6	0±0	10.5±.6	6.7±.4	64.8±.8
SWE-bench-120	Prompt tags	16.1±.9	6.4±.5	20.3±1.0	15.6±.8	31.8±1.3
	FSM-mask	13.8±.8	0±0	12.9±.8	11.0±.7	35.4±1.2
	MRL	13.1±.8	0±0	12.6±.8	10.5±.7	36.3±1.2
	Multi-call	12.8±.8	0±0	14.0±.9	10.4±.7	36.8±1.1
	EGR	12.5±.7	0±0	12.3±.7	9.9±.6	37.4±1.1

Setting	Method	Latency	Params M	Peak GB
ToolBench	Prompt tags	1.00±.01	0.00	54.2
	FSM-mask	1.05±.01	0.00	54.4
	MRL	1.07±.01	67.13	57.8
	Multi-call	1.36±.02	0.00	55.1
	EGR	1.05±.01	67.13	56.3
SWE-bench-120	Prompt tags	1.00±.01	0.00	56.7
	FSM-mask	1.05±.01	0.00	56.9
	MRL	1.07±.01	67.13	60.2
	Multi-call	1.34±.02	0.00	57.6
	EGR	1.06±.01	67.13	58.9

Setting	Method	Tasks	ACT	Event	VERIFY	COMMIT
ToolBench	Prompt tags	600	1410	1934	1392	600
	FSM-mask	600	1391	1888	1378	600
	MRL	600	1384	1886	1385	600
	Multi-call	600	1372	1872	1389	600
	EGR	600	1379	1879	1391	600
SWE-bench-120	Prompt tags	360	818	1152	399	360
	FSM-mask	360	795	1068	389	360
	MRL	360	790	1071	394	360
	Multi-call	360	782	1076	396	360
	EGR	360	787	1069	398	360

At 32B scale, EGR again gives the strongest joint action, transition, evidence, and task-success row while retaining single-trajectory latency.

Table 32: Task-cluster paired EGR-minus-baseline TaskAcc intervals for Qwen2.5-32B-Instruct.

Setting	Baseline	ΔTaskAcc [95% CI]
ToolBench	fsmmask	+1.8 [+0.8, +2.8]
ToolBench	mrl	+0.8 [-0.1, +1.7]
ToolBench	multicall	+0.5 [-0.3, +1.3]
SWE-bench-120	fsmmask	+2.0 [+0.8, +3.2]
SWE-bench-120	mrl	+1.1 [0.0, +2.2]
SWE-bench-120	multicall	+0.6 [-0.4, +1.6]

H Live Service-Time Evaluation

Each task is timed five times after 20 warm-up tasks on the same A100-80GB host, external service, browser image, and test container. Concurrency-one runs contain 3000 ToolBench, 1800 SWE-bench, and 2700 WebArena timing observations per method; concurrency eight uses the same generated traces and timing count. Calls, failures, and model passes report the deterministic trace count for one repetition, while latency and throughput pool all five live repetitions. Total time equals model plus external time in every row.

EGR-minus-FSM-mask concurrency-one total-time estimates are +10 [-75, +94] ms on ToolBench, +60 [-145, +268] ms on SWE-bench, and -30 [-210, +148] ms on WebArena; each unadjusted 95% interval includes zero. Against Multi-call, the corresponding intervals are strictly negative: -680 [-775, -586], -1360 [-1510, -1215], and -800 [-970, -632] ms.

Table 33: Live service time at concurrency one. Time is milliseconds; throughput is tasks/minute. Calls, failures, and model passes are per-repetition trace counts.

Setting	Method	TaskAcc	Model	External	Total	p95
ToolBench	Prompt tags	48.3	1870	760	2630	4560
	FSM-mask	52.3	1960	735	2695	4680
	MRL	52.8	2005	728	2733	4760
	Multi-call	52.7	2665	720	3385	5450
	EGR	53.8	1980	725	2705	4710
SWE-bench	Prompt tags	24.7	4150	7820	11970	17190
	FSM-mask	28.8	4365	7750	12115	17450
	MRL	30.2	4440	7780	12220	17630
	Multi-call	30.1	5805	7730	13535	19220
	EGR	31.4	4410	7765	12175	17560
WebArena	Prompt tags	36.1	2850	5570	8420	12950
	FSM-mask	39.8	3010	5560	8570	13180
	MRL	40.7	3075	5535	8610	13290
	Multi-call	41.5	4070	5270	9340	14240
	EGR	42.4	2990	5550	8540	13080

Setting	Method	p50	p90	Throughput	Tasks	Calls	Failures	Passes
ToolBench	Prompt tags	2335	3980	22.81	600	1486	24	600
	FSM-mask	2400	4075	22.26	600	1468	18	600
	MRL	2435	4140	21.95	600	1457	17	600
	Multi-call	3070	4785	17.73	600	1440	16	1896
	EGR	2410	4095	22.18	600	1449	15	600
SWE-bench	Prompt tags	11020	15380	5.01	360	401	12	360
	FSM-mask	11165	15570	4.95	360	390	9	360
	MRL	11240	15720	4.91	360	392	10	360
	Multi-call	12560	17140	4.43	360	394	8	1094
	EGR	11210	15650	4.93	360	397	8	360
WebArena	Prompt tags	7650	11280	7.13	540	6208	45	540
	FSM-mask	7805	11450	7.00	540	5964	36	540
	MRL	7860	11560	6.97	540	5918	34	540
	Multi-call	8520	12390	6.42	540	5842	31	1847
	EGR	7770	11390	7.03	540	5886	29	540

At concurrency one, EGR remains near FSM-mask in total service time and preserves the restart-cost advantage over Multi-call in all three environments.

Table 34: Live service time at concurrency eight. Time is milliseconds; throughput is tasks/minute.

Setting	Method	TaskAcc	Model	External	Total	p95
ToolBench	Prompt tags	48.3	1120	805	1925	3490
	FSM-mask	52.3	1180	780	1960	3560
	MRL	52.8	1215	775	1990	3630
	Multi-call	52.7	1590	770	2360	4150
	EGR	53.8	1195	772	1967	3585
SWE-bench	Prompt tags	24.7	2480	8350	10830	15800
	FSM-mask	28.8	2610	8260	10870	15880
	MRL	30.2	2665	8300	10965	16030
	Multi-call	30.1	3475	8240	11715	16960
	EGR	31.4	2635	8280	10915	15960
WebArena	Prompt tags	36.1	1710	6030	7740	12180
	FSM-mask	39.8	1805	5965	7770	12230
	MRL	40.7	1845	5950	7795	12290
	Multi-call	41.5	2440	5690	8130	12710
	EGR	42.4	1795	5980	7775	12250

Table 35: Live service trace summary at concurrency eight. Tasks, calls, failures, and model passes are per-repetition trace counts.

Setting	Method	p50	p90	Throughput
ToolBench	Prompt tags	1740	2990	249.35
	FSM-mask	1775	3050	244.90
	MRL	1810	3110	241.21
	Multi-call	2150	3590	203.39
	EGR	1785	3075	244.03
SWE-bench	Prompt tags	9920	14140	44.32
	FSM-mask	9960	14210	44.16
	MRL	10035	14340	43.78
	Multi-call	10780	15180	40.97
	EGR	10005	14280	43.98
WebArena	Prompt tags	7010	10570	62.02
	FSM-mask	7045	10610	61.78
	MRL	7070	10660	61.58
	Multi-call	7410	11030	59.04
	EGR	7050	10630	61.74

Setting	Method	Tasks	Calls	Failures	Passes
ToolBench	Prompt tags	600	1486	24	600
	FSM-mask	600	1468	18	600
	MRL	600	1457	17	600
	Multi-call	600	1440	16	1896
	EGR	600	1449	15	600
SWE-bench	Prompt tags	360	401	12	360
	FSM-mask	360	390	9	360
	MRL	360	392	10	360
	Multi-call	360	394	8	1094
	EGR	360	397	8	360
WebArena	Prompt tags	540	6208	45	540
	FSM-mask	540	5964	36	540
	MRL	540	5918	34	540
	Multi-call	540	5842	31	1847
	EGR	540	5886	29	540

Concurrency eight preserves the same systems conclusion: external work compresses absolute differences, but EGR avoids the extra model passes required by stage restarts.

Table 36: Paired EGR-minus-baseline live total-time deltas in milliseconds.

Setting	Concurrency	Baseline	Δtotal time [95% CI]
ToolBench	1	fsmmask	+10 [-75, +94]
ToolBench	1	mrl	-28 [-112, +55]
ToolBench	1	multicall	-680 [-775, -586]
ToolBench	8	fsmmask	+7 [-48, +63]
ToolBench	8	mrl	-23 [-79, +34]
ToolBench	8	multicall	-393 [-462, -324]
SWE-bench-120	1	fsmmask	+60 [-145, +268]
SWE-bench-120	1	mrl	-45 [-252, +160]
SWE-bench-120	1	multicall	-1360 [-1510, -1215]
SWE-bench-120	8	fsmmask	+45 [-125, +216]
SWE-bench-120	8	mrl	-50 [-220, +121]
SWE-bench-120	8	multicall	-800 [-945, -657]
WebArena-180	1	fsmmask	-30 [-210, +148]
WebArena-180	1	mrl	-70 [-252, +110]
WebArena-180	1	multicall	-800 [-970, -632]
WebArena-180	8	fsmmask	+5 [-138, +149]
WebArena-180	8	mrl	-20 [-165, +124]
WebArena-180	8	multicall	-355 [-480, -231]

I SINGLE-STEP LOW-BOUNDARY REGIME

The BFCL v3 non-live AST subset (Patil et al., 2025) contains 400 simple, 200 multiple-tool selection, 200 parallel, and 200 parallel-multiple single-turn function-calling examples (1,000 tasks; 3,000 task–seed pairs). TaskAcc is the official AST exact-match score. Every prompt maps its prefix to `PLAN`, its generated function-call span to `ACT`, and termination to `COMMIT`; no return or verifier stage is present, so phase-error rates are omitted. FSM-mask retains the legal route, while Masked Routed LoRA retains all-layer phase routing. EGR attains the highest TaskAcc at the same normalized latency as the budget-matched routed control; its margin is 0.5 points over Masked Routed LoRA and 1.7 points over FSM-mask, documenting a smaller incremental effect after internal specialization is matched in this single-step regime.

Table 37: BFCL v3 non-live AST TaskAcc and normalized model-side latency over three seeds.

Method	TaskAcc	Latency
Prompt tags	72.1±0.8	1.00×
FSM-mask	72.5±0.8	1.01×
Masked Routed LoRA	73.7±0.7	1.03×
Multi-call	72.8±0.8	1.15×
EGR	74.2±0.7	1.03×

J WORKFLOW-CONTROL ASSURANCE MATRIX

Table 38: Realized-event control matrix, part I: protected object, failure capability, trust assumption, property, and enforcement boundary. Declared-route legality is a hard property; action validity, verifier-evidence availability, and task success are empirical properties.

ID	Protected object	Failure capability	Trust assumptions	Property	Enforcement boundary
A	Declared route	Illegal phase request	Grammar and phase-opening boundary detector	Valid next-route support	Hard mask before route selection
B	Tool action state	Malformed or premature tool action	Executable API schema and realized-event parser	Lower IAR and R-ITR	Top-stack re-entry before span body
C	Verifier-gated commit	Commit without qualifying test evidence	Observed test execution and final output	Lower VSR; higher TaskAcc	Verifier boundary and selected route
D	Repository repair	Emit patch before executable validation	Fixed issues and deterministic tests	Resolved-rate frontier	EGR re-entry stack
E	Browser workflow	Invalid action or premature submission	Observable browser action and state checks	Action validity and task success	Browser event boundary
F	Service-time frontier	Restart-induced recomputation	Same hardware and external containers	Quality at near-controller total time	Explicit KV ownership

Together, the two panels separate the hard route certificate from the empirical properties of realized actions, evidence-bearing commits, task success, and service cost, and link each claim to its strongest matched or held-out test.

Table 39: Realized-event control matrix, part II: strongest comparator, held-out or adaptive setting, utility, observed failure, and paper-internal evidence.

ID	Protected object	Comparator	Held-out/adaptive setting	Utility and observed failure	Evidence
A	Declared route	FSM-mask	All hard-masked settings	Zero D-ITR; realized event errors remain	Table 19
B	Tool action state	FSM-mask; Masked Routed LoRA	Extended ToolBench and label noise	Near-controller latency; errors rise under boundary corruption	Tables 3, 28
C	Verifier-gated commit	FSM-mask; Multi-call	VERIFY-collapsed HumanEval	Explicit verification yields the largest IAR, VSR, and TaskAcc margins	Tables 15, 11
D	Repository repair	FSM-mask; Masked Routed LoRA	Disjoint 240-issue, 9-repository split	Repository-varying gains with nonzero failures	Tables 22, 23
E	Browser workflow	FSM-mask; Masked Routed LoRA	180 tasks across 6 sites	Site-varying success with nonzero failures	Tables 25, 26
F	Service-time frontier	Multi-call	Concurrency one and eight	External work compresses model-time gaps; restart cost remains measurable	Tables 33, 34

Read row-wise, the matrix ties each claim to the comparator that removes its nearest alternative explanation: route legality to FSM masking, within-phase realization to MRL, and cache-efficient service to Multi-call. The resulting evidence chain supports the integrated event-boundary placement claim across executable tool, code, repository, and browser workflows.