



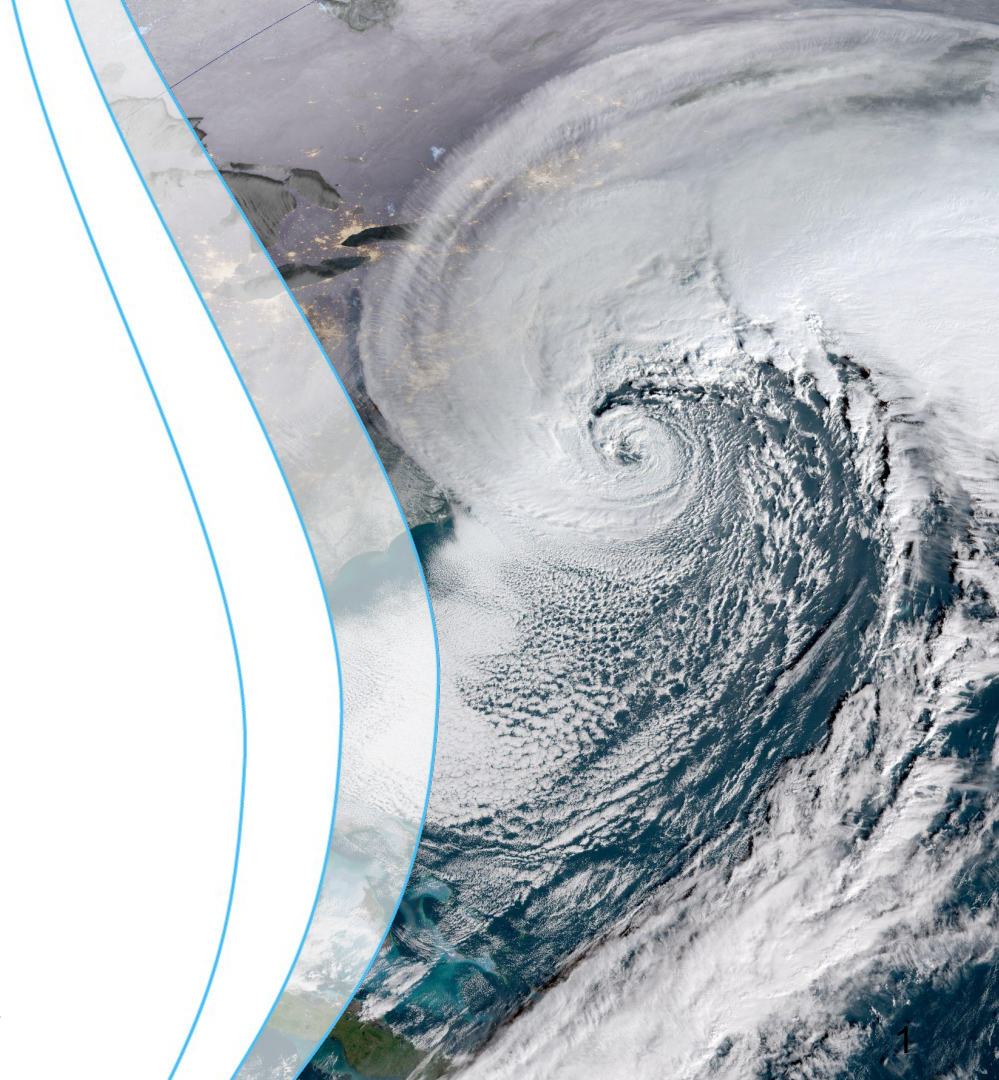
**NCAR**  
OPERATED BY UCAR

March 5, 2026

# Experience with AI tools

John Dennis

*This material is based upon work supported by the NSF National Center for Atmospheric Research, a major facility sponsored by the U.S. National Science Foundation and managed by the University Corporation for Atmospheric Research. Any opinions, findings and conclusions or recommendations expressed in this material do not necessarily reflect the views of NSF.*



# Overview of experience



- Using them for 9-12 months
- Tools
  - Chatgpt [free version, in incognito]
  - Copilot [Integrated into Github]
- Primarily used for pair-programming/thought partner
- Have not used them to transform large code volumes
- Varying levels of success Quality of experience
  - Code exploration [very bad]
  - CESM run modifications [bad]
  - Transplanting CAM physics [ok]
  - Locating memory leaks [good]
  - Generating formal grammar in C++ [amazing]

- Copilot prompt:
  - What does this `init()` method do?
  - Can you commit this change?
- Results
  - Despite have access to a github repo, made up some random stuff
  - Agent refused to perform a simple/trivial operation
- Conclusion:
  - Copilot through GitHub appears to be useless

- Chatgpt prompt: How do I activate [*a particular feature*] in CESM3?
- Results
  - Very confident of its answers
  - Gave me things that might work for CESM1 or CESM2
  - Mostly just made things up that sounded reasonable.
  - Complete and utter waste of time
- Conclusion:
  - Ask an expert would have gotten a correct answer significantly faster.
  - Was likely suffering from insufficient training material

- Chatgpt Prompts:
  - I would like to transplant a CCPP physics package into the E3SM Atmosphere Model in C++. How would I do that?
  - What is this pseudo-density field in EAMxx, I can not find it in CAM.
- Results
  - Gave me a nice overview of some of the key differences been the formulations of CAM and EAMxx
  - Was sufficiently accurately to get me started.
- Conclusion:
  - Save me significant amount of time versus reading all the documentation
  - While not sufficient, I did learn what questions to ask/investigate

# Locating memory leaks



- **Chatgpt Prompts:**

- A large Fortran application is leaking memory. How do I locate the source.
- Tell me more about the Linux virtual memory system
- Could you create a Fortran callable function that will writing out information from /proc/status file system?
- Can you help me understand the output of heaptrack?

- **Results:**

- Allowed the creation of reusable software that has been used by multiple people to identify excessive memory usage in CESM [e.g. CTSM initialization]
- Allowed the isolation of Fortran language error in the NVHPC compiler that leaked memory

- **Conclusion**

- Significantly accelerated a very challenging debugging problem

- **Chatgpt Prompts:**
  - I would like to use a formal grammar specification to parse MOM6 configuration files in C++. What do you suggest?
  - Tell me about why you suggest PEGTL versus ANTLR?
  - Can you you help me modify my grammar to support this feature?
  - Please explain this 200 line C++ error message
- **Results:**
  - Allowed me to leverage my conceptual knowledge by handling the details
  - Did not replace deep understand of the task [formal grammar specification]
  - I need to nudge it when it got stuck
- **Conclusion**
  - Allowed the creation of a sophisticated templated C++ code by a non C++ programmer in several days

# General thoughts

- I like to use the word **specific**
  - Prompt: I have a specific question about the Fortran language standard.
  - If you don't than it will ramble on endless about general material
- Always set the context first
  - Prompt: I have a **specific** about the Message Passing Interface, you know MPI.
- Trust but verify the answer
  - Prompt: Are you sure that Fortran really has that intrinsic? I can not find it.