An Open-Source Data Infrastructure for Alloy Synthesis Research: Stage 3 — Retrieval-Augmented Literature Advice and CALPHAD Feasibility Screening

Iliya Radulov

October 2026

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Abstract

Stages 1 and 2 of this project built a database, a composition-screening front end, and validated characterization pipelines for a single alloy-synthesis laboratory. Stage 3 adds the two pre-synthesis advisory components promised in the Stage 1 paper: a retrieval-augmented literature advisor and a CALPHAD-based thermodynamic module. The advisor retrieves passages from a locally indexed corpus of the laboratory's materials-science books (72 530 text chunks from 96 books in three topical collections) and generates guidance with a locally hosted language model, isolated from the main application behind a JSON subprocess boundary. A benchmark of 39 questions with known answer passages showed that the default embedding model found the right passage among the top five for only 10 % of the questions, because it reads only the first 256 tokens of each chunk and was trained on English only; embedding the same chunks with a long-context, multilingual model raised this to 51 % and reduced the wrong or missing answers from 19 to 7. Every answer whose context held the right passage was correct. Asking the Russian-source questions in English exposed a language gap: from the advisor's English queries the Russian books were practically unreachable. A Russian version of the query, searched among the Russian passages, now adds one passage from them; with it, five of seven such questions were answered correctly, four of them from the Russian source. The thermodynamic module selects and merges published binary assessments from the open SGTE collection, computes ternary isothermal sections on a 19 701-point composition grid with `pycalphad`, and serves them through a nearest-neighbour index to a web advisor, so that a phase query no longer waits for a live equilibrium calculation of several seconds per temperature. Because a description merged from binaries cannot contain a genuinely ternary phase, the module also includes a feasibility screen that asks how stable such a phase would have to be to appear. For the $Nd_2Fe_{14}B$ composition, the screen places that threshold at a formation energy of about -7.0 to -9.2 kJ mol^{-1} atom^{-1} between 800 and 1500 K, 0.2–0.8 kJ mol^{-1} atom^{-1} more negative than with the binaries of two full ternary assessments. Combined with the SGTE binaries, the $Nd_2Fe_{14}B$ phase of either assessment comes out stable over most of its known range, but by at most 0.6 kJ mol^{-1} atom^{-1}. As in the earlier papers, we emphasize architecture, validation and the engineering defects found while building the system. The most consequential was in the merging of binary descriptions, which silently removed iron from the liquid and the iron solid solutions and counted shared parameters twice; every thermodynamic result looked plausible until the merged description was checked against the binary files it was built from.

1 Introduction

Before a new alloy composition is made, an experimental researcher asks two questions that the Stage 1 and Stage 2 infrastructure could not answer. The first is *what the literature says about making something like it*: which phases to expect, which processing routes have worked, and

which pitfalls are known. The second is *whether the composition is thermodynamically plausible at all* — which phases are stable at the intended processing temperatures and how much liquid forms on the way. Stage 1 already narrows the first question with a four-tier literature-match classification and notes that the third tier — a partial subsystem match, where only part of the chemical system is covered by existing data — is exactly where a thermodynamic check adds the most value ([1], §4.4.2).

Stage 3 answers both questions with two largely independent components. A retrieval-augmented generation (RAG) advisor grounds a locally hosted language model in the laboratory's own book collection. A CALPHAD module computes phase equilibria from published thermodynamic assessments and makes the results available fast enough to be part of routine composition entry. Both components were built by the same single domain-expert developer as the earlier stages and are reported here as software systems rather than as materials-science results.

One design decision shaped the stage. Even a laboratory that has already characterized dozens of samples across different material classes rarely possesses enough homogeneous data to train a reliable property-prediction model from scratch. Building the RAG advisor first was therefore a practical choice: when starting work on a new material, one needs guidance before any samples exist; once the first experimental results arrive, simple prediction and fitting models can begin to close the loop. The advisor is a bridge for the early stages of a research programme, not a permanent substitute for machine learning.

Contributions.

1. A retrieval-augmented literature advisor running entirely on local hardware — corpus preparation including OCR of scanned books, a persistent vector store, composition-aware query construction, and a locally served language model — isolated from the main application by a narrow JSON subprocess interface, together with a reusable benchmark that measures retrieval and answer quality against known answer passages and compares two embedding models on identical chunks (Section 4).
2. A generic, element-order-independent pipeline that selects and merges binary assessments from the open SGTE collection, computes ternary isothermal sections in parallel, and serves them through a nearest-neighbour index to an interactive web advisor (Section 5).
3. A feasibility screen that quantifies the main limitation of binary-merged descriptions — the absence of ternary phases — as a threshold formation energy, verified against a hand-computable reference case before being applied to real data (Section 6).
4. A documented account of the engineering defects found in this stage, each generalized into a lesson in the same format as the Stage 1 paper (Section 7).

The remainder of the paper is organized as follows. Section 2 positions the work relative to existing CALPHAD tooling, open thermodynamic databases, the Fe–Nd–B system, and retrieval-augmented generation. Section 3 gives a system overview. Sections 4–6 describe the RAG advisor, the CALPHAD module and its feasibility screen in detail, Section 7 reports the engineering lessons, Section 8 discusses limitations, and Section 9 concludes.

2 Related Work and Positioning

CALPHAD and open tooling. The CALPHAD method describes the Gibbs energy of every phase in a system with models whose parameters are assessed against experimental and computed data, and obtains phase equilibria by minimizing the total Gibbs energy [3, 4]. Pure-element

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

reference data follow the SGTE unary description [5]. `pycalphad` [6] provides an open-source Python implementation of the equilibrium calculation and is the engine used here; ESPEI [7], built on it, supports parameter fitting and is the natural route for the project's own future assessments (Section 9).

Open thermodynamic databases. Assessed parameter sets are increasingly available in machine-readable form. The Thermodynamic Database Database indexes published TDB files [8], and the SGTE collection of binary datasets [9] provides 2 773 datasets for about 1 238 binary systems under a Creative Commons license. This project uses the latter as its sole thermodynamic source. Its breadth is what makes a generic "any three elements" tool possible; its restriction to binaries is what makes the feasibility screen of Section 6 necessary.

The Fe–Nd–B system. The hard-magnetic phase $Nd_2Fe_{14}B$ (Φ) is a ternary compound [10, 11], and dedicated ternary assessments that include it exist [12, 14, 17]. We do not compete with those assessments. Fe–Nd–B serves here as the validation case for a tool intended for systems that have *no* dedicated ternary assessment, precisely because its true phase relations are well known.

Retrieval-augmented generation. Retrieval-augmented generation (RAG) combines a retriever over a document collection with a generative model, so that answers are conditioned on retrieved passages rather than on model parameters alone [19]. Our implementation uses standard open components: a persistent vector store (ChromaDB [31]), sentence-embedding models (the compact all-MiniLM-L6-v2 [26, 27] and the multilingual, long-context bge-m3 [28]), a local model server (Ollama [32]), and an open-weight language model (Qwen2.5 [25]). The contribution is not a new retrieval method but its integration into a laboratory workflow under the constraint that no laboratory data or queries leave local hardware.

Language models in materials science. Language models are being applied to materials science and chemistry in many ways; the report of one hackathon alone collected fourteen applications, from property prediction to knowledge extraction and new interfaces to laboratory tools [20]. Closest to the advisor are systems that ground a model's answers in a corpus. PaperQA retrieves and cites passages from scientific papers [21], and its successor was reported to match or exceed human experts on literature-research tasks [22]; LLaMP grounds answers in the data of the Materials Project [23]. Earlier, text mining extracted nearly 20 000 solid-state synthesis reactions from the literature [24]. These systems work with journal articles or public databases. The advisor described here works offline on a laboratory's own book collection, much of it scanned and partly in Russian, and its interest lies less in retrieval quality as such than in what this setting exposes: silent truncation, colliding identifiers and a language gap (Section 7).

Positioning. The system sits between a general electronic laboratory notebook and a complete workflow for thermodynamic assessment. It does not produce new assessments. It makes existing assessments, and an existing book collection, usable as a routine step of composition entry for one laboratory.

3 System Overview

Figure 1 shows the two Stage 3 components and how they attach to the Stage 1/2 database. They share a common entry point — a composition the researcher is considering — but share neither code, environment, nor data store; either component can be used independently of the other.

Hardware roles. The three-machine layout described in the Stage 1 and Stage 2 papers [1, 2] is what makes Stage 3 practical. The always-on server holds the canonical database. A GPU-

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

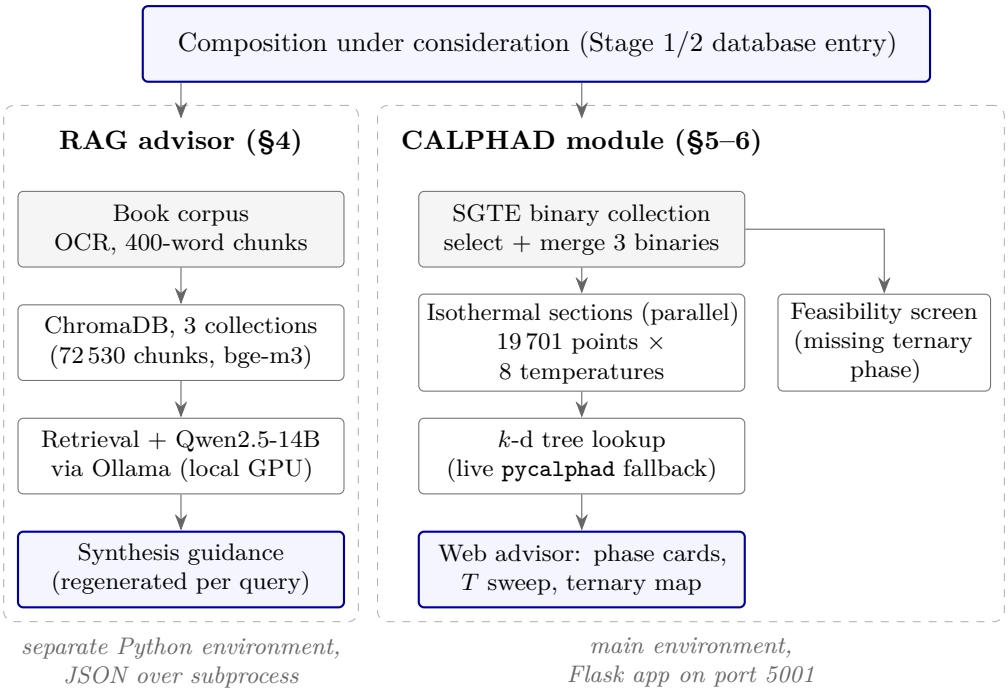


Figure 1: Stage 3 architecture. Both components start from a composition under consideration and run independently. The RAG advisor lives in its own Python environment and is called through a JSON subprocess interface. The CALPHAD module precomputes isothermal sections offline and serves them through a nearest-neighbour index, falling back to a live equilibrium calculation for compositions or temperatures it does not cover.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

equipped desktop runs the language model and the embedding model and, because it also has the most CPU cores (12 cores, 24 threads), performs the offline grid calculations. The laptop serves as the interactive client. The grid calculations are the only part of the CALPHAD module that requires the large machine; the web advisor runs anywhere the cached grid files are present.

Environment isolation. The RAG stack (vector store, embedding model, and model-server client) has dependency requirements that conflict with the rest of the application — a conflict that surfaced during a project-wide move from Python 3.14 back to 3.13. Rather than forcing a single environment to satisfy both sets of constraints, the RAG stack lives in its own virtual environment and is invoked by the main application as a subprocess that returns JSON. The CALPHAD module runs in the main environment (Python 3.13, `pycalphad` 0.11.2, Flask). This is the same isolate-rather-than-reconcile policy used throughout the project.

Scope boundary. Stage 3 as reported here ends at advice: it informs the decision to synthesize a composition but does not record that decision or its outcome. Both components are validated as standalone tools. Wiring the RAG advisor into the web advisor, and both into the daily-use desktop application, remains open work (Section 9).

4 Retrieval-Augmented Literature Advisor

4.1 Corpus and indexing

The corpus is the laboratory's own collection of physics and materials-science books, sorted by hand into topical folders; 96 of them are indexed here. Each folder is indexed as one collection: phase stability, synthesis, and magnetic properties. (A fourth, small collection on powder

metallurgy was an early test and is not used here.) A substantial fraction of the books were scanned rather than born-digital, many of them in Russian, and had to pass through optical character recognition before they could be indexed. This was done with `ocrmypdf` [33] and Tesseract, using Russian and English language models, as unattended batch jobs. Eight scans had been missed at first; the corpus diagnostics of Section 4.4 found them, and they were recognized and indexed in October 2026.

Text is split into chunks of 400 words with a 50-word overlap; the first three chunks of every book (mostly front matter) are skipped. Each chunk's identifier contains a hash of the book's complete file name, so it is unique (Section 7.2.2). The three collections hold 4 660 chunks from 10 books (phase stability), 27 881 from 37 books (synthesis) and 39 989 from 49 books (magnetic) — 72 530 chunks in total — stored in ChromaDB [31]. Every chunk is embedded twice, into two parallel sets of collections: with ChromaDB's default model, all-MiniLM-L6-v2 [26, 27], which reads at most 256 word-piece tokens of a text and was trained on English, and with bge-m3 [28], a multilingual model that reads up to 8 192 tokens and is served by Ollama on the GPU. Chunks, identifiers and metadata are identical in both sets, so that the two models can be compared directly (Section 4.4). Fifty chunks are longer than 2 048 tokens, the longest 6 391. They are embedded in full: the embedding call requests a context window that holds the longest chunk and forbids shortening, and a re-embedding of the 20 longest chunks of each collection reproduced the stored vectors (lowest cosine similarity 1.000). The embedding model is stored with each collection, so queries are embedded with the same model as the chunks.

This index was rebuilt from the books in October 2026. Its predecessor held 84 books: its identifiers collided, which removed some books entirely, and it lacked the eight scans; and its bge-m3 version had read only the first 2 048 tokens of each long chunk (Sections 7.2.2 and 7.3.2). The rebuild changed the benchmark results of Section 4.4 only slightly.

ChromaDB was chosen over a simpler in-memory vector index because of a lesson from an earlier, unrelated RAG project: an in-memory index loads its entire contents into RAM at start-up, which does not scale to a corpus of this size; a persistent store queries from disk without a full load. Neither the store nor the models have to run on the machine that asks: Ollama is itself a network service, and ChromaDB can run as a server, so that a laptop can act as a thin client that sends its queries to the GPU machine instead of loading the models itself. Another application in the laboratory already uses the GPU machine in this way; the advisor does not yet (Section 9).

4.2 Query construction and generation

The advisor does not take a free-text question. It builds its query from the sample: the chemical system, taken from the sample identifier; the processing routes suggested by the Stage 1 screening; and the composition, of which the elements above 5 at.% are named. For an Mn–Al–C sample suggested for mechanical alloying, the query reads: "What synthesis parameters and conditions are recommended for Mn-Al-C alloy (Al (52 at%), Mn (47 at%)) prepared by mechanical alloying? Include temperatures, atmosphere, milling parameters, and any known challenges for this system." The query is embedded with the collection's own model, the four nearest passages of the synthesis collection are retrieved together with a fifth from its Russian-language books (below), and passages and query are passed to Qwen2.5-14B [25], served locally by Ollama [32] on the GPU machine. The prompt asks the model to answer only from the passages, to say so clearly when they do not contain enough information, and not to invent data, parameters or references. The model runs at temperature 0 with a context window of 16 384 tokens, set explicitly. Before each call the prompt length is estimated and the window widened if necessary, and the number of prompt tokens the model actually read is returned with the answer (Section 7.3.2). No query, sample or composition leaves local hardware.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

A second language. About a tenth of the corpus is in Russian (7 005 of the 72 530 chunks), and an English query practically never reaches it (Section 4.4). The advisor therefore builds the same query a second time from a Russian version of the template, with fixed Russian names for the seven routes the screening can suggest. Every chunk carries its language in the metadata, and the Russian query is searched only among Russian chunks; the nearest one becomes the fifth passage. The four passages for the English query are unchanged, and the question given to the model stays in English. A route without a Russian name is kept in English inside the Russian sentence, and the reply says so.

The generated guidance is shown to the researcher but deliberately not stored in the database. It is regenerated on every request, so that it improves automatically as the corpus grows instead of freezing the corpus state at the time of first entry. The cost of this choice — an answer cannot later be reproduced exactly — is discussed in Section 8.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

4.3 Integration boundary

When a sample is saved to the database and the Stage 1 screening has suggested at least one synthesis route — also when it judges melting unsuitable, for example because of a volatile element, and proposes powder routes instead — the main application calls the advisor as a subprocess, with the Python interpreter of the RAG environment, and reads a single JSON object from its standard output. For compositions the screening cannot assess, there is therefore no literature guidance. The interface is deliberately narrow. The call passes one JSON argument with three fields: the system name, the suggested routes and the composition as mole fractions. The reply contains the answer; the query that was sent and its Russian version; the retrieved passages with the file names of their books, their distances and which query found them; the collection searched and the model used; the number of prompt tokens read; and a success flag with an error message. Errors inside the advisor — a missing collection, an unreachable model server — are caught and returned in the same object, so that the caller receives a readable message instead of a traceback; only a malformed call ends with a non-zero exit code. The application waits at most 120 s and shows the answer followed by the books it was drawn from. In a test from the application on 23 October 2026, an $Al_{45}Mn_{55}$ sample with the suggested route "arc/induction melting" received, after about a minute, an answer stating that the four passages describe arc melting and induction heating in general but give no conditions for this composition, followed by the four books. The benchmark of Section 4.4 imports retrieval, prompt, model settings and generation from the same module, so it evaluates the code the application runs; only the query differs. A defect at exactly this boundary is reported in Section 7.3.1, and a more serious one — the advisor that the application ran was not the one evaluated — in Section 7.3.3.

4.4 Evaluation

The advisor was first checked qualitatively on two queries whose answers the researcher knew. Both looked right, but such a check cannot say how often the right passage is found. We therefore built a benchmark that can.

Benchmark. The benchmark has 46 questions. The author wrote 41 of them from the indexed books, each answered by a specific passage, and copied from that passage a verbatim *gold phrase*. A chunk counts as a gold chunk if it contains the phrase; the comparison ignores case, line-break hyphenation, ligatures and spacing, so that it tolerates the artefacts of PDF text extraction. Every question names the book that answers it, so the benchmark can distinguish a passage that is not found from a book that is not in the index. Seven of these questions are answered by a Russian-language source. The other five questions ask for something no book contains — a

Table 1: RAG benchmark on the rebuilt index: the same chunks embedded with all-MiniLM-L6-v2 and with bge-m3. Retrieval: 39 questions with a known gold passage, top-10 retrieval in the collection that holds it; 95 % Wilson intervals [30] in brackets. Answers: Qwen2.5-14B at temperature 0 with a 16 384-token context, from the top four chunks, graded by the author (see text); correctness on the 39 scored questions, faithfulness on all 44 answers (the 39 and five out-of-corpus questions).

	all-MiniLM-L6-v2	bge-m3
Retrieval		
hit@1	0.05	0.33
hit@3	0.10	0.41
hit@5	0.10 [0.04, 0.24]	0.51 [0.36, 0.66]
hit@10	0.10	0.62
MRR	0.07	0.40
Answers (of the 39 scored questions unless stated)		
correct	17	24
correct or partly correct	20	32
wrong or no answer	19	7
of which: declined although the books contain the answer	15	4
correct when the gold passage was among the four	4 of 4	18 of 18
answers with claims not in their passages (of 44)	3	4
out-of-corpus questions correctly declined (of 5)	5	4

temperature from the laboratory's own records, prices and production figures from 2025, and the stable phases of a binary system that the reference handbook does not list — to test whether the advisor declines to answer; one was written by the author, four were drafted with an AI assistant and kept by the author.

Of the 41 answerable questions, two were dropped because their phrases turned out to be the wording of an earlier advisor answer rather than of a book. Five phrases that matched only partly were shortened to the part found in the indexed text. The benchmark therefore scores retrieval on 39 questions; one of them could be scored only after the rebuild, because its book had been lost from the earlier index (Section 7.2.2).

For each question the collection holding the gold book is searched and the rank of the first gold chunk among the ten nearest chunks is recorded; hit@k is the share of questions with a gold chunk among the top k, and MRR the mean reciprocal rank. The same chunks and the same questions are used for both embedding models, so any difference in retrieval is due to the model alone.

Note on the metric. The gold-phrase definition is strict: a chunk is counted only if it contains the exact phrase. Other chunks that express the same fact in different wording are therefore not credited, so the metric understates how often useful passages are retrieved; the answer grades below give a complementary view.

Corpus diagnostics. Before the benchmark, a read-only diagnostic measured what the original embedding model actually sees. The median chunk is 592 to 631 tokens long, depending on the collection, so nearly every chunk exceeds the 256-token limit, and on average 59–62 % of a chunk's text never reaches the embedding. Russian text (3.5 % of the synthesis and 9.2 % of the magnetic collection) was also outside the model's training. The diagnostic also listed the books that had no chunks at all, which led to the eight unindexed scans and to the identifier collisions of Section 7.2.2.

Retrieval. Table 1 and Figure 2 give the results. With all-MiniLM-L6-v2 a gold chunk was among the top five for 4 of 39 questions (hit@5 = 0.10); with bge-m3 for 20 (0.51). The difference is not a matter of a few questions: bge-m3 found 17 gold chunks that MiniLM missed, and

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

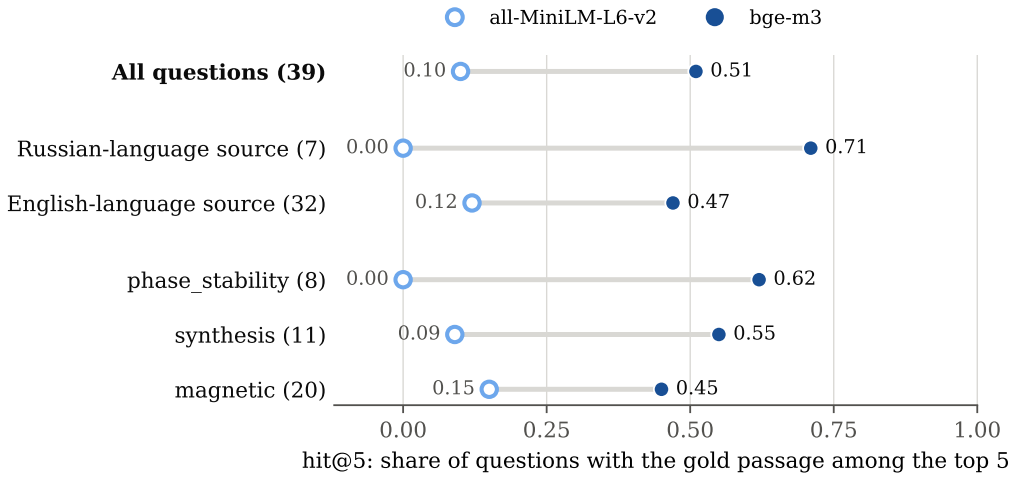


Figure 2: Retrieval hit@5 by subgroup (number of questions in brackets) on the rebuilt index, for the same chunks embedded with all-MiniLM-L6-v2 and with bge-m3.

MiniLM one that bge-m3 missed (exact McNemar test [29], $p \approx 1 \times 10^{-4}$). On the predecessor index the two models had reached 0.13 and 0.50 on 38 questions; the rebuild mainly restored the question whose book had been lost, which bge-m3 now finds at rank 2.

The diagnostic explains most of the difference. For 28 of the 39 questions the gold phrase lies in the part of its chunk that MiniLM never embeds; for these, MiniLM's hit@5 is 0.04, against 0.27 when the phrase lies within the first 256 tokens. The seven Russian-language questions go from 0 of 7 to 5 of 7. All three collections improve — phase stability from 0.00 to 0.62, synthesis from 0.09 to 0.55, magnetic from 0.15 to 0.45 — and the magnetic collection, the largest, improves least and remains the weakest.

Answers. Each question was also answered by the language model from its top four chunks, with the advisor's prompt and settings (Section 4.2). Every answer was graded on three criteria: *correct* (it contains the information of the gold passage: yes, partly or no), *faithful* (its specific facts, numbers, names and formulas appear in the four passages), and *declined*. The grades were drafted by an AI assistant that checked each doubtful claim against the passages and were reviewed by the author. The answers were generated three times: with the model server's default context window, which cut the ten longest prompts to about half without an error (Section 7.3.2); with full prompts on the predecessor index; and with full prompts on the rebuilt index. All answer numbers reported here come from the last run; answers that did not change between runs kept their grades.

Four observations stand out. First, the clearest difference between the two embedding models is in wrong or missing answers: 19 with MiniLM, 7 with bge-m3. Correct or partly correct answers rose from 20 to 32; 14 questions improved and two got worse (exact McNemar test, $p \approx 0.004$). Fully correct answers rose from 17 to 24, a difference within the noise of 39 questions (12 against 5, $p \approx 0.14$). Second, whenever the gold passage was among the four passages given to the model, the answer was correct: in 4 of 4 cases with MiniLM and in 18 of 18 with bge-m3, including all five Russian-language questions whose passage bge-m3 found. Every wrong answer came from a question whose gold passage was not given. With full prompts, retrieval, not generation, limits the advisor. Third, when the passages did not contain the answer, the model mostly said so: with MiniLM's passages it declined 15 times, and only three answers with MiniLM and four with bge-m3 contained claims not in their passages. Most are slips, such as a niobium alloy given as a tantalum alloy; two, one per model, are inferences the passages do not support, such as grain sizes inferred from density. MiniLM's passages still led to 13 correct answers without the gold

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

passage, because other passages often carry the same fact. Fourth, the language of the answers is not under control: with bge-m3 all seven Russian questions were answered in Russian, but with MiniLM's passages four were answered in English and one in Chinese, and in an earlier run one answer switched to Chinese in mid-sentence.

Questions in the other language. The benchmark asks each question in the language of its source, but the advisor always asks in English. To see whether that matters, the seven questions with a Russian source were asked again in English, with the same gold passages. Asked in Russian, bge-m3 placed the gold passage first for five of them; asked in English, it found none, and 27 of the 28 passages it gave the model came from English books. A multilingual embedding model places texts of different languages in one space, but texts in the same language on a similar topic lie closer together than the right passage in the other language, and the English books cover most topics. Restricting the search to Russian chunks while keeping the English question helped little: the gold passage was the nearest Russian chunk for only one of the seven. Translating the query with the language model was tried and rejected. In a check of all 44 questions, five translations into Russian drifted into Chinese characters even after a stricter second attempt, and others used wrong terms — a thermal effect for the caloric effect, cross-linked for sintered magnets. The advisor therefore uses a fixed Russian template (Section 4.2). Its effect was measured with the original Russian wording of the seven questions as the Russian query: the fifth passage was the gold passage for five of the seven, the same five that the Russian questions themselves found. The model used it: four of these five answers were drawn from the Russian passage and written in English. Of the seven English questions, five were now answered correctly and two partly, against one correct, four partly correct and two declined without the fifth passage.

An extra passage could also do harm where it is not needed. All 44 questions were therefore run with a fifth passage as well, searched with the question itself among the chunks of the other language; since the benchmark questions have no translation, it was mostly unrelated. (In the phase-stability collection, which holds no Russian book, the fifth passage is simply the fifth-nearest.) The grades did not change in total — 24 correct, 8 partly correct and 7 wrong or missing answers, as with four passages: one answer improved and one got worse. The out-of-corpus question on Fe–Bi, answered about Fe–B with four passages (Section 7.2.5), was now declined. One more answer contained claims not in its passages, five of 44, among them one that inverted a passage on hydrogen in rare-earth compounds.

Two practical details matter for anyone repeating the evaluation. An automatic check for declined answers, based on phrases such as "not enough information", fired on unrelated wording ("insufficient efficiency") and missed declines written in Russian, so manual grading cannot be replaced by it. And answers are reproducible only with identical settings: changing nothing but the context window changed 32 of the 44 answers in each run, also where no prompt had been cut, although the model ran at temperature 0. Conversely, identical prompts gave identical answers: the 60 prompts that the rebuild of the index left unchanged produced word-for-word the same answers as before. One out-of-corpus question exposed a distinct failure mode (Section 7.2.5).

Status. The advisor is usable inside the development version of the desktop application. Since 23 October 2026 it runs the module described in Section 4.2 (the bge-m3 synthesis collection of the rebuilt index); since 24 October it also includes the fifth passage from the Russian books. Before 23 October the application could not run the advisor at all (Section 7.3.3).

Two limitations remain. First, the advisor currently searches only the synthesis collection; automatic routing to the phase-stability or magnetic collections is not yet implemented, and the advisor is not wired into the web interface of Section 5. Second, the benchmark uses free-text questions, whereas the advisor in the application builds composition-aware queries; how retrieval quality transfers to those queries has not been measured.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Table 2: Merged Fe–Nd–B description used throughout this paper.

Binary datasets (SGTE collection [9])	`FeNd-16Che` [15], `FeB-20Tak` [16], `NdB-19Che` [17]
Phases after merging (14)	`LIQUID`, `FCC_A1`, `BCC_A2`, `HCP_A3`, `DHCP`, `BETA_RHOMBO_B`, `M2B_C16` (Fe_2B), `MB_B27` (FeB), `FE17ND2`, `FE17ND5`, `ND2B5`, `NDB4_D1E`, `NDB6_D21`, `NDB66`
Not representable	$Nd_2Fe_{14}B$ (Φ) and any other ternary compound

5 CALPHAD Module

5.1 Selecting and merging binary assessments

The module takes any three elements, finds the assessments of the three binary subsystems in a local copy of the SGTE collection [9], merges them into one ternary description, and computes equilibria from it. The collection's file names encode the element pair, the year and the first author of the underlying assessment (for example `FeNd-16Che`). An index (`tdb_lookup.py`) maps each unordered element pair to its candidate files, sorted newest first, with a file marked `-corrected` preferred over the uncorrected file of the same year; a Landolt–Börnstein variant can be preferred instead when one is available. A missing pair is reported to the caller, which decides whether to proceed.

Merging is performed on the parsed databases (`tdb_merge.py`): each binary file is loaded by `pycalphad`, and its elements, functions, phases and parameters are combined into a single database, which is then written as a TDB file. A function is taken from the first file that defines it; a conflicting definition in another file is reported. A phase that occurs in several files must have the same site ratios in all of them and receives the union of their constituents on each sublattice (the liquid, for example, then contains all three elements). Each parameter is kept once; a duplicate with a different value is reported. The files are sorted by name before merging, so that the result does not depend on the order in which the user types the elements (Section 7.1.5).

The merged description is checked against its sources: along each edge of the composition triangle it must reproduce the binary file it came from — the same stable phases and the same Gibbs energy at 30 points per edge (800, 1200 and 1600 K, ten compositions each). The grid generator stops if this check fails. For Fe–Nd–B the phases agree at every point, and the largest energy difference rounds to 0.0 $\mathrm{J\,mol^{-1}}$ on all three edges. The first version of the merge, which combined the files as text, failed this check by up to 155 $\mathrm{kJ\,mol^{-1}}$ (Section 7.1.1).

For Fe–Nd–B the selected files and the resulting description are listed in Table 2. The merged description has fourteen phases. It contains the binary compounds $Fe_{17}Nd_2$, $Fe_{17}Nd_5$, Fe_2B, FeB and the Nd borides, but no $Nd_2Fe_{14}B$, because none of the three binary assessments can describe a ternary compound. The merge reports two conflicts between the files: the Fe–B dataset models β-boron with two sublattices (101:4), so that it can dissolve some iron, whereas the Nd–B dataset treats it as a pure element; and the two datasets give the hypothetical fcc Nd different lattice stabilities. In both cases the first file's version is kept, and the edge check shows that the Nd–B diagram remains unchanged.

5.2 Isothermal sections on a fixed grid

An isothermal section is computed on a regular grid over the Gibbs triangle. With $N = 200$ divisions, the grid contains every composition whose three mole fractions are multiples of $1/200$ and all at least $1/200$, which gives $\binom{199}{2} = 19\,701$ interior points. At each point a single equilibrium is computed at the given temperature and atmospheric pressure. Phases with a phase fraction

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Table 3: Grid statistics for the Fe–Nd–B sections at 800 and 1000 K (19 701 points each), as printed by the generator; — = not among the ten largest fields.

	800 K	1000 K
Not converged	54	50
One-phase points	0	13
Two-phase points	205	736
Three-phase points	19 442	18 902
`DHCP+M2B_C16+ND2B5`	9 403	9 401
`DHCP+FE17ND5+M2B_C16`	5 031	—
`FE17ND5+LIQUID+M2B_C16`	—	3 441
`MB_B27+NDB66+NDB6_D21`	1 160	1 096
`FE17ND2+FE17ND5+M2B_C16`	798	798
`BCC_A2+FE17ND2+M2B_C16`	657	657
`MB_B27+NDB4_D1E+NDB6_D21`	549	549
Equilibrium step	7.5 min (24 proc.)	8.5 min (16 proc.)

below 10^{-3} are discarded, and a point counts as converged if at least one phase remains.

Coordinate convention. For elements given in the order (E_1, E_2, E_3), the two independent coordinates stored for each point are $x_1 = x_{E_1}$ and $x_2 = x_{E_2}$, with $x_{E_3} = 1 - x_1 - x_2$; these are the two independent mole fractions fixed as conditions of the equilibrium. For the Fe–Nd–B grids, computed with the order (Fe, Nd, B), this means $(x_1, x_2) = (x_{\mathrm{Fe}}, x_{\mathrm{Nd}})$. We state this explicitly because getting it wrong was one of the most consequential defects of the stage (Section 7.1.2).

Parallelization. The points are independent and are distributed over all CPU cores with Python's `multiprocessing`. Each worker process loads the merged database once in the pool initializer and then processes points one at a time; loading the database per point would dominate the run time. On the GPU desktop (12 cores, 24 threads) the equilibrium step for one full 19 701-point section took 7.5 min at 800 K with 24 worker processes and 0.8–8.5 min at the other seven temperatures with 16 processes. With the defective merge of Section 7.1.1 the same step had taken only 0.7 min; the liquid and the solid solutions now contain all three elements, which makes each equilibrium more expensive.

Each section is cached as a pickle file together with PNG renderings of the section and of the phase-fraction maps of `FE17ND2` and `LIQUID`. Sections exist for eight temperatures (800, 1000, 1100, 1200, 1300, 1375, 1450 and 1600 K), chosen around the sintering range assumed by the advisor (1300–1450 K). Figure 3 shows the section at 1375 K.

Table 3 summarizes the sections at 800 and 1000 K. Of the 19 701 points, only 54 (800 K) and 50 (1000 K) did not converge (0.3 %). At 1100–1600 K between 0.4 and 1.0 % of the points did not converge, except at 1450 K, where 936 (4.8 %) did not. At both temperatures almost all converged points lie in three-phase fields, as expected for a ternary system at fixed temperature and pressure. At higher temperatures the liquid field expands; at 1600 K 6 238 points are fully liquid. At both 800 and 1000 K the $Nd_2Fe_{14}B$ composition lies in the `FE17ND2+FE17ND5+M2B_C16` field, while the largest field (`DHCP+M2B_C16+ND2B5`) covers almost half of the triangle.

5.3 Serving precomputed results

A live equilibrium calculation for one composition takes a few seconds per temperature — too slow for an interactive tool that evaluates seven temperatures per query. The web advisor therefore answers from the cached grids first. On first use each grid is loaded and a k-d tree [34] (`scipy.spatial.cKDTree` [35]) is built over its converged points in the $(x_{\mathrm{Fe}}, x_{\mathrm{Nd}})$ plane (1–2 s in

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

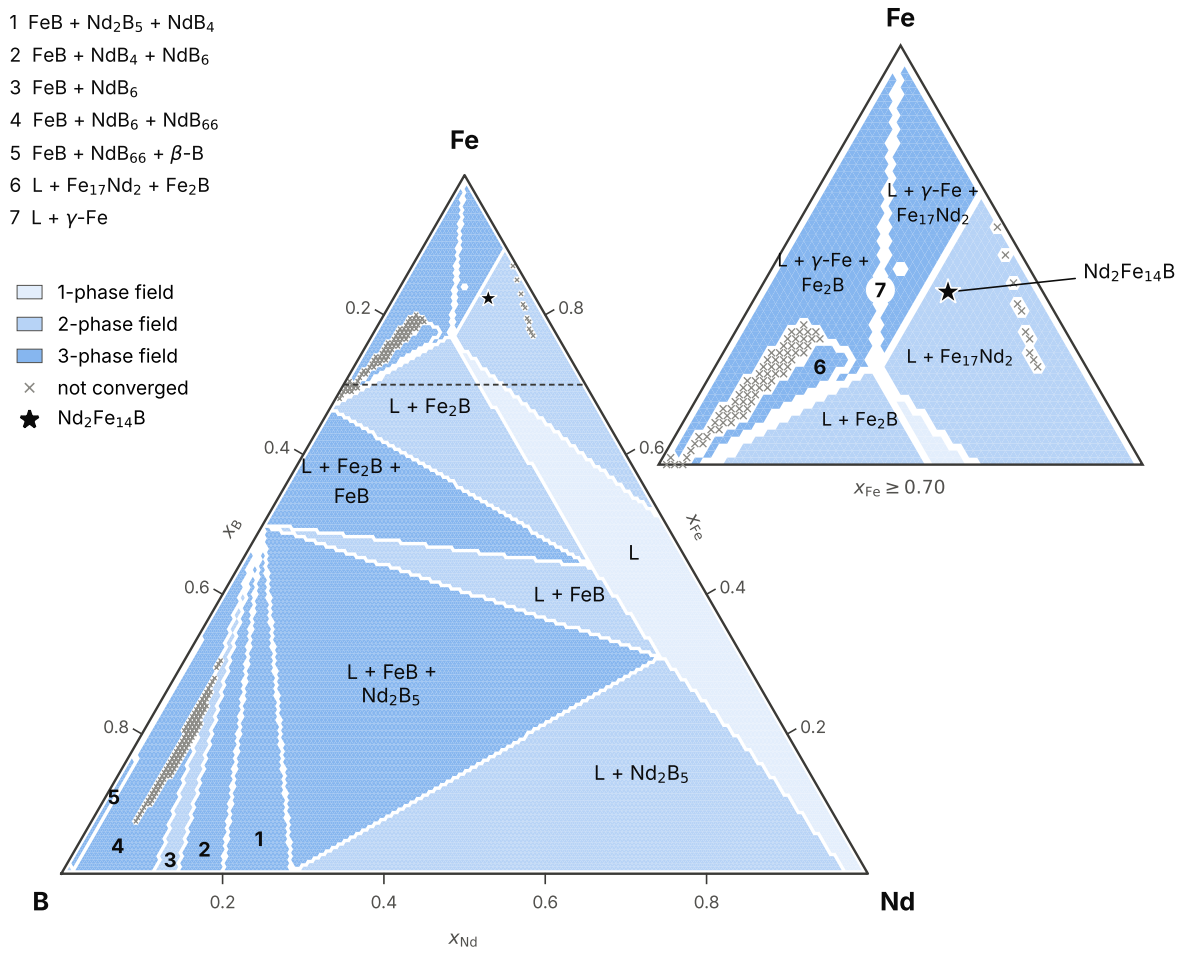


Figure 3: Fe–Nd–B isothermal section at 1375 K computed from the merged binary description (Table 2) on the 19 701-point grid. Fields are shaded by the number of coexisting phases and separated by white lines; grey crosses mark points that did not converge. Large fields are named, small ones numbered (key, upper left). The inset magnifies the Fe-rich corner ($x_{\mathrm{Fe}} \geq 0.70$, dashed line), where the $Nd_2Fe_{14}B$ composition (star) lies in the two-phase field of $Fe_{17}Nd_2$ and liquid. Three white dots are grid points on the line $Fe_{17}Nd_2$–`M2B_C16` that returned only these two compounds. One of them ($x_{\mathrm{Fe}} = 0.84$, $x_{\mathrm{Nd}} = 0.08$) is a missed minimum: the assemblage of its neighbours, `FCC_A1`+$Fe_{17}Nd_2$+`LIQUID`, is 15 J/mol lower. Because the point's composition equals a mixture of the two compounds exactly, the minimiser stopped there.

total). A query returns the phase fractions of the nearest converged point provided it lies within a distance of 0.015 (three grid spacings); otherwise the caller falls back to a live `pycalphad` calculation. The merged database for the fallback is built only when a query actually misses and is then kept in memory. Every result carries a tag indicating whether it came from the grid or from a live calculation.

When the grids are loaded, a check confirms the coordinate convention by boron mass balance. For every converged point whose phases all have a fixed boron content, the phase fractions determine the point's boron content exactly. That value must equal $1 - x_1 - x_2$ (the required convention) and not x_2 (the swapped reading); loading stops with an error otherwise. On the eight Fe–Nd–B grids the mean error is 0.0002–0.0074, against 0.33–0.69 for the swapped reading. The check was added after the defect of Section 7.1.2 and reached its present form after two simpler versions failed (Section 7.1.4).

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

5.4 Web advisor

The advisor is a small Flask application with two pages. The *Composition Advisor* takes x_{Nd}, x_{Fe} and x_{B} (normalized automatically, with presets for the $Nd_2Fe_{14}B$ composition and for Nd-rich and Fe-rich variants) and returns three views of the same answer (Figure 4):

- one card per temperature (800–1600 K) listing the stable phases and their fractions, with a status label;
- a chart of phase fractions against temperature, with the sintering range shaded;
- the precomputed isothermal section for a selectable temperature, with the query composition marked on it.

The *Phase Explorer* browses the precomputed sections and phase-fraction maps by temperature. A navigation link connects the advisor to the Stage 2 database application.

The status label is a deliberately simple rule. A temperature is labelled "No FE17ND2" if `FE17ND2` is absent, "Fully solid" if it is present without liquid, "Liquid-heavy" if the liquid fraction exceeds one half, and "Sintering window" otherwise. A summary line above the cards reports the best label within the sintering range. Why the rule tracks $Fe_{17}Nd_2$ rather than the magnetic phase itself is discussed in Section 8; in the next stage, the tracked phase becomes a setting for each alloy system instead of being fixed in the code.

The composition marker on the section is placed by barycentric interpolation between the pixel positions of the triangle's three vertices in the rendered image. Barycentric interpolation is exact for any triangle, so only three calibration points are needed, rather than a reconstruction of the plotting library's axis transforms. The application runs single-threaded, because `pycalphad` calculations are not safe under concurrent requests; for a tool used by one researcher at a time this costs nothing.

5.5 Validation

The module was validated in four steps of increasing specificity. First, binary diagrams computed from single assessments were compared with their published counterparts. Second, the same equilibrium code was run on the two published ternary assessments of Fe–Nd–B, by Hallemans et al. [12] (as distributed in CPDDB [13]) and by Chen et al. [17]. At the $Nd_2Fe_{14}B$ composition both return the ternary compound as the only stable phase from 800 to 1450 K, and γ-Fe plus liquid at 1500 K, consistent with its peritectic decomposition at 1453 K [18]. Third, the same pipeline applied to the description merged from binaries correctly *failed* to produce $Nd_2Fe_{14}B$, which none of the binaries parameterizes. This is the expected result: the missing compound is a limitation of the input, and the code does not invent it.

Fourth, the Fe–Nd–B grids were checked against what a binary-only description must predict. Without Φ, the $Nd_2Fe_{14}B$ composition falls at 800 K into the `FE17ND2+FE17ND5+M2B_C16` field, and the lever rule then fixes the phase fractions at 0.570 $Fe_{17}Nd_2$, 0.176 Fe_2B and 0.254 $Fe_{17}Nd_5$; the equilibrium calculation of Section 6 returns exactly these values. The nearest converged grid point lies 0.0028 away, at $(x_{\mathrm{Fe}}, x_{\mathrm{Nd}}, x_{\mathrm{B}}) = (0.825, 0.120, 0.055)$, where the lever rule gives 0.572, 0.165 and 0.263; the lookup returns exactly these values. The Nd-rich ($x_{\mathrm{Nd}} = 0.15$, $x_{\mathrm{B}} = 0.06$) and Fe-rich (0.10, 0.07) presets of the web advisor lie on grid points in the same field, and the lookup returns the lever-rule fractions exactly: 0.298 / 0.180 / 0.522 and 0.652 / 0.210 / 0.138 ($Fe_{17}Nd_2$ / Fe_2B / $Fe_{17}Nd_5$). Such a lever-rule check, applied to the web advisor's output, is what exposed the coordinate defect of Section 7.1.2. It could not expose the merge defect of

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

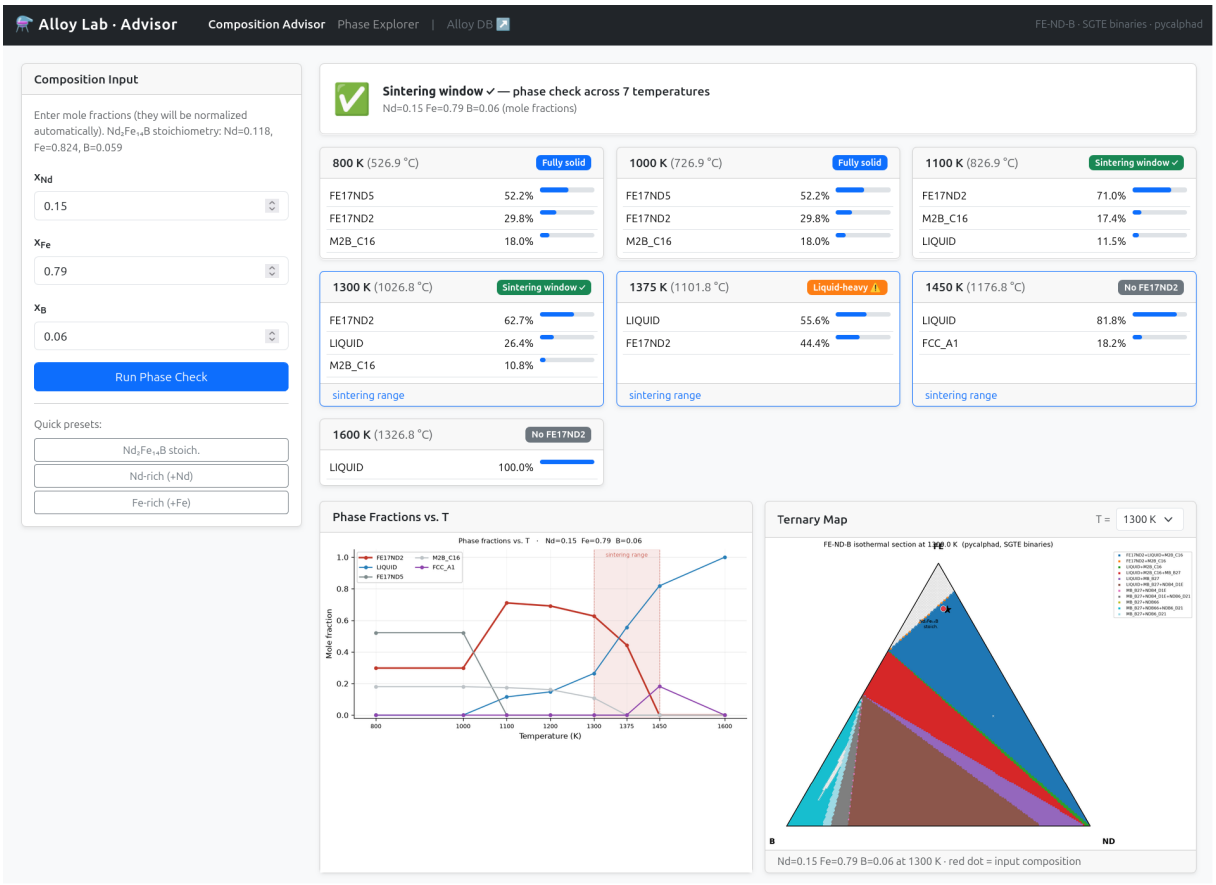


Figure 4: The Composition Advisor for an Nd-rich composition: per-temperature phase cards with status labels, the phase-fraction-versus-temperature chart with the sintering range shaded, and the query composition marked on the precomputed isothermal section.

Section 7.1.1: that defect changed which phases were stable and their energies, but the fractions of the phases found still obeyed the lever rule. It is caught by the edge check of Section 5.1.

6 Feasibility Screen for a Missing Ternary Phase

6.1 Question

A description merged from binaries cannot contain a ternary compound, so at the $Nd_2Fe_{14}B$ composition it predicts a mixture of binary phases. That is correct for what it is, but it raises an obvious question: how stable would a ternary compound have to be to appear anyway? The answer is a threshold formation energy. It turns a qualitative limitation into a number that can be compared with experimental or first-principles formation energies, and it is equally useful for a system where no ternary compound is yet known.

6.2 Method

The target is modelled as a stoichiometric line compound with three sublattices, one element on each. The site ratios are the smallest integers that reproduce the requested composition; for $x = (0.8235, 0.1176, 0.0588)$ in the order (Fe, Nd, B) they are 14:2:1, and the composition is snapped to exactly $(14, 2, 1)/17$. With n_i the site ratios, $n = \sum_i n_i$, and ${}^{\circ}G_i^{\mathrm{SER}}$ the SGTE

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

reference functions, the Gibbs energy of the target per formula unit and per mole of atoms is

$$G^{\mathrm{t}}_{\mathrm{f.u.}} = \sum_i n_i \, {}^{\circ}G^{\mathrm{SER}}_i(T) + n\,\Delta H_f, \qquad G^{\mathrm{t}} = \sum_i x_i \, {}^{\circ}G^{\mathrm{SER}}_i(T) + \Delta H_f, \tag{1}$$

so ΔH_f is a formation energy per mole of atoms relative to the pure-element reference states. The target is appended as text to the merged binary description, which is then loaded once per value of ΔH_f. The reference functions are looked up by name in the merged description, and the screen stops if one is missing, or if the target's energy at $\Delta H_f = 0$ differs from the sum of the reference functions evaluated independently (Section 7.1.7).

For each temperature T and formation energy ΔH_f the screen computes two stability criteria:

1. the phase fraction of the target in a full equilibrium calculation at the target composition, with the target included (stable if above 10^{-3});
2. the driving force

$$\Delta G = G^{\mathrm{t}}(T) - G^{\mathrm{comp}}(T), \tag{2}$$

   where G^{t} is evaluated directly with `pycalphad`'s `calculate` and G^{comp} is the equilibrium Gibbs energy of the binary-only description at the same composition. $\Delta G < 0$ means the target is stable.

G^{comp} does not depend on ΔH_f, so it is computed once per temperature. The two criteria must agree. They are not independent checks of the target's energy, because both use it (Section 7.1.7), but a cell in which they disagree would reveal a numerical problem in the equilibrium step. Because ΔG changes by exactly ΔH_f, the threshold — the ΔH_f at which $\Delta G = 0$ — equals $-\Delta G$ at $\Delta H_f = 0$ and does not depend on the chosen steps of ΔH_f. The scan covers fifteen temperatures from 800 to 1800 K and $\Delta H_f = 0, -5, -10, -15, -20, -30$ kJ mol^{-1} atom^{-1}.

6.3 Verification on a hand-computable case

Before the screen was run on real data, it was run on a toy Fe–Nd–B description in which each element has only its reference phase and Fe_2B is the only compound, with a formation energy of -25 kJ mol^{-1} atom^{-1}. At the 14:2:1 composition the competing assemblage is Fe_2B plus the pure elements, and the lever rule gives its energy relative to the reference states as $(3/17) \times (-25) = -4.41$ kJ mol^{-1} atom^{-1}. The screen must therefore report $\Delta G = +4.4$ kJ mol^{-1} atom^{-1} at $\Delta H_f = 0$, and the target must become stable between $\Delta H_f = 0$ and -5 kJ mol^{-1} atom^{-1}. It did both, at every temperature, and the stability flag agreed with the sign of ΔG in every cell. What this test could not catch is discussed in Section 7.1.7.

The same toy database settled a unit question that would otherwise have silently scaled every result. `pycalphad`'s `calculate` could plausibly report the energy of a 17-site phase per formula unit or per mole of atoms. Comparing it with Equation (1) evaluated by hand showed agreement to all printed digits per mole of atoms, so no further normalization is applied.

6.4 Result for $Nd_2Fe_{14}B$

Table 4 and Figure 5 show the result for the 14:2:1 composition with the merged Fe–Nd–B description of Table 2. In every cell ΔG differs from its value at $\Delta H_f = 0$ by exactly ΔH_f, as Equation (1) requires, and the stability flag agrees with the sign of ΔG. The threshold below which the target is stable is -9.2 kJ mol^{-1} atom^{-1} at 800 K. It becomes less negative up to about 1370 K (-7.0 kJ mol^{-1} atom^{-1}) and more negative above, reaching -10.8 kJ mol^{-1} atom^{-1} at

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Table 4: Feasibility screen for a 14:2:1 (Fe:Nd:B) line compound in the merged Fe–Nd–B description: driving force ΔG (kJ mol^{-1} atom^{-1}) for selected temperatures; **bold** = target stable (phase fraction $> 10^{-3}$). Full scan: 15 temperatures, all cells consistent.

	ΔH_f (kJ mol^{-1} atom^{-1})					
T (K)	0	−5	−10	−15	−20	−30
800	+9.2	+4.2	**−0.8**	**−5.8**	**−10.8**	**−20.8**
1014	+8.1	+3.1	**−1.9**	**−6.9**	**−11.9**	**−21.9**
1300	+7.1	+2.1	**−2.9**	**−7.9**	**−12.9**	**−22.9**
1514	+7.9	+2.9	**−2.1**	**−7.1**	**−12.1**	**−22.1**
1800	+10.8	+5.8	+0.8	**−4.2**	**−9.2**	**−19.2**

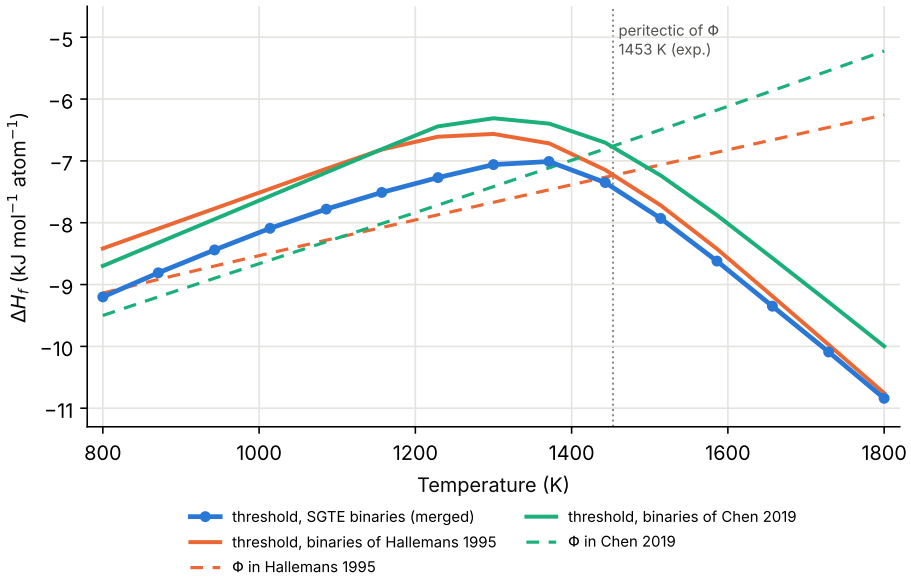


Figure 5: Threshold formation energy below which a 14:2:1 (Fe:Nd:B) line compound is stable against the binary-phase assemblage: for the merged SGTE binaries (full 15-temperature scan) and for the binaries of the full ternary assessments of Hallemans et al. [12] and Chen et al. [17], with the Gibbs energy of Φ in each assessment (dashed). All energies relative to the pure elements. Because the screen's target has no formation entropy, the high-temperature part of the SGTE curve is not a prediction of where $Nd_2Fe_{14}B$ melts (see text).

1800 K. The competing assemblage at 800 K is Fe_2B, $Fe_{17}Nd_2$ and $Fe_{17}Nd_5$; above about 1080 K an Nd-rich liquid replaces $Fe_{17}Nd_5$, and above 1400 K the assemblage is fcc iron and liquid. With the binaries of the two full ternary assessments instead, the threshold is 0.2–0.8 kJ mol^{-1} atom^{-1} less negative between 800 and 1500 K (Section 6.5 and Figure 5).

Two cautions govern how these numbers may be used. First, ΔH_f is a constant: beyond the reference functions, the target has no temperature dependence and no formation entropy. ΔH_f therefore acts as a formation *Gibbs* energy, and the threshold is the formation Gibbs energy the compound would need relative to the reference states. That the threshold becomes more negative above about 1370 K reflects the competing assemblage, which gains entropy, notably once liquid forms, while the target gains none. Second, and for the same reason, the screen reports the target as stable up to 1800 K once ΔH_f is negative enough, whereas real $Nd_2Fe_{14}B$ decomposes peritectically at 1453 K [18]. The high-temperature rows are therefore not a melting prediction. Adding an $a + bT$ term to the target, discussed in Section 9, would remove this limitation.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

6.5 Dependence on the binary descriptions

The threshold is a property of the binary descriptions as much as of the compound. To see by how much, the screen's reference calculation was repeated with two full ternary assessments of Fe–Nd–B: that of Hallemans et al. [12], in the TDB transcription of the NIMS Computational Phase Diagram Database [13], and that of Chen et al. [17]. Both contain Φ and two other ternary borides, and both reproduce the peritectic formation of Φ, $\mathrm{L} + \gamma\text{-Fe} \rightarrow \Phi$, within a few kelvin of the measured 1453 K [18]: 1450–1455 K (Hallemans) and 1452 K (Chen) in our calculations. With their ternary phases removed, the Gibbs energy of the binary-phase assemblage at the 14:2:1 composition gives the threshold for each set of binaries (Figure 5).

The binaries of the two full assessments give thresholds of −8.4 and −8.7 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ at 800 K, and they agree within 0.5 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ at all temperatures up to 1500 K. Their competing assemblage at 800 K is $Fe_{17}Nd_2$ with the borides NdB_4 and Nd_2B_5, whereas the SGTE binaries give Fe_2B, $Fe_{17}Nd_2$ and $Fe_{17}Nd_5$. Despite the different phases, the energies are close: between 800 and 1500 K the SGTE threshold is 0.2–0.8 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ more negative, compared with differences of up to 0.5 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ between the two assessments. The SGTE and Chen binaries are not fully independent: the SGTE Nd–B file (NdB-19Che, Table 2) is taken from the same assessment [17], so the two sets are expected to differ mainly in Fe–Nd and Fe–B. In both assessments, Φ lies below its own binary assemblage by 0.7–1.3 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ between 800 and 1300 K, and the margin vanishes near 1450 K, at the peritectic.

The difference between the sets of binaries is therefore smaller than the margin by which Φ is stable, but not by much. Taken at face value, the Gibbs energy that either assessment gives Φ, combined with the SGTE binaries, makes Φ stable up to about 1380 K (Chen) or 1430 K (Hallemans), 20–70 K below the measured peritectic, with a margin of at most 0.6 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$. At the low-temperature end, the Hallemans Φ falls just short: it lies 0.06 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ above the SGTE threshold at 800 K and becomes stable only above about 830 K, whereas the real compound is stable down to room temperature. The screen with SGTE binaries thus gets the real compound right over most of its range, with little to spare.

A formation energy from experiment or first-principles calculations can therefore be compared with the threshold only like for like, with the same reference states and the same description of the competing phases, and a margin of a few tenths of a $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ is within the uncertainty of the binaries. We do not attempt that comparison here. It is deferred to the next stage, which is planned to take its thermodynamic input from a full ternary assessment.

7 Engineering Validation and Lessons Learned

As in the Stage 1 paper [1], we report the engineering findings that generalize beyond this codebase, in the same Defect / Fix / Generalizable lesson format, grouped by where they arose: thermodynamic data and coordinate handling (Section 7.1), the retrieval and embedding pipeline (Section 7.2), and the boundaries between programs (Section 7.3). Section 7.4 draws the common lessons, and Table 5 summarizes the findings.

7.1 Thermodynamic data and coordinate handling

Seven findings concern the CALPHAD side: how binary descriptions are merged, how grid coordinates are stored and read, and how the feasibility screen builds its target phase. Most of them produced phase assemblages or energies that looked physically reasonable.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Table 5: Summary of the Stage 3 engineering findings.

Finding	Symptom	Found by
Thermodynamic data and coordinate handling		
Text merge (§7.1.1)	Plausible equilibria without iron in the liquid and the solid solutions; threshold 2.3 kJ mol^{-1} atom^{-1} too negative	Comparison with two full assessments; merged description against its source files
Coordinate swap (§7.1.2)	Plausible phase fractions for the wrong composition	Lever rule on a screenshot
Radius tuned under a defect (§7.1.3)	Lookup radius wider than needed (0.025 instead of 0.015)	Review after the swap fix
Self-checks on one point (§7.1.4)	Correct grids rejected at 1200 and 800 K	Loading all real grids
Order-dependent merge (§7.1.5)	Different sections for the same three elements	Running the same system twice
Feasibility screen (§7.1.6)	Uniform tables: all errors, all "not stable", all undefined	Toy case with a hand-computed answer; code review
Undefined reference function (§7.1.7)	None visible; thresholds 0.6–2.6 kJ mol^{-1} atom^{-1} too negative	The same threshold computed by a second route
Retrieval and embedding pipeline		
Embedding truncation (§7.2.1)	None visible; fluent answers	Token count; benchmark with known answers
Identifier collisions (§7.2.2)	None visible; 6.3 % of chunks never stored	Books-without-chunks diagnostic
Size limits (§7.2.3)	A missing book; rejected requests; an SQL error	Full-size runs
Language gap (§7.2.4)	None visible; Russian books unreachable from English queries	Russian-source questions asked in English
Near miss (§7.2.5)	Confident answer about a different system	Out-of-corpus questions, graded by hand
Vector store in git (§7.2.6)	Push rejected; files in history; store deleted by the fix	Real push; missing store
Integration and process boundaries		
JSON on stdout (§7.3.1)	Parse error in the calling code	End-to-end call
Server-side truncation (§7.3.2)	Answers that look like reasoning errors; long chunks embedded from their beginning	Tokens read with and without a large window; stored against fresh vectors
Deployed ≠ evaluated (§7.3.3)	None visible; advisor silently skipped	Describing the interface for this paper
Shifted chart (§7.3.4)	Plausible curves one temperature ahead of the data	Chart read against the cards beside it

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

7.1.1 A merge that removed iron from its own phases

Defect. The first version of the merge combined the binary files as text. It kept the first `FUNCTION`, `ELEMENT`, `PHASE` and `CONSTITUENT` block of each name and passed every `PARAMETER` line through. Both rules failed for Fe–Nd–B. The SGTE files abbreviate the keyword to `CONST`, as the TDB format allows, and the text merge recognized only the full word. Every constituent list was therefore kept, and the one read last, from the Nd–B file, replaced the others. The liquid and the bcc and fcc solid solutions therefore contained only Nd and B: iron could not dissolve in them at all. Parameters that two files share, such as the lattice stabilities of pure Nd and of liquid B, appeared twice, and `pycalphad` adds parameters with the same key, so these phases received twice the Gibbs energy of the pure element.

Every result computed from the merged description looked plausible: real phases, fractions that sum to one, and fractions that agree with the lever rule (Section 5.5). The lever rule could not reveal the defect, because it checks the amounts of the phases found, not whether they are the right phases with the right energies. The defect showed up as an outlier. The feasibility threshold at 800 K came out at -12.0 kJ mol^{-1} atom^{-1}, against -8.4 and -8.7 kJ mol^{-1} atom^{-1} with the binaries of two full ternary assessments, which agree with each other. Our first explanation, a compound counted twice, was wrong. A direct comparison of the merged description with its three source files found the actual defects: along the edges of the triangle, the merged description differed from the binaries by up to 155 kJ mol^{-1}.

Fix. The parsed databases are merged instead of their text, with constituents united, each parameter kept once and conflicts reported, and the result must reproduce each binary along its edge before any grid is computed (Section 5.1). The feasibility screen and the grids of this paper were recomputed with the corrected merge; the threshold at 800 K is now -9.2 kJ mol^{-1} atom^{-1}, after a second correction (Section 7.1.7). With the corrected merge, the sections changed markedly. At 800 K the $Nd_2Fe_{14}B$ composition now lies in a field of $Fe_{17}Nd_2$, Fe_2B and $Fe_{17}Nd_5$ instead of $Fe_{17}Nd_2$, Fe_2B and Nd, and the number of points that fail to converge fell from 719 to 54.

Generalizable lesson. A model assembled from validated parts must be checked to reproduce each part in that part's own domain. Here that check was cheap, had an exact expected answer — the binary file itself — and was possible from the start; it was not run because each part was trusted. A consistency check inside a result, such as the lever rule, tests the bookkeeping of a calculation, not its input. A format with abbreviations and optional syntax should be read by its established parser, not by string matching. And a result that disagrees with two independent references calls for a direct check, not for an explanation.

7.1.2 A coordinate convention fixed in the wrong direction

Defect. The grid generator stores, for elements given as (Fe, Nd, B), the coordinates $(x_{\mathrm{Fe}}, x_{\mathrm{Nd}})$ of each point (Section 5.2). The lookup was first written for $(x_{\mathrm{Nd}}, x_{\mathrm{B}})$, which a test showed to be wrong. The correction changed it to $(x_{\mathrm{Fe}}, x_{\mathrm{B}})$ — right in the first coordinate, wrong in the second. It was justified by a qualitative check: the points with the smallest $x_1 + x_2$ contained Nd–B phases, which was read as "the Nd-rich corner". But those phases (`NDB4_D1E`, `NDB6_D21`, `NDB66`) are borides from the B-rich corner, so the check was equally consistent with the true convention and could not tell the two apart. From then on, every grid-backed result in the web advisor described the composition with Nd and B swapped.

The output looked entirely plausible: real phases, fractions summing to one, smooth variation with temperature. The defect was found while preparing this paper, by checking advisor screenshots against the lever rule. For an input with $x_{\mathrm{Nd}} = 0.15$ and $x_{\mathrm{B}} = 0.06$ the advisor reported 45.0 % Fe_2B. That is exactly 3×0.15, the amount of Fe_2B that $x_{\mathrm{B}} = 0.15$ would produce, whereas

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

$x_B = 0.06$ allows at most 18 %. A second screenshot ($x_{Nd} = 0.10$, 30.0 % Fe_2B) showed the same signature. The live-calculation fallback, the composition marker and the feasibility screen use their own, correct conventions and were not affected.

Fix. The lookup now queries (x_{Fe}, x_{Nd}), taken from the one line of the generator that sets the equilibrium conditions rather than inferred from the data, and the grids are checked at load time (Section 5.3). The fix was tested on a small grid produced by the generator's own code from a toy database: the old lookup reproduces the symptom exactly (45 % Fe_2B for $x_{Nd} = 0.15$), and the corrected lookup returns the assemblage of the requested composition. On the real grids, the corrected lookup reproduces the lever rule for all three test compositions (Section 5.5).

Generalizable lesson. A data file's layout is defined by the code that writes it, and the place to read a convention is that code, not the data. A sanity check, moreover, is only worth something if it can fail: "these phases look Nd-related" was compatible with both conventions, whereas a lever-rule check — a number that must come out right — was not.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

7.1.3 A parameter tuned while another defect was present

Defect. The lookup radius was widened from 0.015 to 0.025 because the query for the $Nd_2Fe_{14}B$ composition found no converged point within 0.015; the nearest was at 0.017. That measurement was made while the coordinate swap of Section 7.1.2 was present, so it described the neighbourhood of a different composition. The wider radius may not be needed, and it has a cost: near a phase boundary, the nearest converged point can lie in the neighbouring field.

Fix. The radius was re-measured on the corrected lookup, whose self-test now prints the distance to the nearest point for each test composition. The nearest converged point to the $Nd_2Fe_{14}B$ composition lies at 0.0028, and the two presets fall on grid points. The radius was set back to 0.015, three grid spacings.

Generalizable lesson. When a defect is fixed, every parameter that was tuned while it was present has been tuned against the defect. Listing those parameters and re-tuning them belongs to the fix, not to later maintenance.

7.1.4 Self-checks that failed on correct data

Defect. The load-time check of the coordinate convention went through two versions that failed on correct grids. The first required the converged point with the largest x_{Nd} to contain `DHCP` or liquid. On the real grids it stopped the loading at 1200 K, where the Nd-rich phase is correctly `BCC_A2`: neodymium is body-centred cubic between 1136 and 1297 K. The second version required pure boron at the boron corner instead, which is solid over the whole temperature range. It failed at 800 K, because the points closest to that corner did not converge, and the converged point nearest to it contained no pure boron. Each version had passed on synthetic test data.

Fix. The check now uses the boron mass balance of Section 5.3. It uses every point with a fixed boron content instead of one extreme point, and it compares the two possible conventions directly, so the answer is a pair of numbers with a wide gap between them rather than a yes or no.

Generalizable lesson. A check that tests one extreme point of the data inherits the physics of that point. A check should rest on a law that holds everywhere — here, conservation of mass — and it must be run on the real data before it is trusted, because synthetic data lack exactly the features that make real data awkward.

7.1.5 Results that depended on the order of the input

Defect. In an early version of the generic generator, the computed section depended on the order in which the three elements were typed. The cause was the merge step: duplicate blocks were resolved first-come-first-served, and "first" followed the dictionary insertion order of the binary files, which followed the order of the elements. Separately, the marker for a reference composition on the plotted section was tied to a fixed pair of vertices, so it was drawn in the wrong place whenever the elements were given in a different order.

Fix. The binary files are sorted by name before merging, and each element's vertex on the plot is looked up rather than assumed.

Generalizable lesson. Any "first one wins" rule turns input order into part of the program's meaning. Either the order must be made irrelevant, by sorting into a canonical order, or it must be an explicit, documented input.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

7.1.6 A screen that agreed with itself for the wrong reason

Defect. The first version of the feasibility screen failed in five successive ways, each hiding the next. (i) The function meant to add the target phase to a loaded database could not work, because a loaded `pycalphad` database cannot be extended; its file-writing step had been replaced by a placeholder guarded by `if False:`, beside a comment pointing to the approach that did work. (ii) The target phase was declared with the type string `:FENDB:`, which `pycalphad` reads as five model-type codes, F, E, N, D and B, so building the phase model failed. Every cell of the result table showed only "ERR", because the exception was caught and discarded. (iii) With that fixed, every cell reported "not stable". The target had been hard-coded as a 1:1:1 compound, which cannot be the stable phase at a 14:2:1 composition within the scanned energy range. (iv) Every driving force was undefined, because it was computed from a single-phase equilibrium of the target at a composition the line compound cannot reach. (v) An independent code review noted that the original definition, equilibrium energy minus target energy, can never be positive, since the equilibrium is the minimum, so half of the printed legend could never apply.

Fix. The redesign of Section 6: site ratios derived from the composition, the default model type, the target energy from `calculate`, a binary-only reference assemblage, two independent stability criteria that must agree, and the first traceback printed in full instead of being discarded. The redesign was verified on the hand-computable toy case before it was run on real data.

Generalizable lesson. A uniform result table — all errors, all "not stable", all undefined — is a finding about the code, not about the material. And a screen must be able to represent the answer it is asked about, and must be shown to reproduce a hand-computable answer, before its output on real data means anything.

7.1.7 A reference energy that was silently zero

Defect. The target phase is written as TDB text whose Gibbs energy refers to the pure-element reference functions by name. The screen built each name as `GHSER` followed by the element symbol, which gives `GHSERB` for boron, whereas the SGTE files call the boron function `GHSERBB`, doubling the one-letter symbol. `pycalphad` sets an undefined function to zero without a warning, so the target lacked its boron term, $x_{\mathrm{B}}\,{}^{\circ}G_{\mathrm{B}}^{\mathrm{SER}}$, and every threshold came out too negative, by $0.55\ \mathrm{kJ\,mol^{-1}\,atom^{-1}}$ at 800 K and $2.6\ \mathrm{kJ\,mol^{-1}\,atom^{-1}}$ at 1800 K. The two stability criteria of Section 6 still agreed in every cell, because both used the same target. The hand-computable toy case of Section 6.3 could not catch this. It passed at every temperature, and with the boron term

missing it would have been off by the same 0.55–2.6 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$ unless its boron function was either named as the code expected or zero. In both cases the test shared the defect: a toy database written by the same hands as the code repeats the code's assumptions.

The defect was found when a separate script computed the same threshold by a different route, from the equilibrium energy and the reference functions evaluated directly. The two disagreed by an amount that grew with temperature and matched $x_\mathrm{B}\,{}^\circ G_\mathrm{B}^\mathrm{SER}$ at all five temperatures compared. The separate script had a defect of its own: evaluated directly, the reference functions are zero above their last temperature range, where `pycalphad` extrapolates them, so at 1800 K it disagreed for a different reason.

Fix. The screen looks up each element's reference function in the merged description and stops if none exists. Before the scan, it checks the target's energy at $\Delta H_f = 0$ against the sum of the reference functions evaluated independently, at 800 K, inside their range. All results of Section 6 were recomputed.

Generalizable lesson. A library that replaces an undefined name by zero turns a naming slip into a plausible number. Two criteria that share an input are not independent checks of that input. And an independent check must evaluate the same quantity by a different route, under conditions where both routes are valid; otherwise it trades one silent error for another.

7.2 Retrieval and embedding pipeline

Six findings concern the literature index: what the embedding model reads, how chunks are identified and stored, which passages a query can reach, and where the index itself is kept. Several of them cost retrieval quality without any visible symptom.

7.2.1 An index that never read most of its text

Defect. The original index used the vector store's default embedding model, which reads the first 256 tokens of a text and ignores the rest without a warning. The chunks were sized in words, 400 of them, which is a median of 592–631 tokens. On average 59–62 % of each chunk's text was therefore never embedded and could not be found, although it was stored and would have been passed to the language model had its chunk been retrieved. Russian text, in a model trained on English, was largely invisible as well. The advisor still produced fluent answers, so nothing looked wrong.

Fix. A diagnostic counted the tokens of a sample of chunks with the model's own tokenizer; the benchmark of Section 4.4 then measured the consequence (hit@5 = 0.13). The chunks were re-embedded with a multilingual model whose context holds a whole chunk, which raised hit@5 to 0.50 (0.10 and 0.51 on the rebuilt index).

Generalizable lesson. A default component has limits that the pipeline around it must respect, and a chunk size chosen in words says nothing about a limit counted in tokens. The input limit of an embedding model belongs in the same table as the chunk-size distribution, and a retrieval system needs questions with known answers before its answers can be judged.

7.2.2 Identifiers that collided silently

Defect. The ingestion script built each chunk's identifier from the first 40 characters of the file name and the chunk number. Books whose file names begin alike — volumes of a series, parts of a handbook — received the same identifiers, and the vector store does not raise an error when

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

a chunk with an existing identifier is added: it skips it. The second book therefore lost every chunk whose number the first book had already used, which for a shorter book is all of them. In total 4 011 of 64 115 chunks (6.3 %) were never stored, among them a complete volume of the Landolt–Börnstein series. The defect was present from the first index in September 2026.

Fix. A diagnostic that lists, for each folder, the books with no chunks in the collection found it; one benchmark question whose answer lies in the lost volume confirmed it. The index was rebuilt in October 2026 with identifiers that contain a hash of the complete file name, and the rebuild compares, for each collection, the chunks produced with the chunks stored in both embedding sets: all 72 530 were stored, and the lost volume answers its benchmark question again (Section 4.4).

Generalizable lesson. An identifier must be unique by construction, not by the coincidence that names differ early. Any store that skips duplicates needs a count after each insert: chunks produced against chunks stored.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

7.2.3 Limits that fail far from their cause

Defect. Three size limits surfaced while rebuilding and evaluating the index, none of them with a message that named the size. The PDF library stops parsing a content stream after a fixed number of steps, a protection against malicious files, and the largest book of the corpus, a 2 787-page encyclopedia, exceeded it, so its 4 935 chunks were missing. The model server uses a context window smaller than the embedding model's own unless a larger one is requested, and rejected batches of long chunks. And reading the 36 197-chunk magnetic collection in a single call failed with "too many SQL variables", a limit of the database engine under the vector store; the message suggested a problem with the data, and the Russian characters were the first suspect.

Fix. The parsing limit was raised for the laboratory's own, trusted books; the full context window is requested explicitly, with a per-chunk fallback that reports any chunk it has to shorten; and large collections are read in pages of 5 000.

Generalizable lesson. Each layer of a pipeline has limits of its own, and they show up only at full size. Every stage should be run once on the largest real input, not a sample, and its output counted against its input.

7.2.4 A multilingual index that served one language

Defect. The index embeds English and Russian books with a multilingual model, and the benchmark found the passages of Russian-source questions well: five of seven. But those questions were asked in Russian, whereas the advisor asks only in English. Asked in English, the same seven questions found none of their passages, because passages in the language of the question outrank the right passage in the other language. The Russian books, a tenth of the corpus, were therefore practically unreachable from the application. Nothing showed it: the answers were fluent and drawn from English books.

Fix. The gap was found when the author asked whether a question could be put in English about a Russian book, and the seven questions were rerun in English. A first fix translated each query with the language model; reading all 44 translations showed Chinese characters in five of them and wrong technical terms in others, so it was dropped. The advisor now tags every chunk with its language and searches the Russian chunks with a Russian version of its own query template, built without a model (Section 4.2). With this fifth passage, four of the seven questions asked in English were answered from their Russian source, and a run of all 44 questions showed no loss elsewhere (Section 4.4).

Generalizable lesson. A benchmark must ask the way the application asks — down to the language. A multilingual model can represent several languages in one space, but which passage wins is still decided by distance, and distance favours the language of the question. And a component added as a fix — here a translator — needs the same scrutiny as the defect it fixes: its output must be read, not assumed.

7.2.5 A near miss answered with confidence

Defect. One out-of-corpus question asked for the stable phases of Fe–Bi, a system the indexed handbook does not cover. With bge-m3, the nearest passages were about Fe–B, from a book on permanent magnets, and the model answered about Fe–B — fluently, with temperatures taken correctly from the passage — as if that were the question. The answer is faithful to its passages by the grading criterion of Section 4.4, and wrong. The weaker original index found nothing similar, and the model declined.

Fix. None yet. Two guards are candidates: an instruction in the prompt to check that the passages concern the system asked about, and an automatic comparison of the chemical elements named in the question and in the answer.

Generalizable lesson. A better retriever also makes near misses more convincing. Faithfulness to the retrieved text is necessary but not sufficient; an answer must also be checked against the question.

7.2.6 A vector store in version control

Defect. To start tracking the Stage 3 code, the line in `.gitignore` that excluded the whole `stage_three/` directory was removed. That also exposed the RAG vector store inside it. The next commit included a 1.25 GB SQLite file and index files of 57–92 MB, and the hosting service rejected the push: its limit is 100 MB per file. Removing the store from the index and adding it back to `.gitignore` did not help, because the files were already part of an earlier commit.

Fix. The store was removed from the entire history with `git filter-repo`, which rewrote all 46 commits. The tool also removes the remote configuration as a safety measure, so the remote had to be re-added before a force push. The store is now excluded permanently.

The fix had a cost that was noticed only later. The store was still tracked in the commit that was checked out, and the rewrite brings the working copy in line with the rewritten history, in which the store no longer exists; it was therefore deleted from disk as well. In October the store was rebuilt from the PDFs with the original ingestion pipeline, and the rebuild is the index evaluated in Section 4.4.

Generalizable lesson. The Stage 1 paper reported the opposite error, a directory ignored only after it had been committed (Stage 1, §5.4). Here, a directory was un-ignored more broadly than intended. Both point to the same practice: a change to `.gitignore` has the reach of the whole directory it touches, so the staged file list and file sizes need checking before the commit. Generated artifacts such as vector stores and cached grids do not belong in version control. The grid caches, at 50–100 MB each and regenerable in minutes, were accepted with size warnings and are candidates for removal too. And a history rewrite acts on the working copy as well as on the history: anything it removes from the history should be backed up first.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

7.3 Integration and process boundaries

Four findings arise where one program hands data to another: between the advisor and its subprocess, between the advisor and the model server, between the evaluated code and the code the application runs, and between the calculation and the chart that displays it.

7.3.1 Human-readable output on a machine-readable channel

Defect. The RAG subprocess printed status messages to standard output before its JSON result. The calling application parsed standard output as JSON, so every call failed with a parse error that pointed at the calling code rather than at the subprocess.

Fix. The entry point was restructured so that in JSON mode it writes only the JSON object to standard output. Human-readable output is reserved for a separate interactive test mode.

Generalizable lesson. When a subprocess's standard output is the interface, everything printed there is data. Diagnostics belong on standard error or in a separate mode, never interleaved with the payload.

7.3.2 Silent truncation in the model server

Defect. The model server truncated inputs twice without an error. The first case concerns the language model, which was called without setting its context window, so the server's default of 4 096 tokens applied. A prompt of the advisor — instruction, four passages and question — is mostly 2 100 to 4 000 tokens long, but Russian text needs more tokens per word, and the ten longest of the 88 benchmark prompts, all for Russian-language questions, were 4 128 to 4 814 tokens long. The server kept only about 2 050 tokens of each and raised no error, so about half of every one of these prompts never reached the model. The resulting answers looked like failures of the model: they included both cases in which the gold passage was ranked first and the answer was nevertheless wrong, and seven of the eight answers with claims not in their passages. The second case concerns the re-index with bge-m3. It sent 32 chunks per request and allowed the server to shorten inputs that did not fit, assuming that inputs too long for the model would be rejected and then handled one by one. In fact, about 2 048 tokens of each chunk were read, although the model and the requested window allow 8 192, and nothing was rejected: the script reported that no chunk had been shortened. A check on 23 October compared the stored vectors of the longest chunks with fresh embeddings of the same text: for every chunk longer than about 2 000 tokens the stored vector matched an embedding of the chunk's beginning (cosine up to 0.99996 at 2 048 tokens), while all shorter chunks matched exactly. The rebuilt index contains 50 such chunks among 72 530 (Section 4); whether a gold chunk of the benchmark was among those cut in the earlier index was not checked, and the rebuilt index embeds all of them in full.

Fix. The suspicion came from estimating prompt lengths while the advisor was rewritten (Section 7.3.3). A short script then sent every benchmark prompt twice, with the default window and with a large one, and compared the number of prompt tokens the model reported having read. The window is now set explicitly to 16 384 tokens, widened automatically if a prompt might not fit, and the tokens read are returned with every answer. With full prompts, every answer whose context held the gold passage was correct (Section 4.4). For the embeddings, the rebuild forbids shortening, so that a chunk that does not fit raises an error; it embeds chunks estimated to be long one at a time and records their token counts, and it ends by embedding the longest chunks again and comparing the result with the stored vectors.

Generalizable lesson. This is the lesson of Section 7.2.1, now in the model server that replaced

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

the truncating embedding model: every component with an input limit must have that limit set explicitly, and its input must be measured against it. A model that does not see its whole input fails in ways that look like errors of reasoning.

7.3.3 The deployed advisor was not the evaluated one

Defect. The benchmark copied its prompt and settings from the advisor's interactive query script. The module that the desktop application actually calls had been written a month earlier and had drifted from it. It queried a small test collection that the October rebuild of the store had not recreated; on its next call it would have silently created an empty one and answered that no literature was indexed. It was started with a Python environment that had since been moved out of the repository, so the application had in fact skipped the advisor ever since, showing only a short "RAG skipped" note in its result panel. And it used a different prompt, one that invited general principles beyond the passages, no fixed temperature, and "sources" that were the first 200 characters of each passage rather than the book.

Fix. The module was rewritten (Section 4.2): it opens the re-embedded collection and fails with a clear message if a collection is missing, uses the benchmarked prompt and settings, and returns the book names. The defect was found while describing the module's interface for this paper. The benchmark now imports retrieval, prompt, settings and generation from this module, so that the evaluation and the application share one code path, and all answers were generated again.

Generalizable lesson. An evaluation is evidence only about the code it ran. If the evaluated and the deployed paths are separate, they drift apart without any error; they must either be one code path or be tested end to end, from the application, after every change to either.

7.3.4 A chart one temperature ahead of its data

Defect. The advisor's chart of phase fractions against temperature (Figure 4) collected its curves in a loop that, for a phase already present at the previous temperature, wrote the new value into that previous temperature's slot and padded the current one with zero. Every curve was therefore drawn one temperature too early, and each phase dropped to zero at the last temperature where it existed: for the Nd-rich preset, liquid appeared at 1.0 at 1450 K and at 0 at 1600 K. A phase present at a single temperature was drawn correctly. The curves were smooth and plausible.

Fix. The defect was found while retaking the screenshot of Figure 4 for this paper, by reading the chart against the temperature cards beside it, which are filled from the same calculation. The chart now collects one value per phase and temperature and builds each curve over all temperatures, with zero where a phase is absent; it was tested against the card values before the screenshot was retaken.

Generalizable lesson. Two views of one result are a free check if they are compared. A display layer can corrupt correct numbers on its own, and indexing by "the last element so far" silently couples each value to the history of the loop.

7.4 Common lessons

Every defect above produced output that looked plausible or failed far from its cause, and none was found by reading the code that contained it. Each was found by a quantitative or end-to-end check — a lever rule applied to a screenshot, a merged description compared with its own source files, a hand-computable reference case, a real push to a real server, a token count, a benchmark with known answers. This is the same conclusion as in Stage 1, now with a sharper form: a

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

check finds defects only if its expected result is a number that could have come out wrong. The RAG findings add a second point. The defects that cost the most retrieval quality, the truncated embeddings, the colliding identifiers and the language gap, produced no symptom at all: the advisor kept answering fluently. The cut prompts made the language model look worse than it was, and the cut chunks of the re-index went unnoticed until the stored vectors were compared with fresh ones. Only measurements made for the purpose — a token count, a list of books without chunks, a benchmark with known answers, the number of tokens a model reports having read — could show them, and they belong before an advisor is put to use, not after.

8 Discussion and Limitations

What this paper deliberately does not report. As in the earlier stages, we report no compositions, measurements or advisor outputs for the laboratory's samples. Fe–Nd–B appears only as a public validation case, and the compositions used as examples are representative rather than drawn from unpublished work.

Binary-only descriptions. The central limitation of the CALPHAD module is its thermodynamic input. A description merged from binaries has no ternary interaction parameters and no ternary compounds, so its ternary phase relations are an extrapolation. For Fe–Nd–B this means that the magnetic phase itself is absent. The feasibility screen turns this limitation into a number but does not remove it. Where a dedicated ternary assessment exists, such as for Fe–Nd–B [12, 14, 17], it should be used instead; the computation and serving layers do not depend on how the description was assembled.

Merge semantics. The merge keeps the first definition of each function and parameter and reports any file that differs; for Fe–Nd–B it reports two conflicts (Section 5.1). Such conflicts are inherent in combining independently assessed binaries, which need not share unary data or phase models, and the edge check shows whether they change any binary. It cannot show whether a ternary result is right, for the reason given above.

What the status label tracks. The advisor's status rule keys on `FE17ND2`, because in the binary-merged description the region around the $Nd_2Fe_{14}B$ composition is occupied by $Fe_{17}Nd_2$-bearing assemblages. The label is therefore a heuristic for "Fe-rich intermetallic with limited liquid", not a statement about the magnetic phase; in real magnets $Fe_{17}Nd_2$ is an unwanted phase. The rule is also specific to this one system: for another alloy family the tracked phase has to be chosen anew, and as long as it is written into the code, every new system means a code change. The next step is therefore not a better rule for Fe–Nd–B but a setting per system that names the tracked phase, so that a new system needs a choice and no new code.

Lookup resolution and coverage. Results come from the nearest converged point of a grid with spacing 0.005. Near a phase boundary this can be the neighbouring field, and compositions within the lookup radius of a boundary should be read with that in mind. A single grid point can also hold a local rather than the global minimum, as one point of Figure 3 does. Only eight temperatures are cached; any other temperature requires a live calculation.

Feasibility model. The target is a line compound with fixed site ratios and no formation entropy, so the threshold is a formation Gibbs energy relative to the binary-phase assemblage of these particular assessments. It inherits every limitation of those assessments and of the merge, and Section 6.5 shows that the choice of binaries moves it by 0.2–0.8 $\mathrm{kJ\,mol^{-1}\,atom^{-1}}$, not much less than the margin by which Φ is stable in a full assessment.

RAG advisor. The benchmark is small: 39 scored questions, written and graded by one person, so a difference of a few questions is within the uncertainty (Table 1 gives the intervals);

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

only the large differences between the two embedding models — in retrieval and in wrong or missing answers — are firmly established. The retrieval metric is strict and understates how often useful passages are found (Section 4.4). The holes of the first index — eight unindexed books and colliding identifiers — are closed (Section 4), but the advisor searches only one of the three collections, and it reaches the Russian books only through one passage found with a fixed Russian template; a free-text query in one language would still miss most sources in the other. Because answers are regenerated rather than stored, and change with any setting of the model (Section 4.4), an answer shown to the researcher cannot later be reproduced exactly. Formulas need particular caution. Text extraction, and optical character recognition even more, loses sub- and superscripts and confuses symbols, so an answer can be faithful to its passage and still state a wrong formula: in one answer the model reproduced a Laplace equation whose extracted text divides the surface tension by the curvature instead of multiplying, and in another it wrote a formula for the domain size that appears in none of its passages. Formulas in an answer should be checked against the book, not against the extracted text. The corpus consists of copyrighted books, so the index and the benchmark's retrieved passages cannot be published with the code; others can reproduce the system and the procedure, but not its answers.

Reproducibility and testing. A code review in October 2026 found absolute paths to the author's machine in several of the Stage 3 phase-diagram scripts. As in Stage 1, there is no automated test suite. The toy-database checks of Sections 6.3 and 7.1.2, the edge check of the merge and the lever-rule checks of Section 5.5 are the natural first entries of one.

Single lab, snapshot. The system was built by and for one researcher's laboratory and has not been evaluated elsewhere. Like the Stage 1 and Stage 2 papers, this one describes a dated snapshot of a living codebase.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

9 Conclusion and Future Work

Stage 3 added the two pre-synthesis advisory components promised in the Stage 1 paper. A retrieval-augmented advisor grounds a locally served language model in the laboratory's book collection without data leaving local hardware. Measured on a benchmark of the author's own questions, replacing the default embedding model with a multilingual long-context one raised the share of questions whose source passage is among the top five from 10 to 51 % and reduced the wrong or missing answers from 19 to 7. A CALPHAD module turns the open SGTE binary collection into precomputed ternary sections served interactively, and adds a feasibility screen that expresses the absence of ternary phases as a threshold formation energy — about -7.0 to -9.2 kJ mol^{-1} atom^{-1} between 800 and 1500 K for the $Nd_2Fe_{14}B$ composition with the SGTE binaries, 0.2–0.8 kJ mol^{-1} atom^{-1} more negative than with the binaries of two full ternary assessments. The most useful lessons are that a sanity check must be quantitative to be worth anything — a qualitative one let a coordinate swap survive, and a lever rule found it — and that a description assembled from parts must be checked against each part: a merge that removed iron from the liquid passed every other check. Their counterpart for the RAG advisor is that fluent answers are no evidence of a working index, and that a benchmark must ask the way the application asks, down to the language.

Six directions follow directly. First, the benchmark will be repeated with queries built the way the advisor builds them; the near miss of Section 7.2.5 calls for a guard on the generation side, and the language of the answers should be fixed in the prompt. Second, the RAG advisor will be added to the web interface, with automatic routing of each query to the right collection, and retrieved sources will be linked to the `literature_sources` table, as anticipated in the Stage 1 paper. At the same time, the vector store, the cached grids and the raw measurement data read by the Stage 2 evaluation apps will move to the always-on server, so that the advisors and the

apps can be used from any computer in the laboratory without copying the corpus, the grids or the data to it; only the language model stays on the GPU machine, which the server reaches over the local network. Third, the web advisor will be generalized beyond Fe–Nd–B: a choice of elements, grids computed on demand as background jobs, colours for phases the interface has not seen before, and a configurable tracked phase. Fourth, the feasibility screen will be given an $a + bT$ term for the target, so that it can also describe decomposition at high temperature. Fifth, ESPEI [7] will be used to fit ternary assessments to the laboratory's own experimental data, once enough data exist for a system, which would remove the binary-only limitation for the systems the laboratory actually works on. Sixth, a property-prediction model, deferred at the start of Stage 3, becomes possible once the advisors have helped produce enough data to train one.

Together, the three stages show that a practising researcher can build durable research infrastructure incrementally, validating and reporting each stage before the next begins.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

Code availability

The source code is available at `https://github.com/iliya-radulov/my_db_project` under the MIT License. The version described in this paper is archived on Zenodo as release `v3.0-stage3`; the DOI `https://doi.org/10.5281/zenodo.23155920` refers to all archived versions and resolves to the latest. The book corpus, the vector store and the passages retrieved for the benchmark are not included, because the books are copyrighted (Section 8).

References

[1] Radulov, I. (2026). An open-source data infrastructure for alloy synthesis research: Stage 1 — Composition screening and literature cross-referencing. Zenodo. `https://doi.org/10.5281/zenodo.23158894`

[2] Radulov, I. (2026). An open-source data infrastructure for alloy synthesis research: Stage 2 — Validated multi-instrument characterization pipelines. Zenodo. `https://doi.org/10.5281/zenodo.23499416`

[3] Lukas, H. L., Fries, S. G., & Sundman, B. (2007). *Computational Thermodynamics: The Calphad Method*. Cambridge University Press.

[4] Saunders, N., & Miodownik, A. P. (1998). *CALPHAD (Calculation of Phase Diagrams): A Comprehensive Guide*. Pergamon.

[5] Dinsdale, A. T. (1991). SGTE data for pure elements. *Calphad*, 15(4), 317–425.

[6] Otis, R., & Liu, Z.-K. (2017). pycalphad: CALPHAD-based computational thermodynamics in Python. *Journal of Open Research Software*, 5(1), 1. doi:10.5334/jors.140

[7] Bocklund, B., Otis, R., Egorov, A., Obaied, A., Roslyakova, I., & Liu, Z.-K. (2019). ESPEI for efficient thermodynamic database development, modification, and uncertainty quantification: application to Cu–Mg. *MRS Communications*, 9(2), 618–627.

[8] van de Walle, A., Nataraj, C., & Liu, Z.-K. (2018). The Thermodynamic Database Database. *Calphad*, 61, 173.

[9] Hallstedt, B. (2025). The SGTE collection of binary datasets. *Calphad*, 89, 102833. doi:10.1016/j.calphad.2025.102833

[10] Sagawa, M., Fujimura, S., Togawa, N., Yamamoto, H., & Matsuura, Y. (1984). New material for permanent magnets on a base of Nd and Fe. *Journal of Applied Physics*, 55(6), 2083–2087.

[11] Herbst, J. F. (1991). $R_2Fe_{14}B$ materials: Intrinsic properties and technological aspects. *Reviews of Modern Physics*, 63(4), 819–898.

[12] Hallemans, B., Wollants, P., & Roos, J. R. (1995). Thermodynamic assessment of the Fe–Nd–B phase diagram. *Journal of Phase Equilibria*, 16(2), 137–149. doi:10.1007/BF02664851

[13] Abe, T., Hashimoto, K., Goto, Y., Sawada, Y., & Hirose, K. (2007). CPDDB: Computational Phase Diagram Database [Data set]. National Institute for Materials Science. doi:10.48505/nims.3060. Fe–Nd–B dataset `bfend_hal.tdb`, after Hallemans et al. (1995).

[14] Van Ende, M.-A., & Jung, I.-H. (2013). Critical thermodynamic evaluation and optimization of the Fe–B, Fe–Nd, B–Nd and Nd–Fe–B systems. *Journal of Alloys and Compounds*, 548, 133–154.

[15] Chen, T. L., Wang, J., Rong, M. H., Rao, G. H., & Zhou, H. Y. (2016). Experimental investigation and thermodynamic assessment of the Fe–Pr and Fe–Nd binary systems. *Calphad*, 55, 270–280. doi:10.1016/j.calphad.2016.10.004

[16] Takahashi, K., Ishikawa, K., Fujioka, M., & Ohtani, H. (2020). Thermodynamic analysis of the Fe–Mo–B ternary system. *Tetsu-to-Hagané*, 106(6), 310–320.

[17] Chen, T. L., Wang, J., Guo, C. P., Li, C. R., Du, Z. M., Rao, G. H., & Zhou, H. Y. (2019). Thermodynamic description of the Nd–Fe–B ternary system. *Calphad*, 66, 101627. doi:10.1016/j.calphad.2019.101627

[18] Schneider, G., Henig, E.-Th., Petzow, G., & Stadelmaier, H. H. (1986). Phase relations in the system Fe–Nd–B. *Zeitschrift für Metallkunde*, 77(11), 755–761.

[19] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.

[20] Jablonka, K. M., Ai, Q., Al-Feghali, A., Badhwar, S., Bocarsly, J. D., et al. (2023). 14 examples of how LLMs can transform materials science and chemistry: a reflection on a large language model hackathon. *Digital Discovery*, 2(5), 1233–1250. doi:10.1039/D3DD00113J

[21] Lála, J., O'Donoghue, O., Shtedritski, A., Cox, S., Rodriques, S. G., & White, A. D. (2023). PaperQA: Retrieval-augmented generative agent for scientific research. arXiv:2312.07559.

[22] Skarlinski, M. D., Cox, S., Laurent, J. M., Braza, J. D., Hinks, M., Hammerling, M. J., Ponnapati, M., Rodriques, S. G., & White, A. D. (2024). Language agents achieve superhuman synthesis of scientific knowledge. arXiv:2409.13740.

[23] Chiang, Y., Hsieh, E., Chou, C.-H., & Riebesell, J. (2025). LLaMP: Large language model made powerful for high-fidelity materials knowledge retrieval. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing* (pp. 25189–25221). doi:10.18653/v1/2025.emnlp-main.1280

[24] Kononova, O., Huo, H., He, T., Rong, Z., Botari, T., Sun, W., Tshitoyan, V., & Ceder, G. (2019). Text-mined dataset of inorganic materials synthesis recipes. *Scientific Data*, 6, 203. doi:10.1038/s41597-019-0224-1

[25] Qwen Team (2024). Qwen2.5 technical report. arXiv:2412.15115.

[26] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP 2019*, 3982–3992.

[27] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33.

[28] Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., & Liu, Z. (2024). M3-Embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. arXiv:2402.03216.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936

[29] McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2), 153–157.

[30] Wilson, E. B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158), 209–212.

[31] Chroma. ChromaDB: open-source embedding database. `https://www.trychroma.com` (software).

[32] Ollama. `https://ollama.com` (software).

[33] OCRmyPDF. `https://github.com/ocrmypdf/OCRmyPDF` (software).

[34] Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9), 509–517.

[35] Virtanen, P., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17, 261–272.

zenodo:23666936 26 Oct 2026 DOI: 10.5281/zenodo.23666936