

Exploring Agentic AI for CEPC Software Development and Validation

Toward AI-assisted validation, code migration, and software workflows

Weidong Li, Zhihao Li, Tao Lin, Yizhou Zhang, Jiaheng Zou

CEPC Detector and Physics Plenary Meeting

13th May 2026

Outline

- ❖ Motivation: why CEPC software needs automation
- ❖ From AI Assistant to Agentic AI
- ❖ Applications for CEPC software
 - CEPCSW Code Migration
 - Validation Tool Development
- ❖ Risks and Guardrails
- ❖ Summary

Motivation: why more automation?

CEPC software development faces long-term and repetitive engineering challenges:

❖ Complex software integration

- Key4hep-based software stack: Gaudi, EDM4hep, DD4hep, Geant4, ROOT, etc.
- Algorithms may come from different sources: CEPC studies, legacy software, and standalone prototypes.

❖ Long-term evolution

- CEPCSW must support detector optimization, physics studies, and future experiment operation.
- Software interfaces, data models, and algorithms will continue to evolve.

❖ Repeated validation

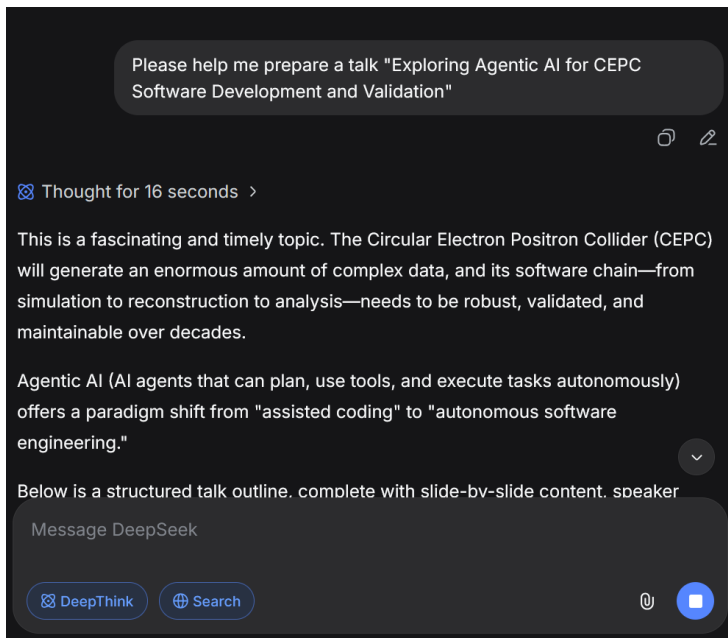
- Each software change requires comparison plots, performance checks, and understanding of differences.
- Many validation tasks are still manual and time-consuming.

Need: more automation for routine coding, testing, and validation workflows. 3

From AI Assistant to Agentic AI

❖ AI assistant

- Examples: ChatGPT, DeepSeek.



Start

Finish

❖ Agentic AI

- Examples: Claude Code, Codex.



Goal

Plan

Edit

Run

Observe

Fix

Report

Example: claude code + DeepSeek v4 pro

```
Planning:
←  Duration  Audience  Goal  Content ↘ Submit →

What is the expected duration of this talk?

> 1. 20-25 minutes
   Short conference talk or seminar presentation
2. 30-40 minutes
   Standard seminar or workshop talk
3. 45-60 minutes
   Full colloquium or lecture slot
4. Type something.

5. Chat about this
6. Skip interview and plan immediately
```

```
Planning:
←  Duration  Audience  Goal  Content ↘ Submit →

Who is the primary audience for this talk?

1. CEPC software developers
   People actively working on CEPCSW – familiar with the framework
> 2. HEP physicists (general)
   High energy physics community, familiar with collider software
   concepts
3. AI/ML + Physics mixed
   Mix of AI researchers and physicists, some domain knowledge gaps
   expected
4. Type something.

5. Chat about this
6. Skip interview and plan immediately
```

```
Planning:
←  Duration  Audience  Goal  Content ↘ Submit →

What's your primary goal for this talk?

> 1. Educate & inspire
   Introduce agentic AI concepts and show what's possible for CEPCSW
2. Report progress
   Present concrete results from applying agentic AI to CEPCSW
   development
3. Propose a research program
   Lay out a vision/roadmap for agentic AI in CEPC software
4. Type something.

5. Chat about this
6. Skip interview and plan immediately
```

```
Planning:
←  Duration  Audience  Goal  Content ↘ Submit →

Do you have concrete demos, benchmarks, or case studies of agentic AI
applied to CEPCSW that should be featured?

1. Yes, with concrete results
   I have specific experiments/demos with measurable outcomes to showcase
> 2. Some initial explorations
   We've done some tests but results are preliminary
3. No, mostly conceptual
   This is a forward-looking talk – the ideas are still being formulated
4. Type something.

5. Chat about this
6. Skip interview and plan immediately
```

Example: claude code + DeepSeek v4 pro

```
• Plan(Design talk structure)
  □ Web Search("Dr.Sai BESIII LLM agent architecture HEP Xiwu 2024 2025")
  Web Search("Gaudi framework CEPCSW algorithm sequence configuration
              pain points developer experience")
  Web Search("HEP software development pain points DD4hep geometry
              detector description verbose")
  ... +6 tool uses (ctrl+o to expand)
  (ctrl+b to run in background)

• Discombobulating... (9m 15s · ↓ 2.3k tokens)
  □ Tip: Use /btw to ask a quick side question without interrupting
  Claude's current work

> █

▣ plan mode on (shift+tab to cycle) · esc to interrupt
```

• Plan updated for Round 1 feedback. Now let me create tasks and start generating the Marp slides.

• **Write(slides.md)**

```
□ Wrote 603 lines to slides.md
  1 ---
  2 marp: true
  3 theme: uncover
  4 class:
  5   - lead
  6 paginate: true
  7 size: 16:9
  8 backgroundColor: "#1a1a2e"
  9 color: "#e0e0e0"
 10 header: ""
  ... +759 lines (ctrl+o to expand)
```

• Let me verify the slides and check if we can render them.

```
Searched for 3 patterns (ctrl+o to expand)
```

• 32 slides total (28 main + 4 backup). Let me try to render them with Marp.

```
Bash(npx --yes @marp-team/marp-cli@latest slides.md --pptx -o slides.pptx
  2>&1 | tail -5)
□ Running... (32s · timeout 2m)
  (ctrl+b to run in background)
```

* **Generating Marp slides...** (18m 49s · ↓ 23.6k tokens)

```
□ ■ Generate Marp slide deck for the talk
  □ Review and refine slides with user
```

> █

```
□□ accept edits on (shift+tab to cycle) · esc to interrupt · ctrl+t to...
X Auto-update failed · Try claude doctor or npm i -g @anthropic-ai/cla...
```

• Done. The first draft of the Marp slide deck is generated at slides.md (22 KB, ~32 slides). Here's a summary of what was created:

Structure: 28 main slides + 4 backup slides

Act	Slides	Content
I: Pain	1-5	Title, Day-in-the-life vignettes, CEPCSW stack diagram, Complexity gap, Transition question
II: Concept	6-10	ChatGPT vs Agentic comparison table, Agent loop flowchart, Toolbox catalog, Limitations (traffic-light table), HEP landscape survey
III: Application	11-19	6 workflow deep-dives (geometry debugging, algorithm dev, validation, migration, analysis, test generation), Early lessons (qualitative), Context quality matrix
IV: Vision	20-28	Vision timeline, 4 infrastructure components, 5-phase roadmap, Risk/mitigation table, Collaboration graph, Closing
Backup	B1-B4	Agent internals, Model comparison, Cost analysis, ML-fast-sim complementarity

Files created:

- slides.md - Marp markdown source (editable, version-controllable)
- slides.html - Rendered HTML preview

Key design choices:

- Dark theme (#1a1a2e bg) with color-coded emphasis (yellow for key points, green for strengths, red for pain/limitations)
- Every workflow follows the same pattern: pain → what the agent does (numbered steps) → resonance hook
- Slide 18 is framed as qualitative "early lessons" rather than quantitative benchmarks (per your general-impressions feedback)
- No demo slide - time redistributed across workflows and roadmap

This is Round 1. What would you like to adjust? Content, emphasis, structure, visuals, or anything else?

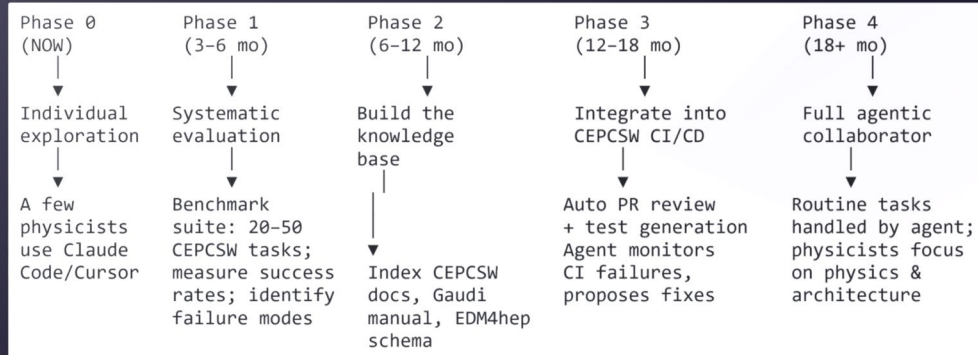
* Cooked for 20m 18s

2 tasks (1 done, 1 in progress)

✓ ~~Generate Marp slide deck~~

■ **Review and refine slides**

Phased Roadmap



Example cost in this test: 1.36M tokens, about 0.27 RMB after discount. using DeepSeek v4 Pro on the IHEP AI API platform.

模型名称	请求数	Token 用量	费用 (¥)	成功率
deepseek-ai/deepseek-v4-pro	38	1,362,515	0.2644	100%

Agentic AI workflows

- ❖ An agentic AI workflow is a process where an LLM-based app executes multiple steps to complete a task.

Andrew Ng (吴恩达), 2024

Essay-writing example:

Write an essay outline on topic X

LLM

Do you need any web research?

LLM

+

web search

Write a first draft.

LLM

Consider what parts need revision or more research.

LLM

+

request human review

Revise your draft.

LLM

 DeepLearning.AI

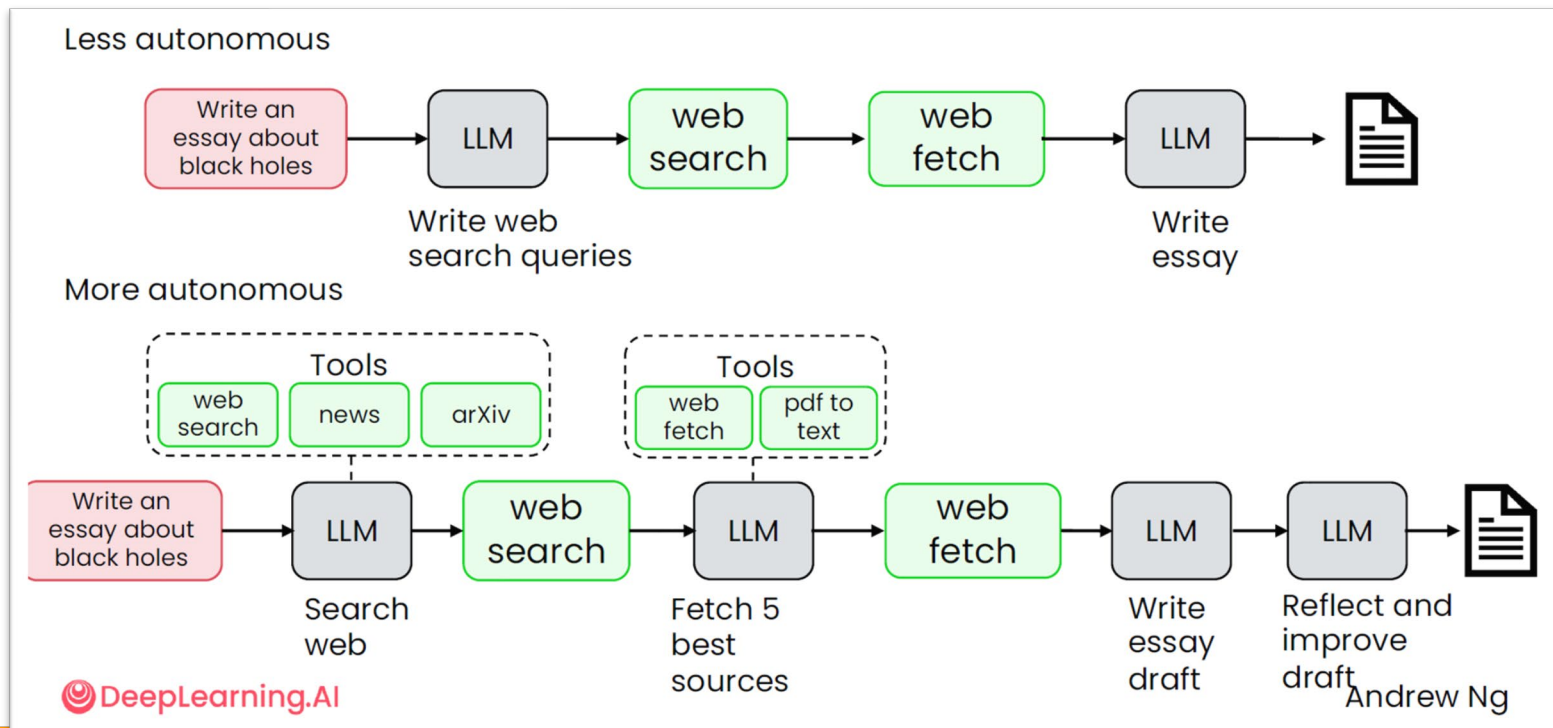


Note: slides from the course “Agentic AI”:
<https://learn.deeplearning.ai/courses/agentic-ai/information>

Degrees of autonomy

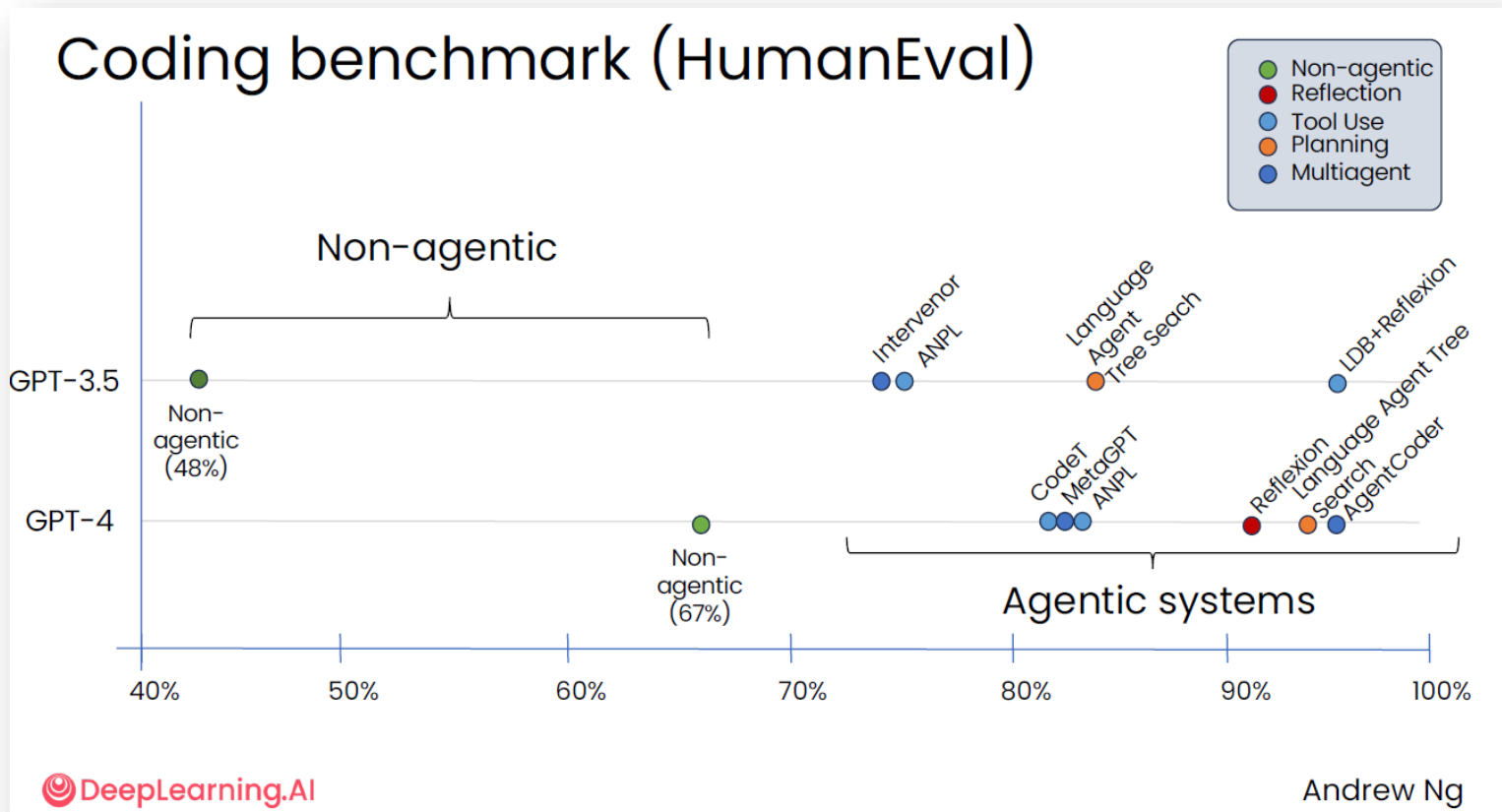
❖ Agentic AI can be less or more autonomous

- Less: All steps and tools are predefined; autonomy is in text generation
- Semi: Agent can make some decisions, choose tools; All tools predefined
- Highly: Agent makes many decisions autonomously



Benefits of agentic AI

❖ Much better performance



Applications of agentic AI for CEPCSW

- ❖ **AI agents do not replace expert judgment**; they automate repetitive software steps under human supervision.
- ❖ For CEPCSW users: sample production and validation
 - Generate configuration and steering files
 - Produce small MC simulation samples
 - Run standard validation workflows
 - Generate plots, tables, and reports
 - Reduce the barrier to using CEPCSW
- ❖ For algorithm developers: coding, debugging, and migration
 - Search examples and understand CEPCSW conventions
 - Generate code skeletons and configuration files
 - Build, test, debug, and iterate
 - Accelerate development and migration of algorithms

Demo 1: CEPCSW Code Migration

- ❖ The Gaudi core API has changed.
 - GaudiAlg has been removed from recent Gaudi versions.
 - For compatibility, a legacy Algorithm interface is still available.

Gaudi/Algorithm.h (modern interface)

```
73  ▾  /** The action to be performed by the algorithm on an event. This method is
74      |  |  invoked once per event for top-level algorithms by the application manager.
75      |  |  */
76  virtual StatusCode execute( const EventContext& ) const = 0;
```

GaudiKernel/Algorithm.h (legacy compatibility interface)

```
57  /// Alias for backward compatibility
58  class Algorithm : public Gaudi::details::LegacyAlgorithmAdapter {
59  public:
60      using Gaudi::details::LegacyAlgorithmAdapter::LegacyAlgorithmAdapter;
61      // this is to explicitly hide the other signature of `execute`
62      StatusCode execute() override = 0;
63  };
```

Demo 1: CEPCSW Code Migration

- ❖ Agentic AI can help perform repetitive migration tasks across the CEPCSW codebase

- https://code.ihep.ac.cn/cepc/CEPCSW/-/merge_requests/284

Agentic workflow

1. Identify affected packages
 - Find packages depending on GaudiAlg.
2. Modify packages one by one
 1. Remove GaudiAlgLib from CMakeLists.
 2. Update the algorithm base class.
 3. Fix related include files and namespaces
3. Build and debug iteratively

Migrate from GaudiAlg to GaudiKernel/Algorithm

lintao@ihep.ac.cn requested to merge lintao/gaudi-algo-interface into master 2 weeks ago

Overview 1 Commits 23 Pipelines 3 Changes 112

Compare master and latest version 112 files +225 -241

Files 112

Search (e.g. *.vue) (Ctrl+P)

Analysis

- AnalysisPID
 - CMakeLists.txt +1 -2
- ReadDigi
 - src
 - ReadDigiAlg.cpp +3 -3
 - ReadDigiAlg.h +2 -2
 - CMakeLists.txt +0 -1
- TotalInvMass
 - src
 - TotalInvMass.cc +3 -3
 - TotalInvMass.hh +2 -2
 - CMakeLists.txt +0 -1
- Detector/DetGeomMetaWriter/src
 - DetGeomMetaWriter.cpp +3 -3
 - DetGeomMetaWriter.h +2 -2

Analysis/AnalysisPID/CMakeLists.txt @@ -3,8 +3,7 @@

```
3 3 # Modules
4 4 gaudi_add_module(AnalysisPID
5 5 SOURCES src/AnalysisPIDAlg.cpp
6 - LINK Gaudi::GaudiAlgLib
7 - Gaudi::GaudiKernel
8 + LINK Gaudi::GaudiKernel
9 + DataHelperLib
10 DetSegmentation
11 DetInterface
```

Analysis/ReadDigi/src/ReadDigiAlg.cpp @@ -6,7 +6,7 @@

```
6 6 DECLARE_COMPONENT(ReadDigiAlg)
7 7
8 8 ReadDigiAlg::ReadDigiAlg(const std::string& name, ISvcLocator*
9 - : GaudiAlgorithm(name, svcLoc),
10 + : Algorithm(name, svcLoc),
11 + _Evt(0)
12 {
13 declareProperty("MCParticle", m_MCParticleCol, "MCParticle
14 collection (input)");
15 }
16 @@ -159,7 +159,7 @@ StatusCode ReadDigiAlg::initialize()
```

112 files modified. 225 lines changed.

This demonstrates an important capability: the agent can search the codebase, apply systematic changes, build the software, and iterate over errors.

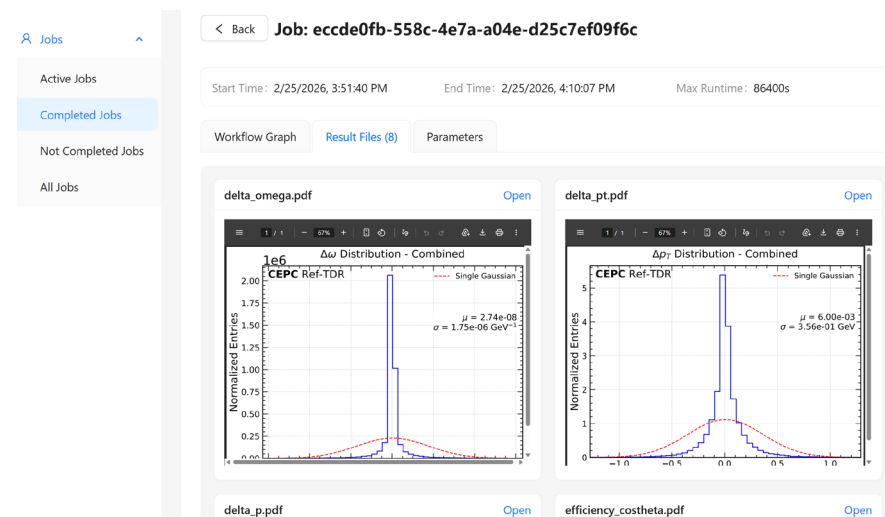
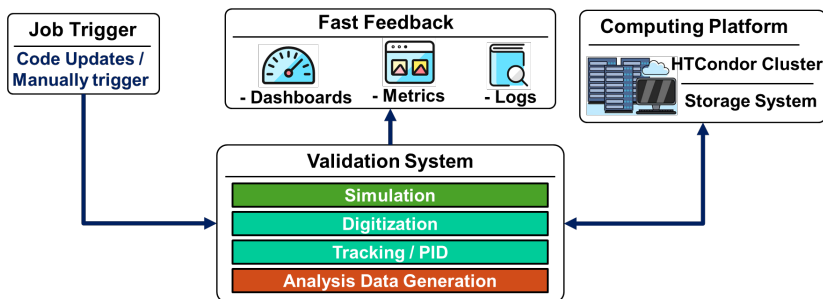
Demo 2: Validation Tool Development

❖ Validation workflow

- It includes simulation, digitization, reconstruction, physics performance analysis, and result comparison.
- Manual validation is often time-consuming, hard to reproduce, difficult to trace, and too slow for frequent software updates.

❖ Agentic AI assisted validation prototype.

- Generate workflow scripts
- Run selected CEPCSW steps
- Produce comparison plots & reports



W.D. Li, CEPC workshop Lisbon

Risks and Guardrails

Risks	Guardrails
Wrong APIs or incorrect logic.	code review + compile tests + CEPCSW reference examples
Code compiles but change physics behavior.	Validation plots + benchmark samples + expert physics review
Non-reproducible results	Fixed releases + fixed samples + recorded commands and configurations
Too many files changes or overwrite	Sandbox branch + approval before commit, push, or merge request
Exposure of private code/data/documents	Approved platforms + restricted access
Unsafe or expensive commands	Predefined tools + limited shell/storage/batch access

Summary & Plan

❖ Summary

- CEPC software will require extensive validation, migration, and long-term maintenance.
- Agentic AI can automate repetitive software loops: search, code, build, test, debug, and report.
- The practical path is supervised: [AI prepares, humans review, tools validate.](#)

❖ Near-term Plan

- Train more users to use agentic AI tools effectively.
- Explore agentic AI in CEPCSW development workflows.
- Use agents to support migration to newer software stacks.
- Deploy and improve the validation tool prototype.