UniBo

Intelligent System Engineering

2026

Domain

Abstract

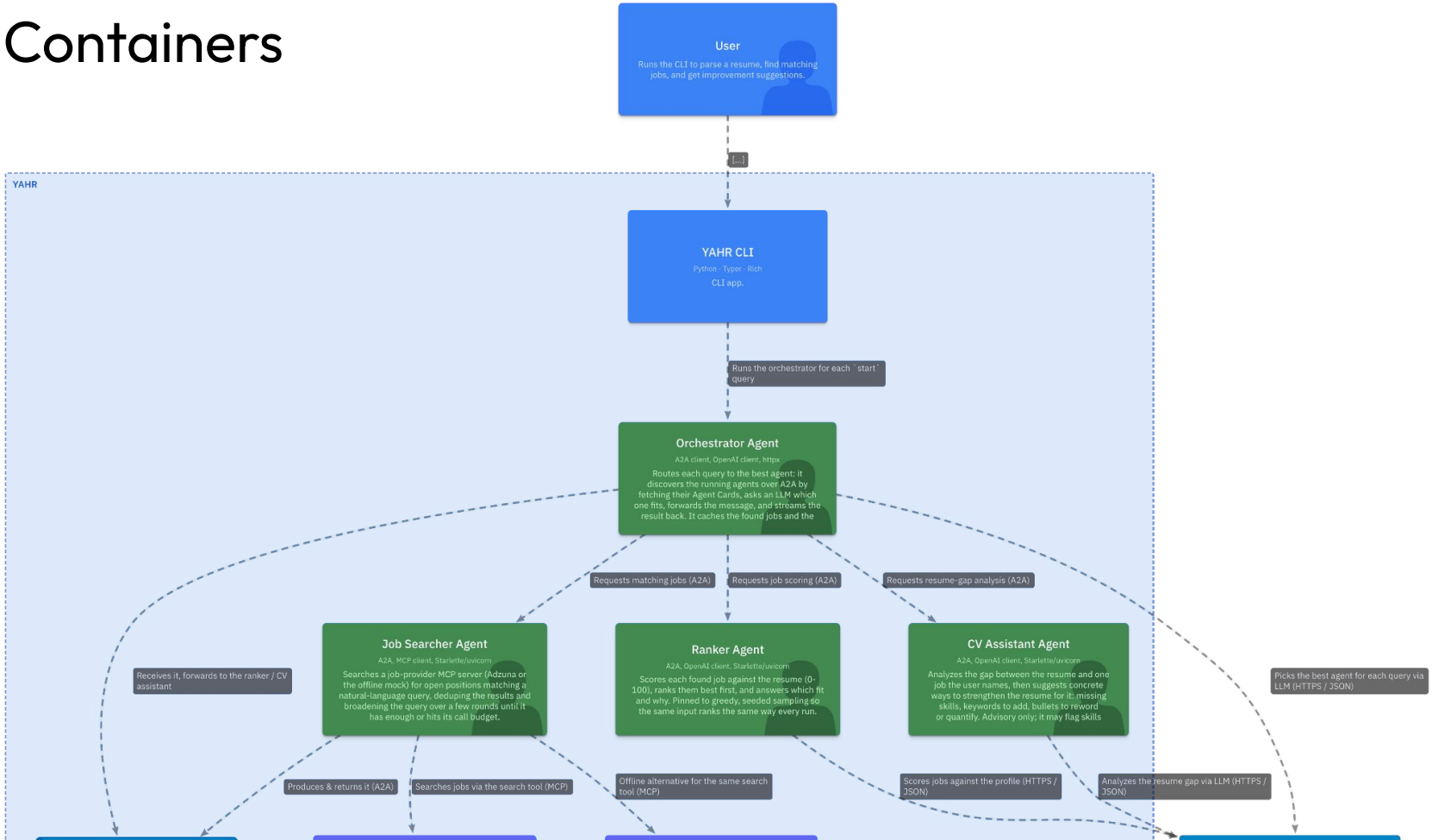
- YAHR (Yet Another HR) is a command-line tool that automates the job search, all in the terminal.
- Give it your resume as a PDF. It builds a structured profile, finds matching jobs, scores each one, and suggests resume edits for a role you pick.
- Four agents talk over A2A (Agent-to-Agent): an Orchestrator routes each request; the Job Searcher finds openings, the Ranker scores them, the CV Assistant reports what to fix.

Design

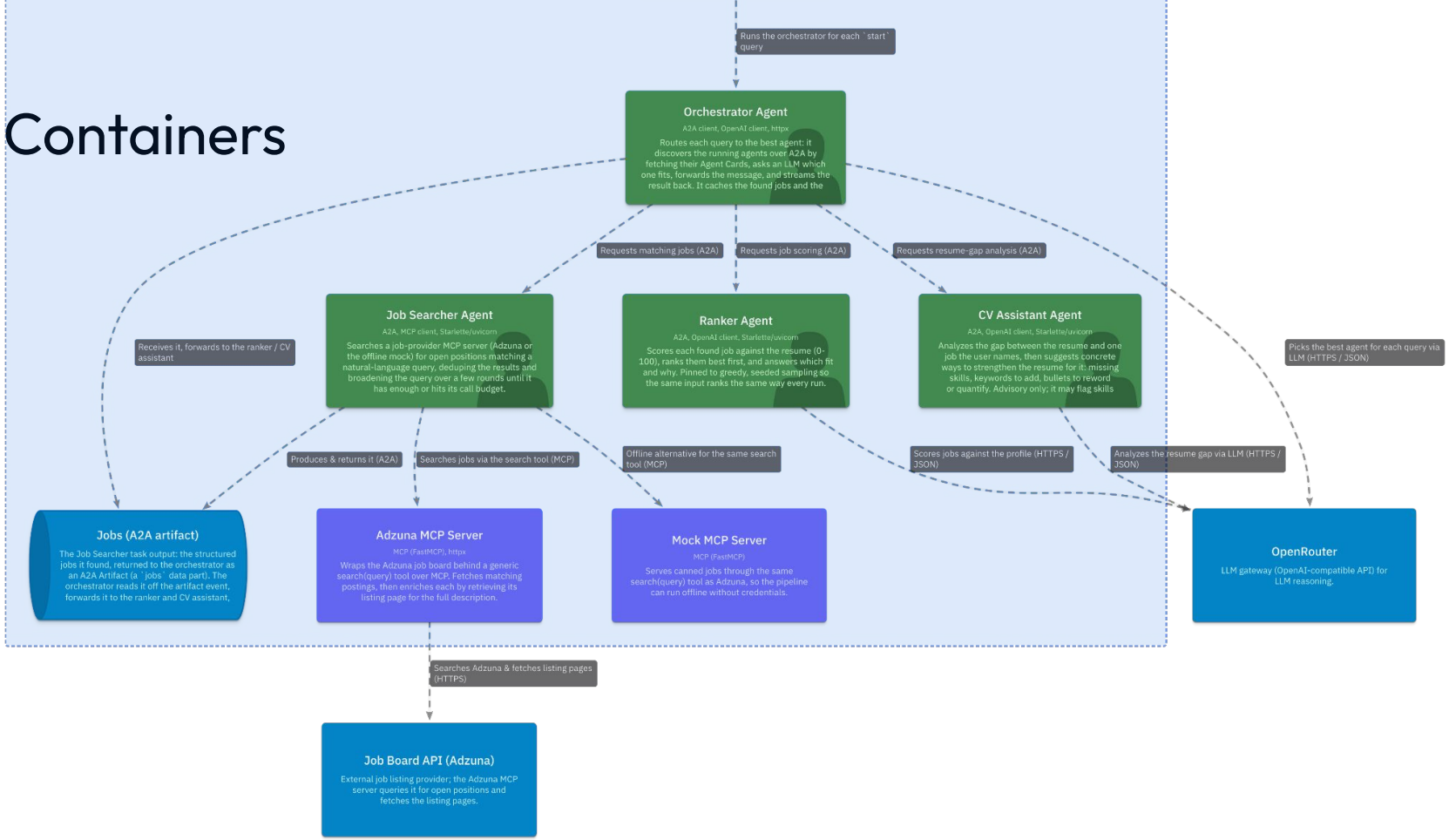
System Context

- The user run the CLI that encapsulate the full system.
- OpenRouter, an OpenAI-compatible LLM gateway, handles every reasoning step over the resume and jobs, over HTTPS.
- Adzuna's API supplies the actual postings, also over HTTPS.

Containers


User
Runs the CLI to parse a resume, find matching jobs, and get improvement suggestions.
[...]
YAHR
YAHR CLI
Python · Typer · Rich
CLI app.
Runs the orchestrator for each `start` query
Orchestrator Agent
A2A client, OpenAI client, httpx
Routes each query to the best agent: it discovers the running agents over A2A by fetching their Agent Cards, asks an LLM which one fits, forwards the message, and streams the result back. It caches the found jobs and the
Requests matching jobs (A2A)
Requests job scoring (A2A)
Requests resume-gap analysis (A2A)
Job Searcher Agent
A2A, MCP client, Starlette/uvicorn
Searches a job-provider MCP server (Adzuna or the offline mock) for open positions matching a natural-language query, deduping the results and broadening the query over a few rounds until it has enough or hits its call budget.
Ranker Agent
A2A, OpenAI client, Starlette/uvicorn
Scores each found job against the resume (0-100), ranks them best first, and answers which fit and why. Pinned to greedy, seeded sampling so the same input ranks the same way every run.
CV Assistant Agent
A2A, OpenAI client, Starlette/uvicorn
Analyzes the gap between the resume and one job the user names, then suggests concrete ways to strengthen the resume for it: missing skills, keywords to add, bullets to reword or quantify. Advisory only; it may flag skills
Receives it, forwards to the ranker / CV assistant
Picks the best agent for each query via LLM (HTTPS / JSON)
Produces & returns it (A2A)
Searches jobs via the search tool (MCP)
Offline alternative for the same search tool (MCP)
Scores jobs against the profile (HTTPS / JSON)
Analyzes the resume gap via LLM (HTTPS / JSON)

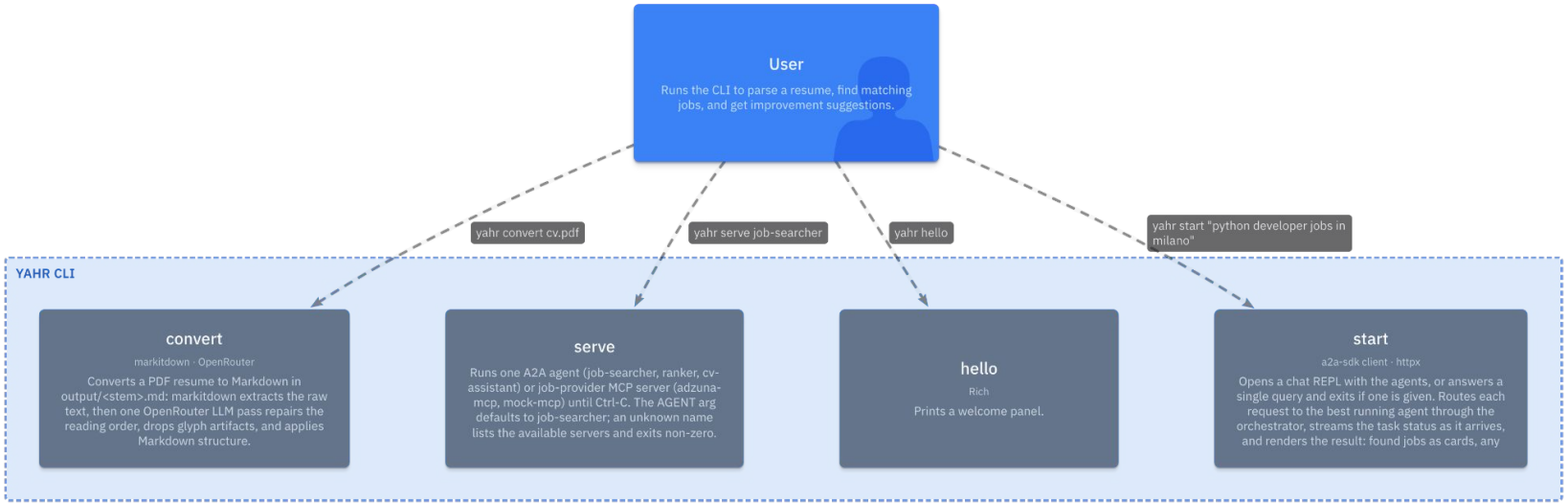
Containers


Runs the orchestrator for each `start` query
Orchestrator Agent
A2A client, OpenAI client, httpx
Routes each query to the best agent: it discovers the running agents over A2A by fetching their Agent Cards, asks an LLM which one fits, forwards the message, and streams the result back. It caches the found jobs and the
Requests matching jobs (A2A)
Requests job scoring (A2A)
Requests resume-gap analysis (A2A)
Receives it, forwards to the ranker / CV assistant
Job Searcher Agent
A2A, MCP client, Starlette/uvicorn
Searches a job-provider MCP server (Adzuna or the offline mock) for open positions matching a natural-language query, deduping the results and broadening the query over a few rounds until it has enough or hits its call budget.
Ranker Agent
A2A, OpenAI client, Starlette/uvicorn
Scores each found job against the resume (0-100), ranks them best first, and answers which fit and why. Pinned to greedy, seeded sampling so the same input ranks the same way every run.
CV Assistant Agent
A2A, OpenAI client, Starlette/uvicorn
Analyzes the gap between the resume and one job the user names, then suggests concrete ways to strengthen the resume for its missing skills, keywords to add, bullets to reword or quantify. Advisory only; it may flag skills
Picks the best agent for each query via LLM (HTTPS / JSON)
Produces & returns it (A2A)
Searches jobs via the search tool (MCP)
Offline alternative for the same search tool (MCP)
Scores jobs against the profile (HTTPS / JSON)
Analyzes the resume gap via LLM (HTTPS / JSON)
Jobs (A2A artifact)
The Job Searcher task output: the structured jobs it found, returned to the orchestrator as an A2A Artifact (a `jobs` data part). The orchestrator reads it off the artifact event, forwards it to the ranker and CV assistant,
Adzuna MCP Server
MCP (FastMCP), httpx
Wraps the Adzuna job board behind a generic search(query) tool over MCP. Fetches matching postings, then enriches each by retrieving its listing page for the full description.
Mock MCP Server
MCP (FastMCP)
Serves canned jobs through the same search(query) tool as Adzuna, so the pipeline can run offline without credentials.
OpenRouter
LLM gateway (OpenAI-compatible API) for LLM reasoning.
Searches Adzuna & fetches listing pages (HTTPS)
Job Board API (Adzuna)
External job listing provider; the Adzuna MCP server queries it for open positions and fetches the listing pages.

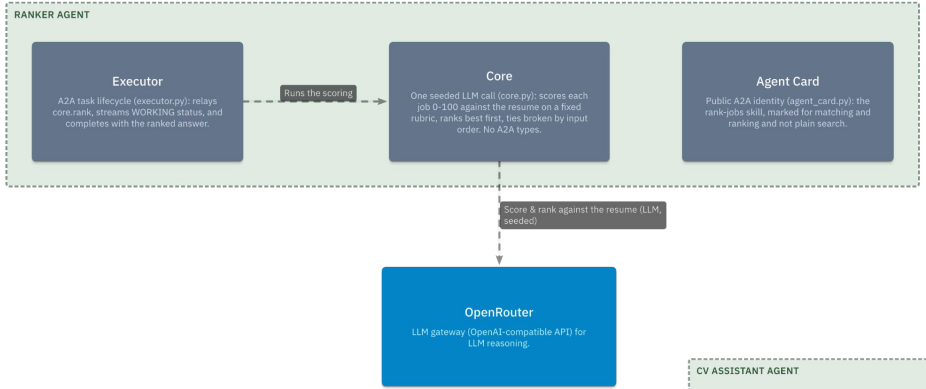
Design rules

- **Protocol apart from logic**: A thin executor speaks A2A; a protocol-free core does the real work. That core is plain Python, so it tests offline without a server.
- **Determinism**: The agents that reason over the resume call the LLM with temperature 0 and a fixed seed, so the same input gives the same output.
- **Best effort, not a guarantee** - it holds only as far as the provider honors the seed.

Command Line Interface

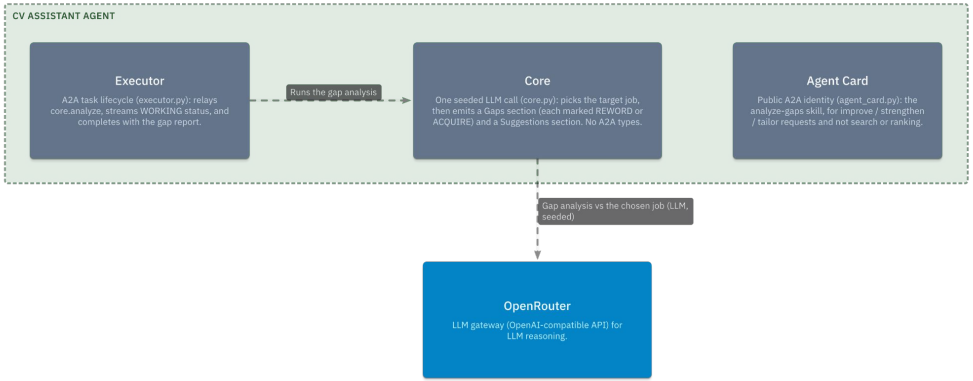


Ranker Agent & CV Assistant Agent



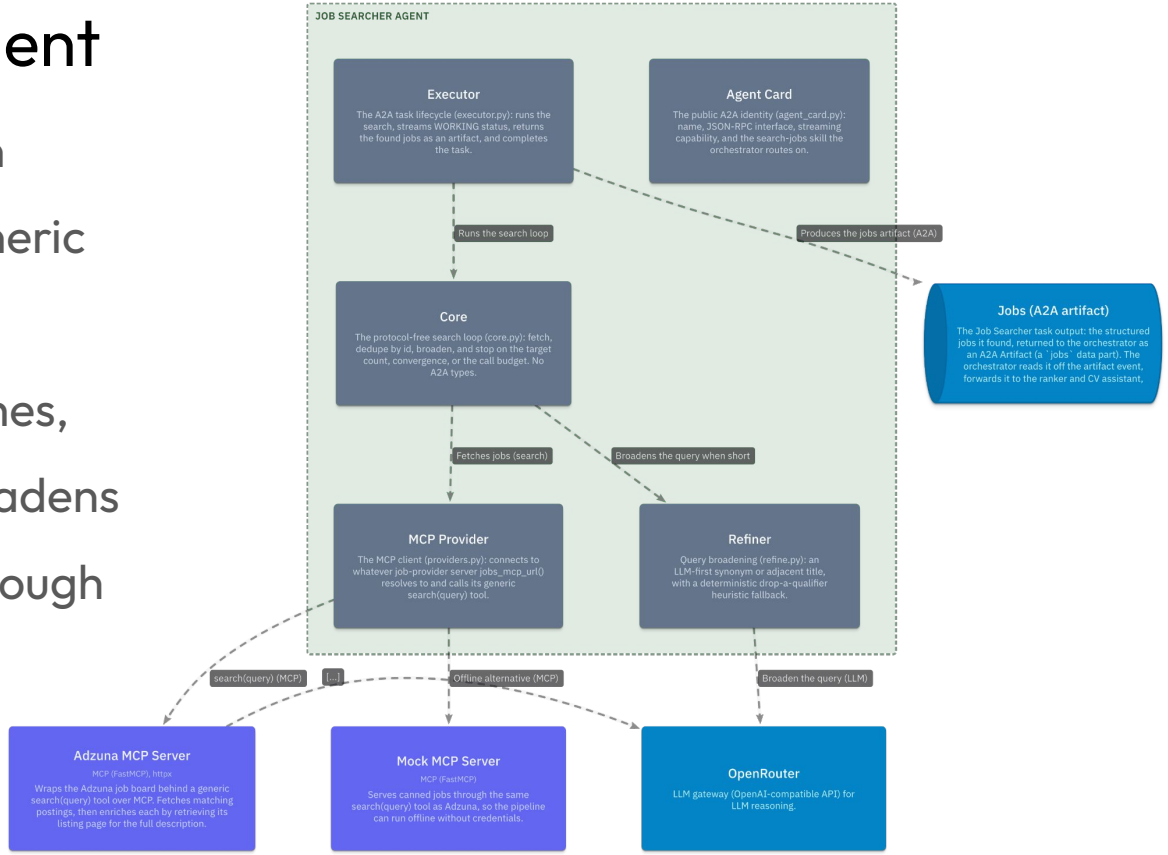
Scores the found jobs against the resume, best first, ties broken by arrival order. It judges only from the text it is given.

Picks one named job, then writes Gaps (REWORD / ACQUIRE) and Suggestions.



Job Searcher Agent

Turns a query into open positions through a generic MCP *search* tool. A goal-seeking loop fetches, dedupes by id, and broadens the query until it has enough or spends its budget.

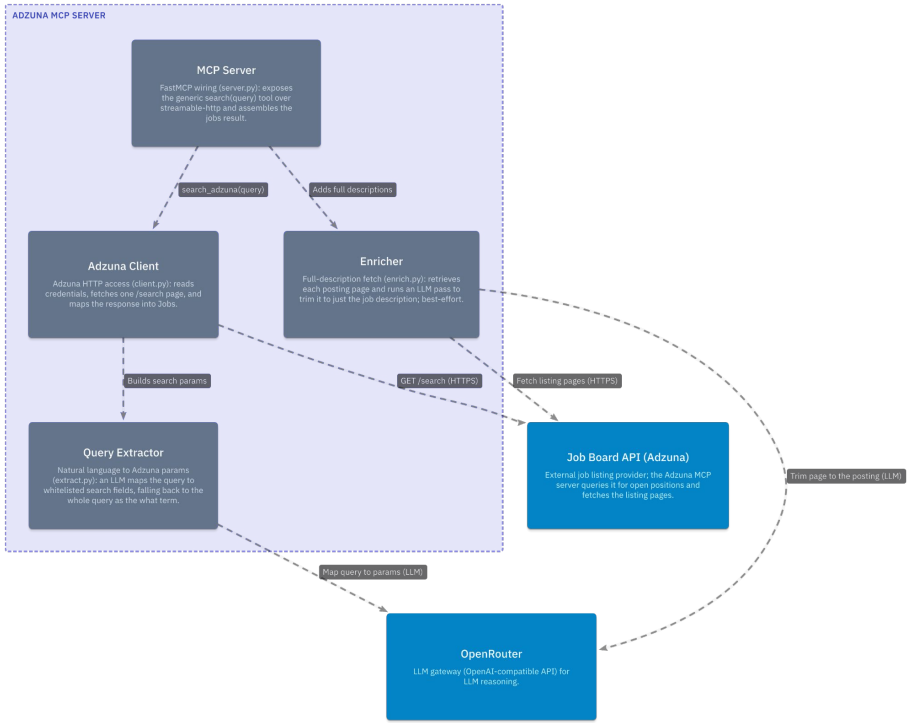


Adzuna MCP

- The one place that talks to the real job board.

- Maps the query to Adzuna's search fields, and enrich jobs scraping each posting for its full description.

- Both LLM steps are best effort: if extraction fails the raw query still runs; if a page cannot be fetched the job keeps its truncated text.



Tech Stack

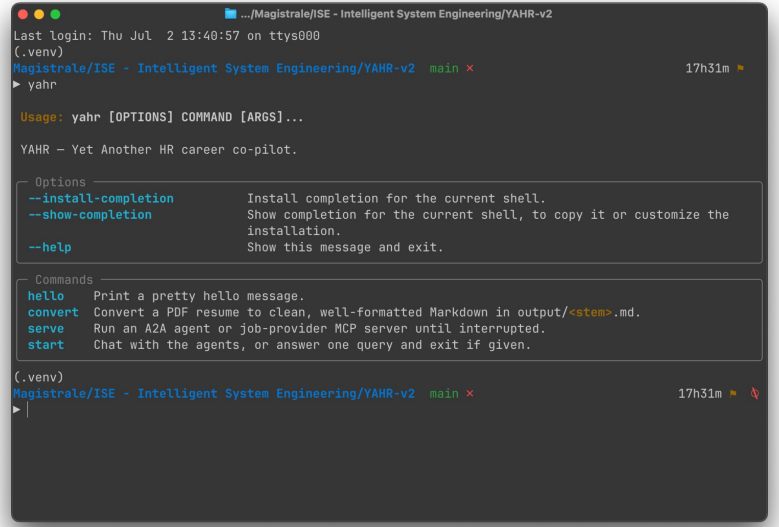
- Python 3.14 + *Typer* + *Rich*
- a2a-sdk (agents are served by *uvicorn*).
- FastMCP
- OpenRouter
- *markitdown*
- *Adzuna's public API*.
- *ruff* and *Prettier* format

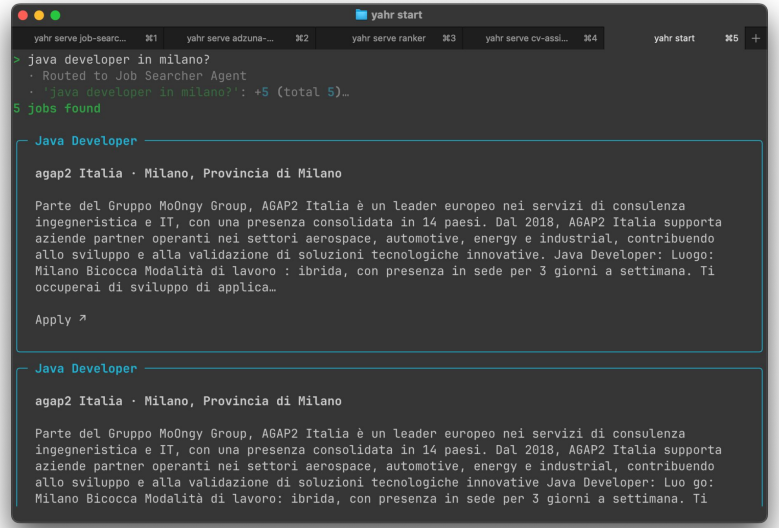
A2A Protocol

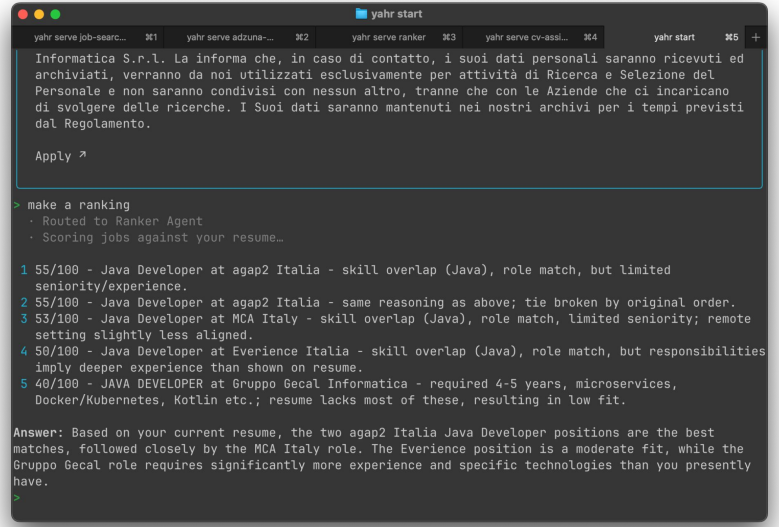
An open protocol for agents to talk over HTTP. It hands YAHR three things for free:

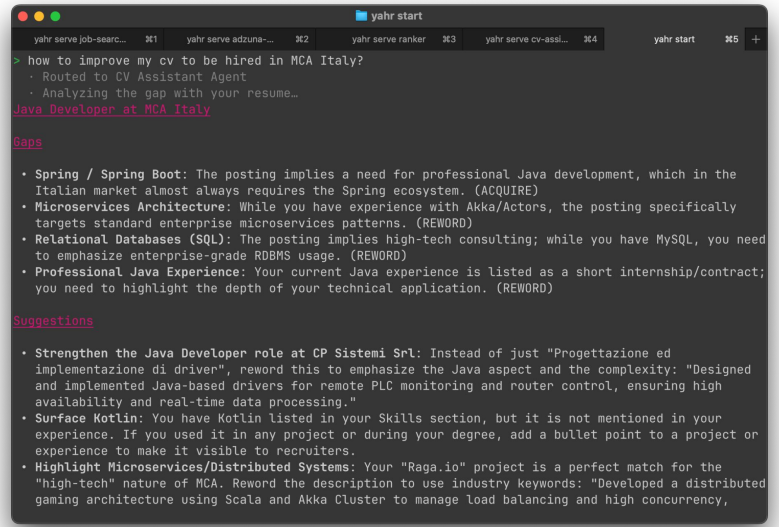
- **Discovery**: every agent publishes an **Agent Card** (identity, url, capabilities, skills). A client reads the card before sending, so it knows who is up and what they do.
- **One shape for every agent**: a request is a **Message** of Parts; real work returns as a **Task** with Artifacts, moving submitted → working → completed.
- **Live results**: cards advertise streaming, so the orchestrator uses *message/stream* and reads Server-Sent Events as the task runs, not one reply at the end.

Testing 💻


.../Magistrale/ISE - Intelligent System Engineering/YAHR-v2
Last login: Thu Jul 2 13:40:57 on ttys000
(.venv)
Magistrale/ISE - Intelligent System Engineering/YAHR-v2 main ×
17h31m
▶ yahr
Usage: yahr [OPTIONS] COMMAND [ARGS]...
YAHR — Yet Another HR career co-pilot.
Options
--install-completion
Install completion for the current shell.
--show-completion
Show completion for the current shell, to copy it or customize the installation.
--help
Show this message and exit.
Commands
hello
Print a pretty hello message.
convert
Convert a PDF resume to clean, well-formatted Markdown in output/<stem>.md.
serve
Run an A2A agent or job-provider MCP server until interrupted.
start
Chat with the agents, or answer one query and exit if given.
(.venv)
Magistrale/ISE - Intelligent System Engineering/YAHR-v2 main ×
17h31m
▶


yahr start
yahr serve job-searc... ⌘1
yahr serve adzuna-... ⌘2
yahr serve ranker ⌘3
yahr serve cv-assi... ⌘4
yahr start ⌘5
> java developer in milano?
· Routed to Job Searcher Agent
· 'java developer in milano?': +5 (total 5)…
5 jobs found
Java Developer
agap2 Italia · Milano, Provincia di Milano
Parte del Gruppo MoOngy Group, AGAP2 Italia è un leader europeo nei servizi di consulenza ingegneristica e IT, con una presenza consolidata in 14 paesi. Dal 2018, AGAP2 Italia supporta aziende partner operanti nei settori aerospace, automotive, energy e industrial, contribuendo allo sviluppo e alla validazione di soluzioni tecnologiche innovative. Java Developer: Luogo: Milano Bicocca Modalità di lavoro : ibrida, con presenza in sede per 3 giorni a settimana. Ti occuperai di sviluppo di applica…
Apply ↗
Java Developer
agap2 Italia · Milano, Provincia di Milano
Parte del Gruppo MoOngy Group, AGAP2 Italia è un leader europeo nei servizi di consulenza ingegneristica e IT, con una presenza consolidata in 14 paesi. Dal 2018, AGAP2 Italia supporta aziende partner operanti nei settori aerospace, automotive, energy e industrial, contribuendo allo sviluppo e alla validazione di soluzioni tecnologiche innovative Java Developer: Luo go: Milano Bicocca Modalità di lavoro: ibrida, con presenza in sede per 3 giorni a settimana. Ti


yahr start
yahr serve job-searc... ⌘1
yahr serve adzuna-... ⌘2
yahr serve ranker ⌘3
yahr serve cv-assi... ⌘4
yahr start ⌘5
Informatica S.r.l. La informa che, in caso di contatto, i suoi dati personali saranno ricevuti ed archiviati, verranno da noi utilizzati esclusivamente per attività di Ricerca e Selezione del Personale e non saranno condivisi con nessun altro, tranne che con le Aziende che ci incaricano di svolgere delle ricerche. I Suoi dati saranno mantenuti nei nostri archivi per i tempi previsti dal Regolamento.
Apply ↗
> make a ranking
· Routed to Ranker Agent
· Scoring jobs against your resume…
1 55/100 - Java Developer at agap2 Italia - skill overlap (Java), role match, but limited seniority/experience.
2 55/100 - Java Developer at agap2 Italia - same reasoning as above; tie broken by original order.
3 53/100 - Java Developer at MCA Italy - skill overlap (Java), role match, limited seniority; remote setting slightly less aligned.
4 50/100 - Java Developer at Everience Italia - skill overlap (Java), role match, but responsibilities imply deeper experience than shown on resume.
5 40/100 - JAVA DEVELOPER at Gruppo Gecal Informatica - required 4-5 years, microservices, Docker/Kubernetes, Kotlin etc.; resume lacks most of these, resulting in low fit.
Answer: Based on your current resume, the two agap2 Italia Java Developer positions are the best matches, followed closely by the MCA Italy role. The Everience position is a moderate fit, while the Gruppo Gecal role requires significantly more experience and specific technologies than you presently have.
>


yahr start
yahr serve job-searc... ⌘1
yahr serve adzuna-... ⌘2
yahr serve ranker ⌘3
yahr serve cv-assi... ⌘4
yahr start ⌘5
> how to improve my cv to be hired in MCA Italy?
· Routed to CV Assistant Agent
· Analyzing the gap with your resume…
Java Developer at MCA Italy
Gaps
• Spring / Spring Boot: The posting implies a need for professional Java development, which in the Italian market almost always requires the Spring ecosystem. (ACQUIRE)
• Microservices Architecture: While you have experience with Akka/Actors, the posting specifically targets standard enterprise microservices patterns. (REWORD)
• Relational Databases (SQL): The posting implies high-tech consulting; while you have MySQL, you need to emphasize enterprise-grade RDBMS usage. (REWORD)
• Professional Java Experience: Your current Java experience is listed as a short internship/contract; you need to highlight the depth of your technical application. (REWORD)
Suggestions
• Strengthen the Java Developer role at CP Sistemi Srl: Instead of just "Progettazione ed implementazione di driver", reword this to emphasize the Java aspect and the complexity: "Designed and implemented Java-based drivers for remote PLC monitoring and router control, ensuring high availability and real-time data processing."
• Surface Kotlin: You have Kotlin listed in your Skills section, but it is not mentioned in your experience. If you used it in any project or during your degree, add a bullet point to a project or experience to make it visible to recruiters.
• Highlight Microservices/Distributed Systems: Your "Raga.io" project is a perfect match for the "high-tech" nature of MCA. Reword the description to use industry keywords: "Developed a distributed gaming architecture using Scala and Akka Cluster to manage load balancing and high concurrency,

YAHR-V3

Harness and Price of Control

- Agents became tools, the orchestrator became the loop — control flow lives in the model; tools are discovered at runtime.
- Errors became observations the model recovers from — search→rank chaining emerged instead of being written.
- Explainability shifted: not why an agent was picked, but the message composed for it, at every hop.
- The price: preconditions moved from `if` to prompt — the model can now decline to act, and did.

Thank you for your attention.