Phase Zero is Pre-filter and Routing:

The goal of this phase is to remove records that are not worth processing.

Task zero point one is Record Filter.

This task removes records that are empty, test or dummy. It also labels records with tags like has_address, has_country, high_priority or low_priority.

The output of this task is a clean input stream.

For example, if the input is "Test Company", "ABC Ltd", "XYZ Corp in Dubai", the filter removes "Test Company" because it is a dummy test record. It tags "ABC Ltd" with has_address equals false and priority equals low. It tags "XYZ Corp in Dubai" with has_country equals true and priority equals high.

Phase One is Name Normalization:

The goal of this phase is to convert the company name to forms that can be used for domain guessing.

Task one point one is Normalize Raw Name.

This task converts the name to lowercase, removes extra spaces with trim, normalizes unicode characters, and removes punctuation marks.

For example, if the input is "ABC Company Ltd.", the output after normalization is "abc company ltd".

Task one point two is Legal Suffix Handling.

This task removes global legal suffixes like ltd, limited, llc, inc, corp, gmbh, sarl, bv, oy, kk, spa, sa, ag, ab, as, nv, pte.

The output is two versions. First is strict_name which is without suffix. Second is relaxed_name which has the suffix removed but can be added back.

For example, if the input is "abc company ltd", the strict_name is "abc company" and the relaxed_name is "abc".

Task one point three is Tokenization.

This task splits the name on spaces and dashes. It removes common stopwords like group, holding, international, but only for strict_name.

For example, if the input is "abc company", the tokens are "abc" and "company".

The output of this phase is a normalized_name_object.

Phase Two is Domain Candidate Generation:

The goal of this phase is to generate a small and prioritized list of probable domains.

Task two point one is TLD Selection.

If the country is known, this task selects the related country code TLD and global TLDs like .com, .net, .biz or .etc. If the country is unknown, it selects only global TLDs.

For example, if the company is "ABC Company" and the country is Germany, the selected TLDs are de for Germany and com and net for global.

Task two point two is Pattern Expansion.

This task creates different domain patterns like name dot tld, name dash company dot tld, name group dot tld, name hq dot tld, name services dot tld.

The ceiling is maximum ten to fifteen domain candidates.

For example, if the name is "abc" and TLDs are de and com, the patterns are abc dot de, abc dot com, abc dash company dot de, abc dash company dot com, abc group dot de, abc group dot com.

Task two point three is Ordering.

This task prioritizes the candidates. First priority is country code TLD plus strict_name. Second priority is com plus strict_name. Third priority is relaxed_name variants.

For example, the ordered list is abc dot de first, abc dot com second, abc company dot de third, abc company dot com fourth, abc dash company dot com fifth, abc group dot de sixth.

The output of this phase is ordered_domain_candidates array.

Phase Three is Fast Domain Validation:

The goal of this phase is to quickly remove dead domains without crawling them.

Task three point one is Bulk DNS Check.

This task checks DNS records like A or AAAA or CNAME. It also detects wildcard DNS and uses a hard timeout.

For example, if the candidates are abc dot de, abc dot com, abc company dot de, the DNS check shows that abc dot de has an IP address and is alive, abc dot com does not exist, abc company dot de has an IP address and is alive.

Task three point two is HTTP Reachability.

This task sends a HEAD request to the domain. If it returns 403 or 405, it sends a limited GET request with bytes range. It rejects parked domains, for-sale pages, and directory or marketplace redirects.

For example, abc dot de opens the website and is valid. abc company dot de shows a domain sale page and is rejected.

The output of this phase is valid_domains array which contains only abc dot de.

Phase Four is Domain Resolution:

The goal of this phase is to make a deterministic decision without artificial intelligence.

Task four point one is Single Domain Resolution.

If there is only one valid domain, this task selects it as canonical.

For example, if valid_domains contains only abc dot de, the canonical domain is abc dot de.

Task four point two is Ambiguity Detection.

If there are multiple live domains or no live domains, this task marks the record as ambiguous.

For example, if valid_domains contains abc dot de and abc dash group dot com, the ambiguity_flag is set to true. If valid_domains is empty, the ambiguity_flag is also set to true.

The output of this phase is resolved_domain or ambiguity_flag.

Phase Five is Web Crawling and Data Extraction.

The goal of this phase is to extract email (if exists phone) from the official website.

Task five point one is Target URL Selection.

This task selects target URLs like the root slash, slash contact, slash about, slash legal, slash impressum for global websites.

For example, if the domain is abc dot de, the target URLs are abc dot de slash, abc dot de slash contact, abc dot de slash about, abc dot de slash impressum.

Task five point two is Async Fetch.

This task fetches maximum five pages with depth two. It uses per-domain rate limit and total byte limit.

For example, it fetches abc dot de slash and abc dot de slash contact simultaneously without blocking.

Task five point three is Extraction.

This task extracts emails using regex, mailto links, and structured data like ld plus json or schema dot org. It optionally extracts phone numbers using phone regex.

For example, from the contact page it finds info at abc dot de, sales at abc dot de, and a phone number.

Task five point four is Filtering.

This task removes emails like noreply at domain and personal free providers like gmail or yahoo unless no other email exists. It classifies emails as role-based, generic, or personal-looking.

For example, it removes noreply at abc dot de and john at gmail dot com, but keeps info at abc dot de and sales at abc dot de.

The output of this phase is emails array, phone, and crawl_status.

Phase Six is AI-assisted Ambiguity Resolution:

The goal of this phase is to resolve deadlocks, not to replace logic.

The entry condition is ambiguity_flag equals true.

Task six point one is Prompt Input Builder.

This task builds the input for the artificial intelligence with company name, address if it exists, and domain candidates if any.

For example, the input is company name "ABC Company", address "Berlin, Germany", domain candidates "abc dot de" and "abc dash group dot com".

Task six point two is Gemini Flash Call.

This task calls Gemini Flash with batch size of fifty. The task is to choose the most likely official domain or return "not found". It does not crawl and does not guess emails.

For example, the prompt is "Which is the official website for ABC Company in Berlin? The candidates are abc dot de and abc dash group dot com." The answer from AI is "abc dot de is more likely the official website."

Task six point three is Decision Handling.

If a domain is returned, this task goes back to Phase Five. If not found is returned, it marks the final status.

For example, if AI returns abc dot de, the system goes to Phase Five to crawl abc dot de. If AI returns not found, the status is marked as not_found.

The output of this phase is resolved_domain or not_found.

Phase Seven is Post-validation:

The goal of this phase is to increase output quality without using large language models.

Task seven point one is Email Syntax Validation.

This task validates email syntax using RFC compliant regex.

For example, it checks that sales at abc dot de has a valid format.

Task seven point two is MX Record Check.

This task checks domain-level MX record existence to confirm the domain can receive emails.

For example, it checks that abc dot de has mail exchange records in DNS.

Task seven point three is Deduplication.

This task removes duplicate emails per company and per domain.

For example, if info at abc dot de appears twice, it keeps only one instance.

The output of this phase is validated_contacts.

Phase Eight is Persistence and Reporting:

Task eight point one is Store Result.

This task stores the final result with company_id, website, emails, phone, status, and source which can be crawler or ai or none.

For example, the stored record is company_id equals 12345, company_name equals "ABC Company", website equals "abc dot de", emails equals "info at abc dot de" and "sales at abc dot de", status equals "success", source equals "crawler".

Task eight point two is Metrics.

This task calculates success rate, ai usage rate, and average latency per phase.

For example, success rate is 85 percent, ai usage rate is 25 percent, average latency per phase is 2 seconds.

Key engineering notes that are very important.

Every phase must be retryable and idempotent. Retryable means if a phase fails, you can run only that phase again without starting from the beginning. Idempotent means if you run a task ten times, the result is the same.

AI call should be less than 30 percent of all records. This keeps costs low.

Crawler should not be a blocker in the pipeline. It should work asynchronously.

No long state should be kept in workers. Workers should be stateless.

Hard but real conclusion.

This architecture is scalable, designed with cost focus, considers large language models as a helping tool not the main engine, and works for global data.

Complete example with all phases:

Input is "Siemens AG, Munich, Germany".

Phase Zero which is filter shows this is a real company so it continues.

Phase One which is name correction produces "siemens".

Phase Two which is domain guessing produces "siemens dot de" and "siemens dot com".

Phase Three which is DNS check shows both are alive.

Phase Four which is decision shows ambiguous equals true because two domains are alive.

Phase Five is skipped because of ambiguity.

Phase Six which is AI help returns "siemens dot de" as probably more official.

Now going back to Phase Five which is crawl and it finds "info at siemens dot de".

Phase Seven which is validation confirms the email is valid.

Phase Eight which is save stores website equals siemens dot de and email equals info at siemens dot de.