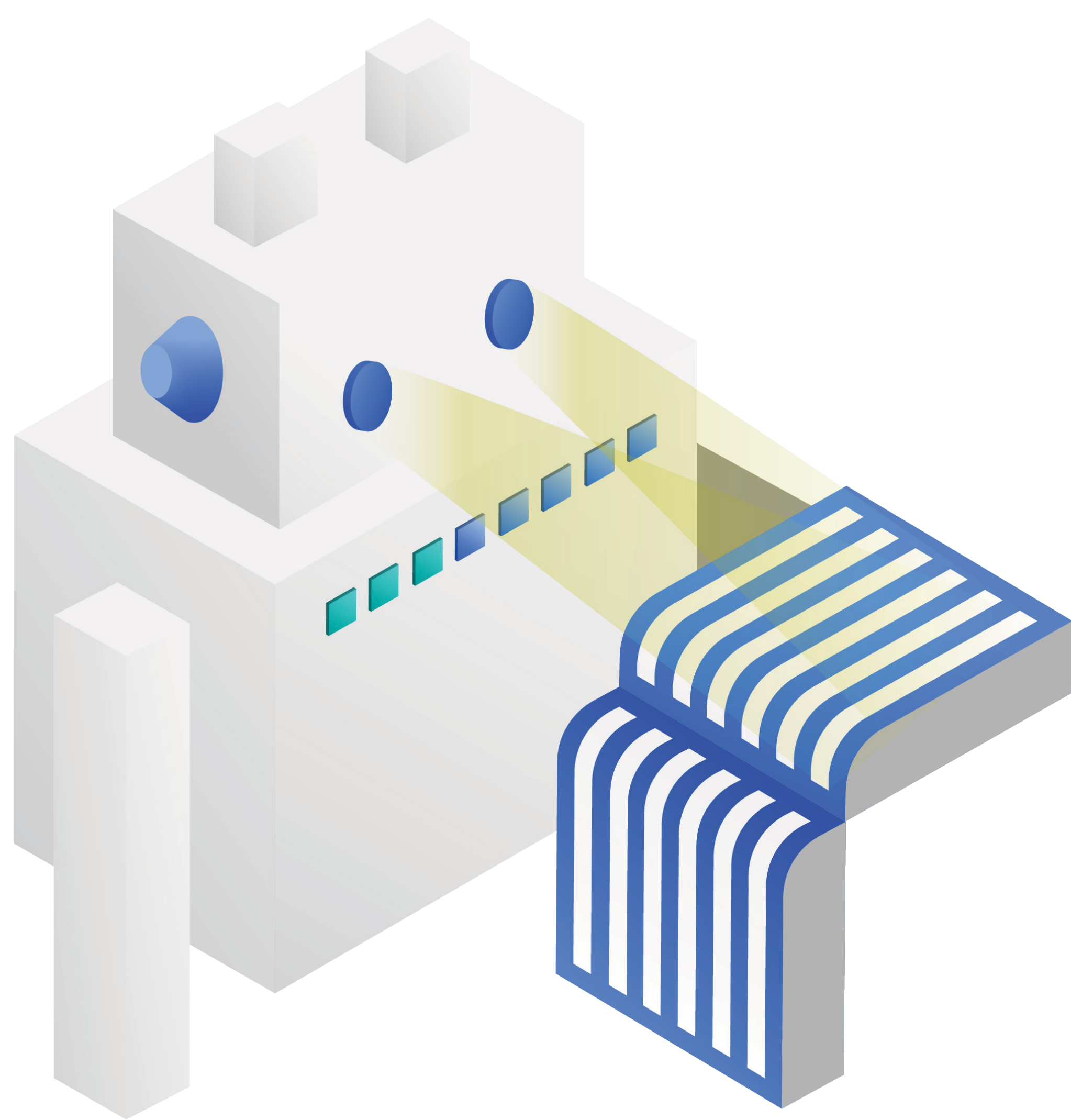


# OPTIMIZING CONTENT FOR RAG

## Traditional content design process

1. Identify concepts to clarify
2. List tasks to explain
3. Collect supporting reference details
4. Write concept, task, reference topics
5. Review with team for feedback

Traditional content design doesn't guarantee RAG success



## New requirements for RAG

6. Collect real user questions
7. Test that RAG can answer those questions grounded in your topics
8. Create or update content as needed
9. Update writing guidelines based on results

### Real user questions

Collect questions real human users have asked - from places like workshops, support tickets, and user communities.

- Users ask surprising questions in quirky, ungrammatical, misspelled, unexpected ways.
- If you try to guess what questions users will ask, your familiarity with your product and its jargon will bias your questions and the way you ask them.

### Testing topics

Just adding content to a topic to answer a question does not guarantee a RAG solution can answer that question with a prompt grounded in that topics.

- You can test a topic by prompting a model to answer collected user questions and then evaluating the response.
- You can also test a topic by prompting a model to list the questions answered by that topic and then comparing the generated questions with collected user questions.

### Writing guidelines

- Add summaries to long topics
- Clearly introduce lists
- Avoid ambiguity
- Explain graphics in text
- Match users' terminology

### Tactical writing

Writing content to answer questions might require deviating from traditional information architecture and writing style.