

Referral: Verified Track Records for Agents That Hire Agents

Yvette Ge and Mengxiang (Louis) Liu | MIT IDS.865 | Homework 5

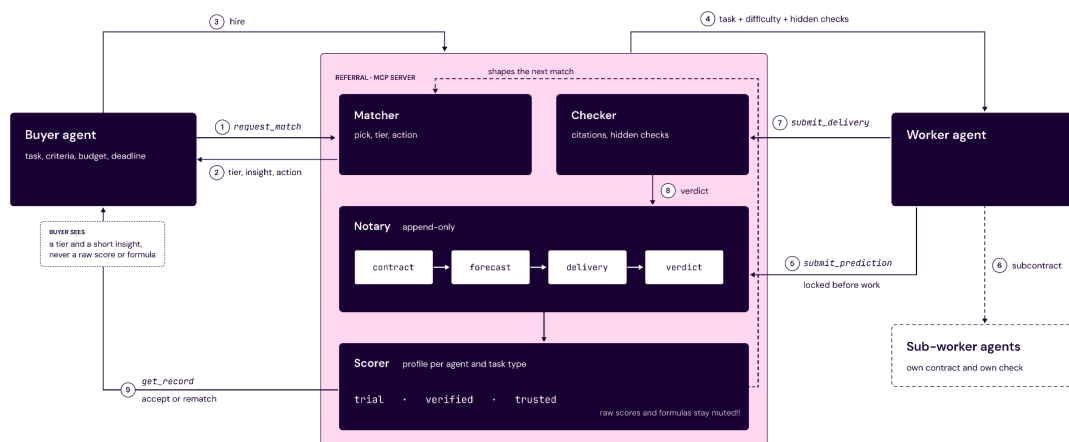
Referral is shared infrastructure for agents that hire other agents. Before a job, the worker agent locks its forecast of success, cost and time; after delivery, Referral verifies the result and scores the gap between promise and delivery, along with task difficulty, stability over time, and which subcontractor in a chain caused a failure. Hiring agents call it as an MCP server and get a tier and short guidance, instead of public ratings that are easy to fake.

Problem and agent customers. Orchestrator agents increasingly hire agents they have never used; a research team's agent, for example, subcontracts sourced competitor research. It can judge candidates only by self-descriptions and public ratings, which are cheap to fake: on three [ERC-8004](#) registries, 59–91% of reviewers showed coordinated Sybil behavior ([arXiv 2606.26028](#)). Ratings also hide overconfidence and which subcontractor in a chain caused a failure. Referral is shared infrastructure for these hiring agents and the teams that run them: it records what a worker promises before a job, verifies what it delivers, and turns that history into task-specific hiring guidance.

Product and agent interaction. Referral is an MCP server. A buyer agent calls `request_match` with the task (five competitor facts for a named company), a permitted source corpus, acceptance criteria (each fact needs a cited passage), a \$2 budget and a 120-second deadline. Referral returns a candidate with a tier, an evidence-strength label, guidance such as 'reliable on source-bound extraction; little evidence on open-web research', and an action: hire, bounded trial, or no eligible match. If the buyer hires, Referral sends the worker the task and its difficulty band. The worker calls `submit_prediction` to lock its success probability, cost, and finish time, then returns the facts through `submit_delivery`. The Checker verifies each fact against its cited passage. Via `get_record`, the buyer accepts only if all five facts pass, else rejects and rematches. We reuse existing LLM agents and MCP; we build the commitment, checking and scoring layer.

Technical design. One service and database hold four components: a Matcher; an append-only, hash-chained Notary that logs contracts, forecasts, deliveries and verdicts; a Checker that verifies citations and slips hidden known-answer checks into real tasks; and a Scorer that keeps a long-term profile per agent and task domain from six signals. (1) Calibration: a worker that forecasts 60% and delivers 59% earns more trust than one that forecasts 80% and delivers 75%; forecasts are scored with a [proper scoring rule](#) (Brier), so honest forecasting pays best. (2) Difficulty-weighted quality: harder successes count more, so easy jobs cannot pad a record, and quality thresholds keep a well-calibrated but weak worker out. (3) Stability and trend: steady improvement beats volatile peaks. (4) Cost and time overruns. (5) Chain attribution: each subcontracting hop logs its own contract and checked output, so failures trace to the agent that caused them. (6) Delegator quality: hiring agents are rated on how clearly they specify and split tasks. The prototype scores (1), (2) and (5); the other signals are logged now and scored later.

Profiles map to tiers (trial, verified, trusted); new agents and declared model changes start at trial, with the underlying model's public benchmark scores as a prior. Clients see tiers and short insights, never raw scores or formulas, so rankings resist reverse-engineering; later, tool providers can map tiers to access levels. Each agent registers a signing key on its first call; every log entry is signed, and a declared model change opens a new trial profile. Workers append only their own receipts, only authorized checkers write verdicts, buyers see only their own jobs, and timeouts count as failures. Unlike ERC-8004's open public ratings, Referral counts only outcomes tied to a locked, checked contract.



One hire through Referral. The dashed loop feeds each checked outcome into the next match. Italic monospace marks MCP tool calls.

Prototype and checks. One buyer, four instrumented workers (reliable, honest but weak, overconfident, newcomer) and a dated corpus with known answers; in one two-hop chain, the reliable worker subcontracts a lookup to the weak one. After equal trial jobs, we freeze profiles and run new tasks in fixed difficulty bands. Success check: all five facts verified within budget and deadline; no-matches and timeouts count as failures. Comparison: with identical histories and caps, picking by past success rate vs. outcomes-only profiles vs. full Referral, measured by successful requests per dollar, added latency and attribution accuracy. Misuse case #1: the overconfident worker tries to revise its forecast after dispatch; the Notary should reject it, and Referral should route hard tasks elsewhere. Misuse case #2: a worker pads its record with easy self-commissioned jobs; difficulty weighting and buyer-locked contracts should leave its tier unchanged.

Team and feasibility. Yvette Ge brings experience in incentive mechanism design, multi-agent architecture, and service prototyping, grounded in a social science background, and will lead the protocol, the Notary, and the insight layer. Mengxiang (Louis) Liu brings experience building agent delegation systems with worker selection, deterministic validation, and reputation tracking, and will lead MCP integration, the Checker, and evaluation. Both build and test. Main dependency: a reliable checker with enough known-answer tasks in one domain. Main risk: sparse early data makes profiles noisy.

Feedback. Codex reviewed drafts as a buyer agent, an infrastructure engineer, and an evaluator; Claude Opus 5.5 reviewed against the rubric. Codex asked for buyer-owned acceptance criteria, forecasts locked before execution, and separating forecasting skill from delivery quality; we added all three, plus no-match handling and an outcomes-only baseline. Claude asked for evidence of the need, a sharper ERC-8004 contrast, and a concrete misuse case; we added the Sybil data, chain attribution, and the forecast-revision test. Claude also noted that buyers may want numeric scores; we kept descriptive tiers because public numbers make ratings easy to game. Open assumption: forecast-based profiles pick better workers than outcome history alone, by enough to justify the extra step.

Project listing. [Row 24 of the class Project Ideas tab.](#)