Axiom Me a Question!

The Effectiveness of Visual Language Models in the Generation of Axioms for Systems Diagrams

ERIN SMITH CRABB, KEEGAN FREEMAN, LOGAN SHAMBERGER, DANIEL DAKOTA

AI4SE 2025

OUTLINE

- The Problem
- The Key Questions
- Axioms
- Protocols
  - System Diagrams
  - Question Categories
- Experiment
- Findings
- Limitations & Reproducibility
- Conclusion

THE PROBLEM

- Systems engineering (SE) can be viewed as a set of principles that have a logical and ordered flow to yield a functional working system
- Pain Points:
  1. Non-engineer users can find diagram interpretation challenging, relying on experienced systems engineers to visually inspect a system diagrams and "translate" it for them
  2. Current methods for supporting SE activities with AI use different techniques to enable information retrieval, such as model translation, direct API interactions, and treating model content as code

THE KEY QUESTIONS

- Is there a way to (1) increase base LLM understanding without the need for complex training efforts and therefore (2) lower the barrier for adoption of SE by non-experts?
- What if we only have the images or select views, rather than the full models?

THE KEY QUESTIONS

- Is there a way to (1) increase base LLM understanding without the need for complex training efforts and therefore (2) lower the barrier for adoption of SE by non-experts?
- What if we only have the images or select views, rather than the full models?

And this leads us to the following question…

Can we use Visual Language Models (VLMs) to translate diagrams into statements of logical operations?

AXIOMS

An axiom, as defined in this context, is a statement that is inferred strictly from the diagram image.

Each statement is assumed to be relevant and follow a logical order to capture the overall representation and flow of information encoded in the diagram.

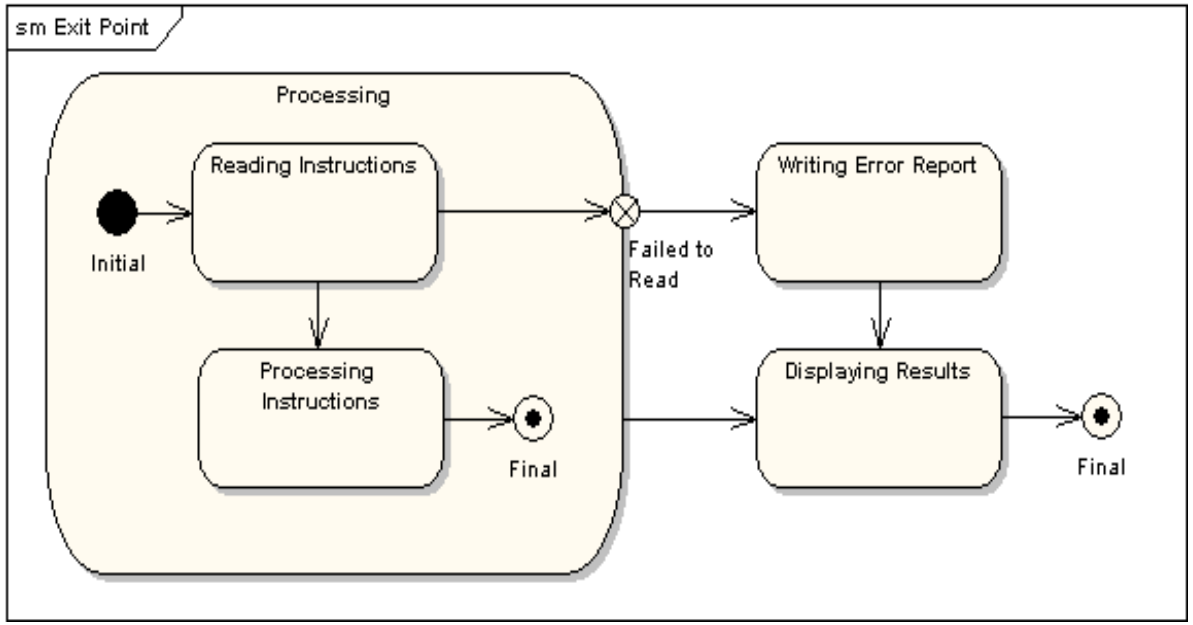

Figure 1: State Machine Diagram for an "exit point"

https://sparxsystems.com/resources/tutorials/uml2/state-diagram.html

AXIOMS

1. The state machines begins at the initial state
2. The "Initial" state transitions to the "Reading Instructions" state

- Understanding is assessed by following the directional relationship between connected entities.
- This axiom defining process can continue until all unique states are covered with any associated transitions until a termination point and final axiom is reached.
- While developing such a set of axioms may be intuitive to a systems engineer, our focus here is how effectively a VLM can mimic this.

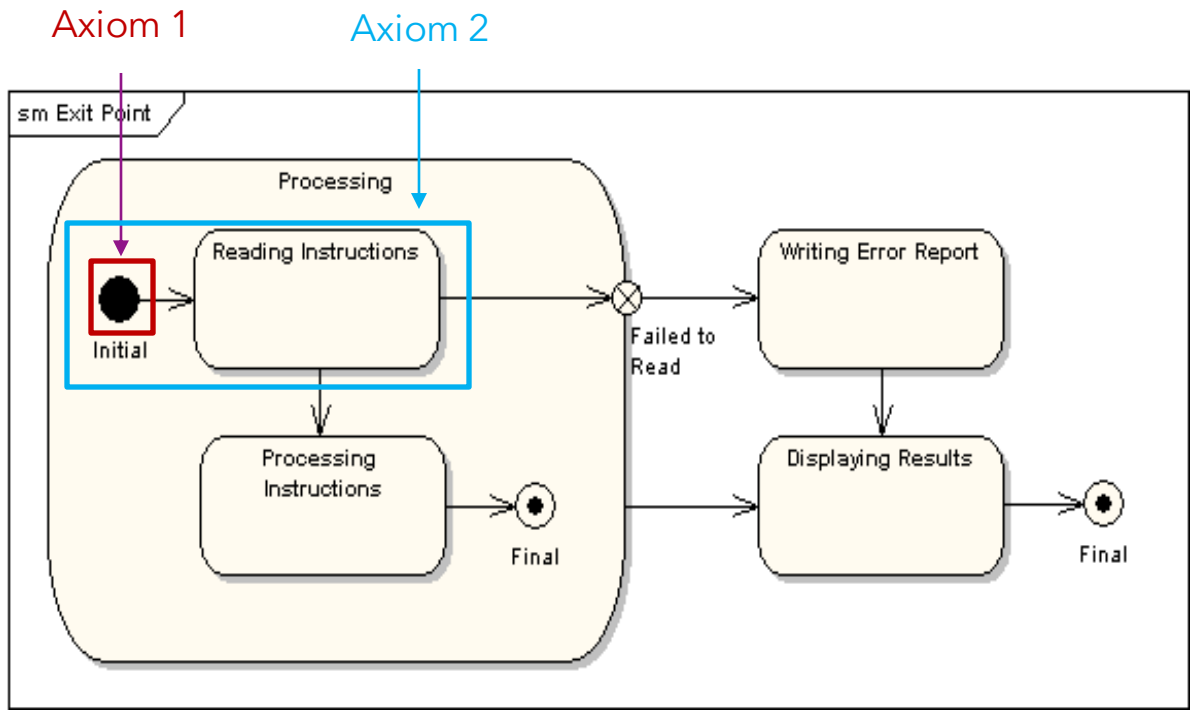

Figure 1: State Machine Diagram for an “exit point”

PROTOCOL

What?	Which?
VLM Models	GPT-4o and Claude Sonnet 3.5
Participants	4 (2 juniors and 2 seniors)
Question categories	States, behaviors, transitions, and overall
Diagrams	5 state machine diagrams of different complexity levels

PROTOCOL

What?	Which?
VLM Models	GPT-4o and Claude Sonnet 3.5
Participants	4 (2 juniors and 2 seniors)
Question categories	States, behaviors, transitions, and overall
Diagrams	5 state machine diagrams of different complexity levels

- Participants were given each of the 5 diagrams (such as seen in Figure 1) and the two corresponding axiom lists generated by each model.
- They were then asked to evaluate the VLM's axioms against their own logic and understanding.
- Asked to identify what in the axioms were correct, missing, extra, and so on

PROMPT FOR AXIOM GENERATION

“Regarding the {systems_diagram}, please create a list of axioms. The axioms should be suitable for advanced professionals, and contain all information available in the given area of interest. Please only include factual relationships demonstrated on the image. Think step by step. Read the image from the top to bottom. If there are directional relationships between entities, make sure that the axiom reflects them. Be sure to include general information that could be useful in validating the diagram integrity. Return only the list of axioms."

PROMPT FOR AXIOM GENERATION

“Regarding the {systems_diagram}, please create a list of axioms. The axioms should be suitable for advanced professionals, and contain all information available in the given area of interest. Please only include factual relationships demonstrated on the image. Think step by step. Read the image from the top to bottom. If there are directional relationships between entities, make sure that the axiom reflects them. Be sure to include general information that could be useful in validating the diagram integrity. Return only the list of axioms."

This seems simple enough...

...but we all know there is an art to prompting!

PROMPT FOR AXIOM GENERATION

“Regarding the {state_machine_diagram}, please create a list of axioms. The axioms should be suitable for advanced professionals, and contain all information available in the given area of interest. **Please only include factual relationships demonstrated on the image**. Think step by step. **Read the image from the top to bottom**. **If there are directional relationships between entities, make sure that the axiom reflects them**. **Be sure to include general information that could be useful in validating the diagram integrity**. Return only the list of axioms."

AXIOM SCORING EXAMPLE

Based on Figure 1:

Question: “How many unique states are there?
Correct Answer: Eight

- Quantitative assessment performed pertaining to the VLM generations as scored by users.
- Those that required numerical answers were cross tabulated using numerical values.
- Questions that were binary were scored in a binary manner for tabulation.
- Statistical tests were performed to identify consistency and trends in user responses.
- Participants completed qualitative assessment questionnaire to assess acceptability and overall impressions regarding axiom lists.

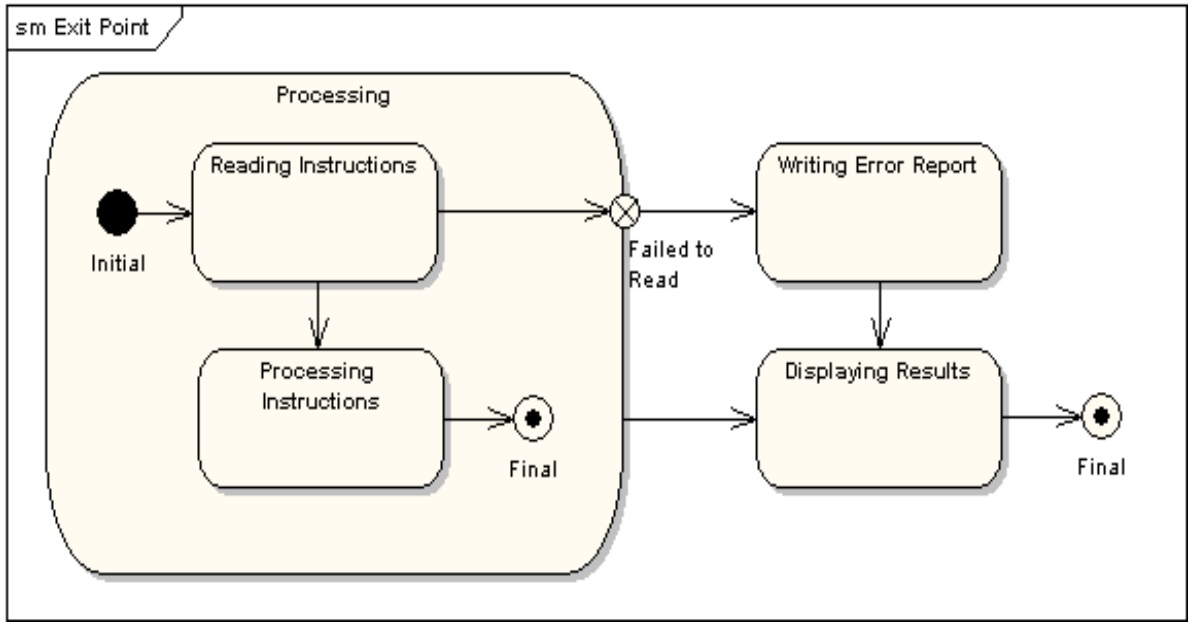

Figure 1: State Machine Diagram for an “exit point”

KEY RESULTS

- VLMs are able to do this task with a promising level of accuracy, although they are not perfect
- Claude Sonnet more closely aligns with system engineers' expectation
- Interclass Correlation Coefficient (ICC) across the two sources of information used to answer the questions
  - Questions that participants use only the image as the source
  - Questions that require both the image and the generated axiom lists
  - Strong ICC of .68 and .67 respectively for each source.

KEY RESULTS – STATES BY DIAGRAM

Diagram 1

Models	Image	Claude	GPT4o
Total	6.5	7.75	7
Missing	---	0	0.75
Extra	---	0.75	0.75
Orphan	---	2	2

Diagram 2

Models	Image	Claude	GPT4o
Total	9.75	9	9
Missing	---	0.75	0.75
Extra	---	0	0
Orphan	---	0.75	0

Diagram 3

Models	Image	Claude	GPT4o
Total	6.25	6.5	6
Missing	---	0.25	0.25
Extra	---	0.5	0
Orphan	---	0	0

Diagram 4

Models	Image	Claude	GPT4o
Total	4.5	4	4
Missing	---	0.5	0.5
Extra	---	0	0
Orphan	---	0	0

Diagram 5

Models	Image	Claude	GPT4o
Total	8.75	8.25	8.25
Missing	---	0.5	0.5
Extra	---	0	0
Orphan	---	0.25	0

KEY RESULTS – BEHAVIORS BY DIAGRAM

Diagram 1

Models	Image	Claude	GPT4o
Total	0.5	0.25	0.5
Missing	---	0	0
Wrong	0	0.25	0
Orphan	---	0.25	0

Diagram 2

Models	Image	Claude	GPT4o
Total States	6.5	7	7
Missing	---	1.25	1
Wrong	0	2	0.75
Orphan	---	0.75	0.5

Diagram 3

Models	Image	Claude	GPT4o
Total	2.5	2.5	3
Missing	---	0.25	0.75
Wrong	0	0	0.75
Orphan	---	0.25	0.5

Diagram 4

Models	Image	Claude	GPT4o
Total	9	9	8.25
Missing	---	0	0.5
Wrong	3.5	4	3.75
Orphan	---	0.25	0.5

Diagram 5

Models	Image	Claude	GPT4o
Total	7.25	8.25	6
Missing	---	1.5	2
Extra	---	1	0.25
Orphan	---	1	0

KEY RESULTS – TRANSITIONS BY DIAGRAM

Diagram 1

Models	Image	Claude	GPT4o
Total	7.5	7	6.75
Missing	---	1	0
Extra	---	1	0
Orphan	---	1	0

Diagram 2

Models	Image	Claude	GPT4o
Total States	16.75	14.75	14
Missing	---	1	0
Extra	---	1	0
Orphan	---	1	0

Diagram 3

Models	Image	Claude	GPT4o
Total	9.75	8.25	10
Missing	---	0	1
Extra	---	0	1
Orphan	---	0	1

Diagram 4

Models	Image	Claude	GPT4o
Total	13.5	13.75	14.5
Missing	---	1	0
Extra	---	1	0
Orphan	---	1	0

Diagram 5

Models	Image	Claude	GPT4o
Total	17.25	12	15
Missing	---	1	0
Extra	---	1	0
Orphan	---	1	0

QUALITATIVE ASSESSMENT

- **All four engineers** strongly **preferred Claude** over GPT-4o
  - Its axiom lists were considered superior by the majority of users for four of the five diagrams
  - **The exception** was "exit point", which **was the simplest of the systems diagrams** (naturally resulting in the shortest axiom list), suggesting that Claude may be superior as the systems diagrams increased in complexity
  - Preferred axiom lists tended to be more explicit, but not necessarily more verbose (e.g., Diagram 5)
  - Better axiom lists had more obvious mistakes – they were easier to correct and took less time to work with overall
- Two noted shortcomings based on the provided feedback were identified
  - **VLMs struggled with decision gates**, where the LLM tended to select a single outcome or fail to follow outcomes more extensively
  - **Directed transitions which ran opposite the flow of analysis** (e.g., an arrow than pointed up, when the VLMs were processing their information top-down, as is standard) **were frequently omitted from the statement lists, even when the LLMs were instructed to process in both directions**

LIMITATIONS AND REPRODUCIBILITY

- Team only experimented with two models and a limited set of systems diagrams and evaluative questions
  - Do not want to suggest that the findings would hold to all VLMs, especially to smaller models
- Generations from VLMs provides an area of variability due to specific parameters (e.g., temperature)
  - Affect the quality and consistency of the responses
  - Should be considered when determining how reliable and consistent a response may be
  - Further refinements to the prompt may improve

CONCLUSIONS

- This pilot demonstrated that **VLMs show promise in their ability to understand and capture information from systems diagrams and translate that axioms using natural language** that follow a logical flow.
- **Assorted VLMs may have different utilities and benefits** to diverse problems facing the systems engineering community
- **The basis of the results provides a strong foundation for additional directions of research in applying VLMs to help continue to modernize old models** and support informational retrieval application in the multi-modal spaces that system engineering inhabits.