

Voice User Interface Design

Building User Interfaces

Cole Nelson

Make a copy of today's ICA and share it with your group members!

Learning Objectives

1. Understand how voice user interfaces differ from traditional user interfaces.
2. Be able to apply design principles to voice user interface design.
3. Be able to test voice user interfaces through expert-based testing using heuristics.
4. Be able to test voice user interfaces through user-based testing using experience prototyping.

Introduction to Conversational Interface Technologies

What is a conversational interface technology?

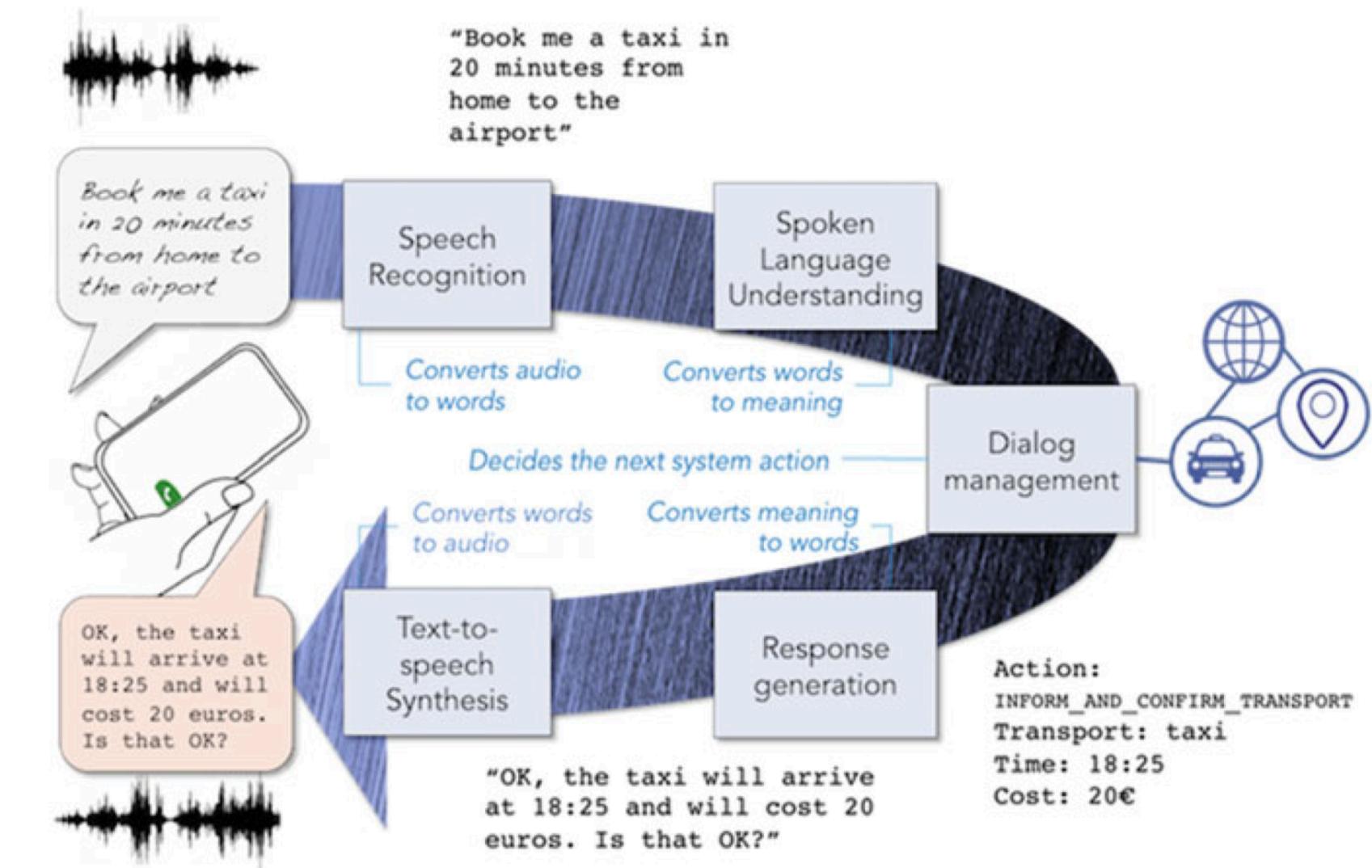
Definition: "Technology that supports conversational interaction with these VPAs [virtual personal assistants] by means of speech and other modalities."¹

¹ McTear et al., 2016, *The Conversational Interface*

Components of Conversational Interfaces²

Conversational interfaces integrate a number of technologies:

- Speech recognition
- Spoken language understanding
- Dialog management
- Response generation
- Text-to-speech synthesis



² Image source: [McTear et al., 2016, *The Conversational Interface*](#)

Platforms for Building Conversational Interfaces

In this course, we will be using Wit.AI by Meta.

However, there are alternatives:

- DialogFlow by Google (Commercial)
- Mycroft (Open Source)
- CoreNLP (Academic)

Design Principles for Conversational Interfaces

Elephant in the Room

Recap: Definition of Usability: The effectiveness, efficiency, and satisfaction with which a specified set of users can achieve a specified set of tasks in a particular environment.— ISO 9241-11

Conversational interfaces are almost always less effective, less efficient, and less satisfactory than graphical user interfaces.

Effectiveness: Conversational interfaces are more error prone due to technology, ambiguities, and environmental influences.

Efficiency: Using conversational interfaces is almost never as fast as using graphical user interfaces.

Satisfaction: Interacting with conversational interfaces can be awkward, socially inappropriate, and frustrating.

So, what is the point of conversational interfaces?

Where do these interfaces deliver value?³

1. In some contexts, e.g., while driving, CIs are more effective, efficient, and satisfactory due to resource constraints.
2. CIs address many accessibility problems, including vision (e.g., blindness), motor (e.g., tremor), and cognitive (e.g., dyslexia) deficiencies.
3. Streamlining app installation, login, payment, notifications, and so on in a conversational paradigm.

³ Grover, 2016, Bots won't replace apps. Better apps will replace apps.

Design Principles

Gricean Maxims⁴

Definition: Proposed by Paul Grice, conversations follow the cooperative principle and four key maxims:

- Maxim of quality (truthful and accurate communication)
- Maxim of quantity (just the right amount of information)
- Maxim of relevance (appropriate and relevant information)
- Maxim of manner (clear, cooperative communication)

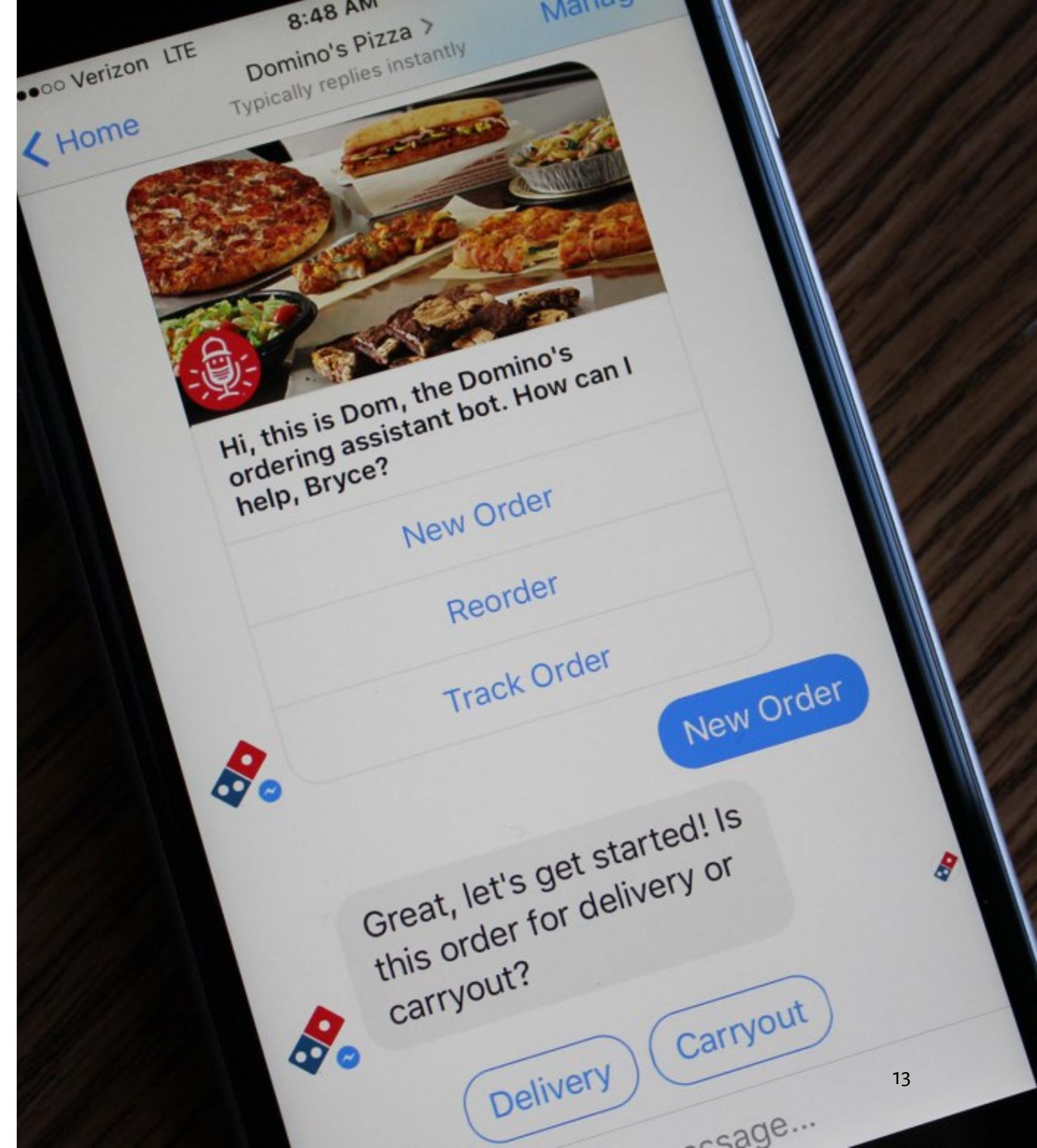
⁴ Grice, 1975, Logic and Conversation

Multimodality⁵

Definition: Multimodal interfaces utilize multiple modalities, including visual information, speech, touch, and so on, in user experiences they afford.

Most conversational interfaces are multimodal interfaces.

⁵ Image source



Multimodality Principle: Take advantage of other modalities, e.g., visual information, vibrations, etc., wherever appropriate.

Using multimodal components, you can provide users with breaks for decision making, interruptions, etc.

Potential caveats:

- Ask, "does my interface still support a speech-only interaction?"
- The conversational and other components must be designed together to fit within the conversation.

Interaction Paradigm

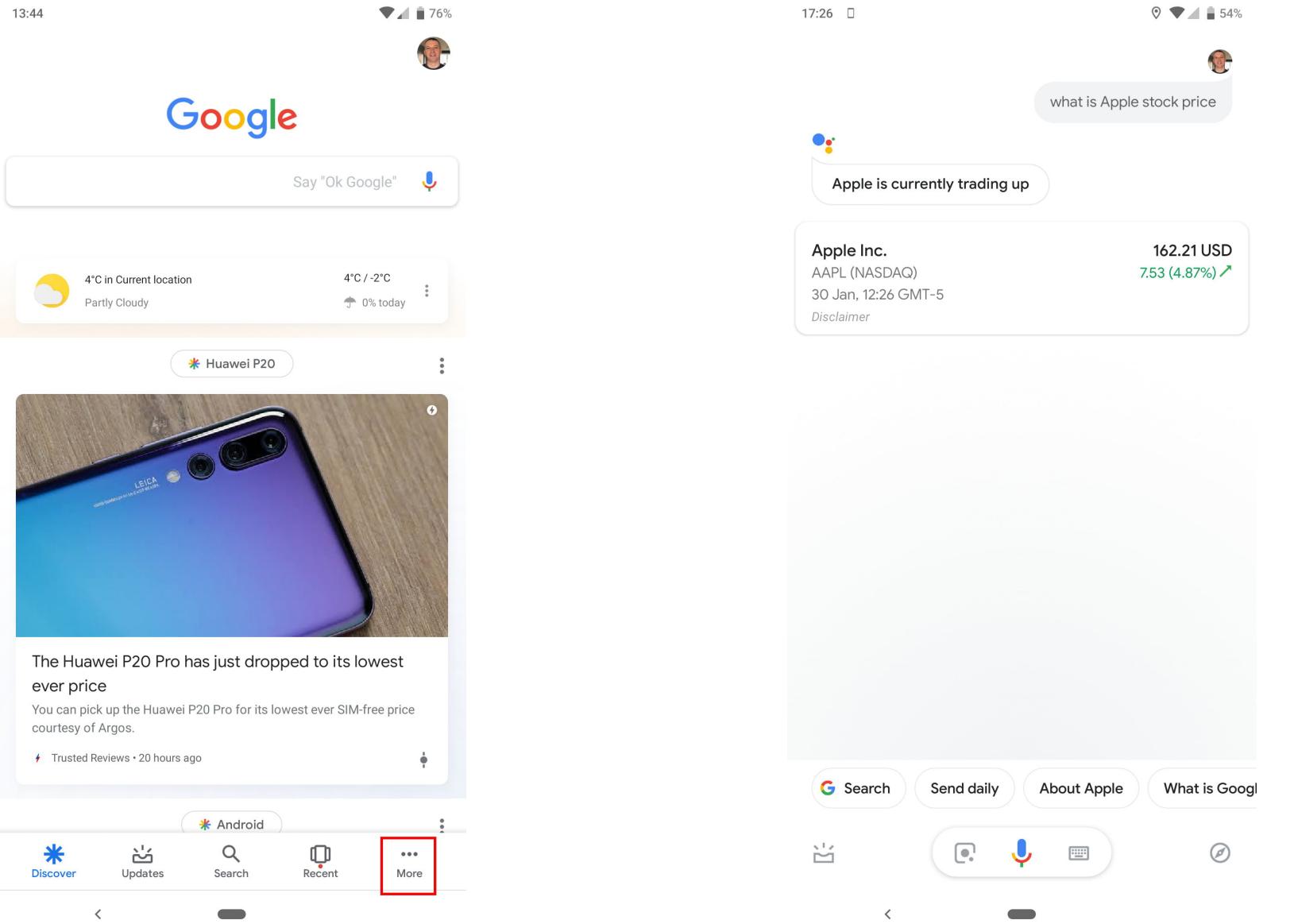
Conversational interfaces can follow different paradigms depending on the context of use and the design of the application:

- Command-and-control interfaces
- Conversational interfaces

Command-and-Control Interfaces

Definition: Interfaces where speech input is mapped to specific system functions that are called immediately. These interfaces commonly utilize:

1. Expressing user intent using a wake word (e.g., "OK, Google") or the pressing of a button (e.g., home button in the iPhone)
2. Indicating listening and understanding
3. Executing the mapped function



⁶Image source: [left](#), [right](#), [middle](#)

Conversational Interfaces

Definition: Interfaces where the interaction with the system has the characteristics of human conversations, including turn taking, theoretically infinite depth, conversational markers, etc.

Turn-taking

Definition: Speaking turns are the core, cooperative structure of conversations that involves one speaker at a time and an explicit exchange of tokens.

Principles:

1. One speaker at a time — *transparency in who is speaking*
2. Turn exchanges — *explicit signaling of who will speak next*
3. Interruption handling — *very difficult with CIs*

Conversational Markers⁷

Definition: Speech cues that indicate the state or the direction of the conversation. Types of conversational markers:

1. Timelines ("First," "Halfway there," "Finally")
2. Acknowledgements ("Thanks," "Got it," "Alright," "Sorry about that")
3. Positive feedback ("Good job," "Nice to hear that")

⁷ Pearl, 2016, Designing Voice User Interfaces: Principles of Conversational Experiences

Assistant: I'll be asking you a few questions about your health. First, how many hours of sleep did you get last night?

User: About seven.

Assistant: Good job. And how many servings of fruits and vegetables did you eat yesterday?

User: Maybe four.

Assistant: Got it. Last question—were you able to take your medication last night?

User: Yes.

Assistant: All right. That's it for now. I'll talk to you again tomorrow. Goodbye.

Confirmations

Definitions: CIs are designed with some forms of confirmation to improve system usability and transparency.

Can be explicit vs. implicit and speech-based vs. non-speech based (visual, action).

Explicit confirmation: Requiring the user to confirm: "I think you want to set a reminder to 'buy insurance before going skydiving next week.' Is that right?"¹⁰

Implicit confirmation: Letting user know what was understood: "Ok, setting a reminder to buy insurance..."

¹⁰ [Image Source](#)

Error Handling

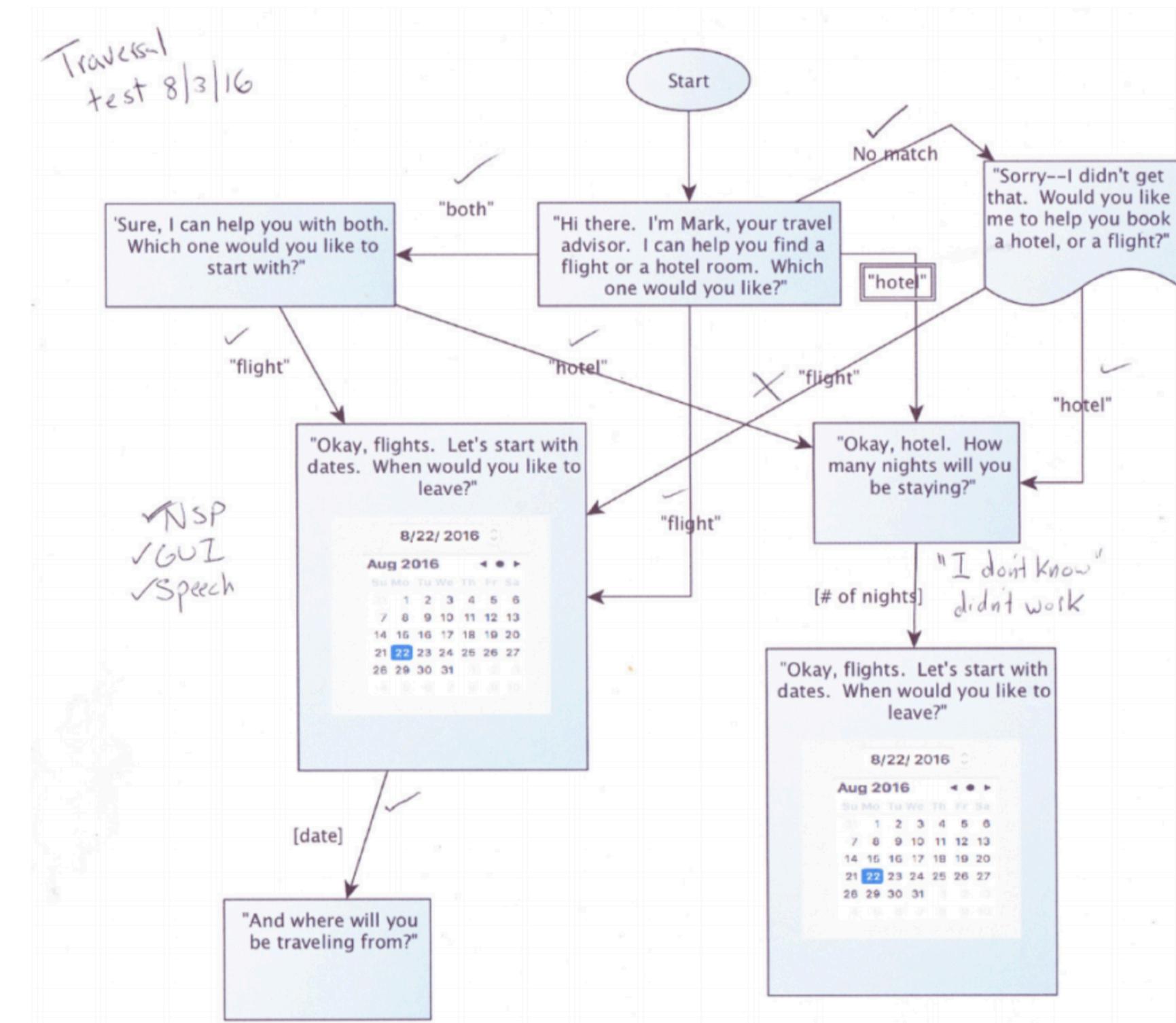
Definitions: Deviations from expected conversational flow due to technical mistakes, unexpected user behavior, environmental influences, etc.

Types of errors:

1. No speech detected
2. Speech detected, but nothing recognized
3. Something was recognized correctly, but the system does the wrong thing with it
4. Something was recognized incorrectly

Flowcharting Conversational Interactions

The most commonly used method of modeling and prototyping conversational interactions is defining flows that show how the interaction will flow depending on system state, user behavior, or external influences.



Heuristics for Conversational Interfaces⁸

Seventeen heuristics that fall into five broad categories:

1. General
2. Conversational style
3. Guiding, Teaching, and Offering Help
4. Feedback and prompts
5. Errors

⁸ Wei & Landay, 2018, Speech-based Conversational Agent Heuristics

CI Heuristic #5

S5: Pay attention to what the user said and respect the user's context.

- Leverage user input when it can be used as a parameter to a command.

User: “Find a flight to San Francisco on June 25th”

Agent: “Searching for flights to San Francisco on June 25th...”

- Remember what the user has said in the current conversation.

User: “What is the weather like in San Jose today?”

- Use context you already know about the user to fill in fields, but confirm them.

User: “*Make a reservation for 2 tonight at Jack's at 8pm*”

Agent: “*Ok, I made a reservation for John Smith for 2 people at Jack's tonight at 8.*”

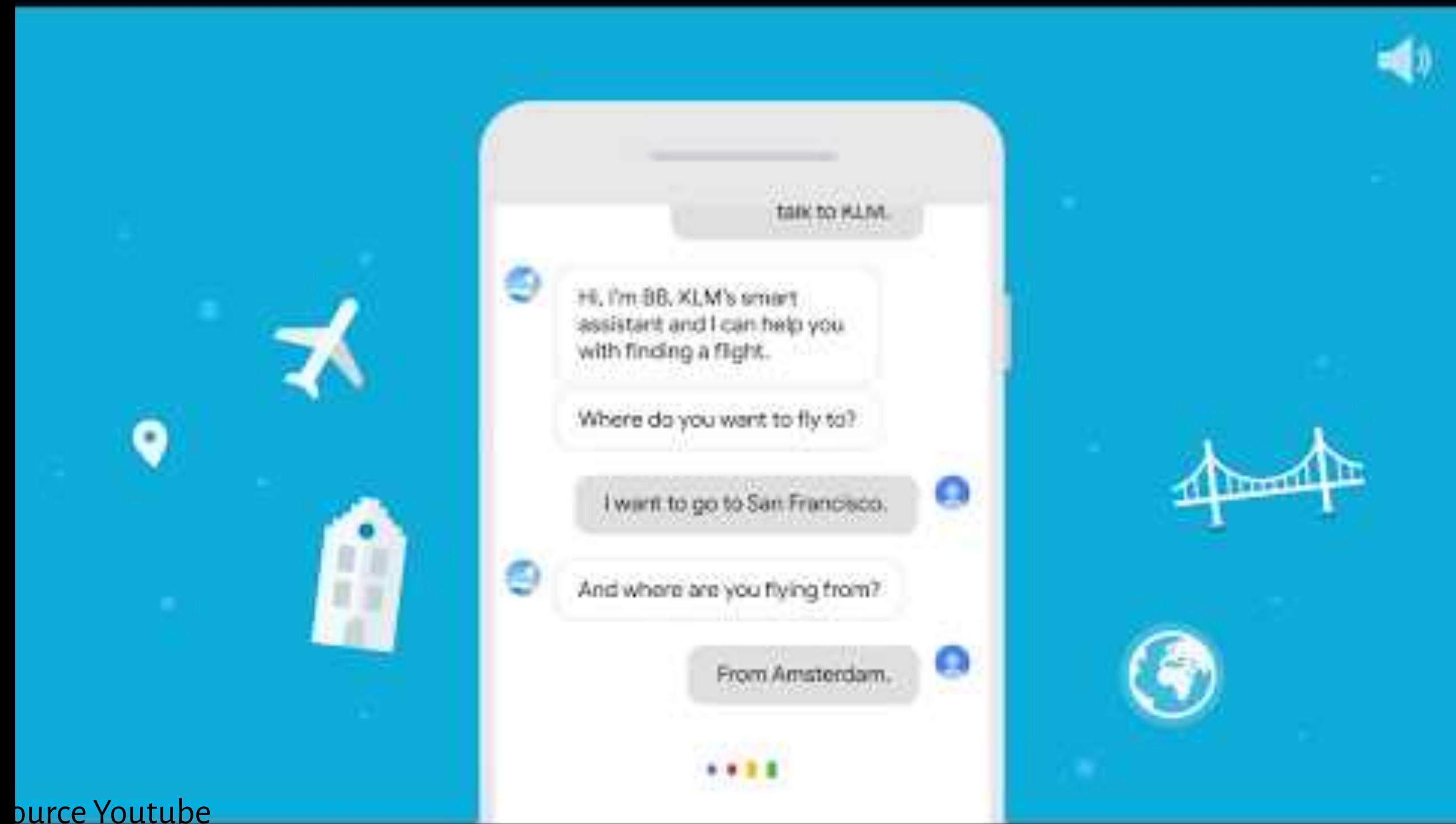
- Use context to respond intelligently (e.g., location/environment, time constraints, # of users, user identity/age).

User: “*What will the weather be like today?*”

Agent: “*The weather in Palo Alto will be sunny today. The high will be 77.*”

S12: Confirm input intelligently.

- Confirm input implicitly through results or next prompt.
- Confirm irreversible or critical actions explicitly and even allow undo after confirmation.



source Youtube

Experience Prototyping

The Problem¹⁰

Conventional prototyping methods provide limited support for conversational interfaces.



¹⁰ [Image Source](#)

Human Dialogue

Social interactions are driven by *tacit knowledge*:¹¹

...we can know more than we can tell...

An evolutionarily encoded and culturally situated set of rules, patterns, and practices for effective interpersonal communication.

How do we design interfaces that follow these rules, patterns, and practices?

¹¹ Michael Polanyi, 1958, *Personal Knowledge*

The Solution: Experience Prototyping

Definition: Prototyping the holistic experience of interacting with a product.

A related definition:¹²

An experience prototype is any kind of representation, in any medium, that is designed to understand, explore or communicate what it might be like to engage with the product, space, or system we are designing.

¹² Buchenau & Fulton Suri, 2000, Experience Prototyping

How does experience prototyping solve the problem of designing interfaces to follow human norms of interaction?

We have tacit knowledge about how conversational interactions work. By *acting out* an interaction, we apply our knowledge to a scenario.

What I hear I forget. What I see, I remember. What I do, I understand!¹³

¹³Chinese proverb by Xunzi:

不闻不若闻之，闻之不若见之，见之不若知之，知之不若行之；学至于行之而止矣

We use ourselves as *decoders* for the norms encoded in us!¹⁴



¹⁴ [Image source](#)

When do we do experience prototyping?

Three key uses:

1. Understanding existing user experiences and context
2. Exploring and evaluating design ideas
3. Communicating ideas to an audience

What is it that we prototype?

1. System behavior
2. User behavior
3. Interactions with context

How do we do experience prototyping?

Step 1: Define context¹⁵

What is the context of the interaction?

E.g., passengers using entertainment system on a bus, travelers packing their luggage.



¹⁵ [Image source](#)

Step 2: Develop Scenarios¹⁶

What are the concrete interaction scenarios that we want to support?

E.g., buying a ticket, users packing, cooking a meal.



¹⁶Image source

Step 3: Identify Design Goals¹⁷

What role does my design play in these scenarios? How does it support the user in the target activity? What capabilities will it offer?

E.g., find, filter, and purchase flights; help the user set and follow personal goals through daily reminders.



¹⁷ [Image source](#)

Step 4: Set up the Environment¹⁸

How can I represent the context of the interaction?

E.g., creating props to represent devices, environmental constraints.



¹⁸ [Image source](#)

Step 5: Act out Interaction¹⁹

How will the interaction unfold? How will the user behave? How should the system behave?



¹⁹ [Image source](#)

Bodystorming²⁰

Definition: *Bodystorming* is a creativity method that involves physically experiencing a situation to develop new ideas and insights.



²⁰Image source

Step 6: Develop Insight²¹

What did you learn about system behavior, user behavior, and interactions with context?



²¹ [Image source](#)

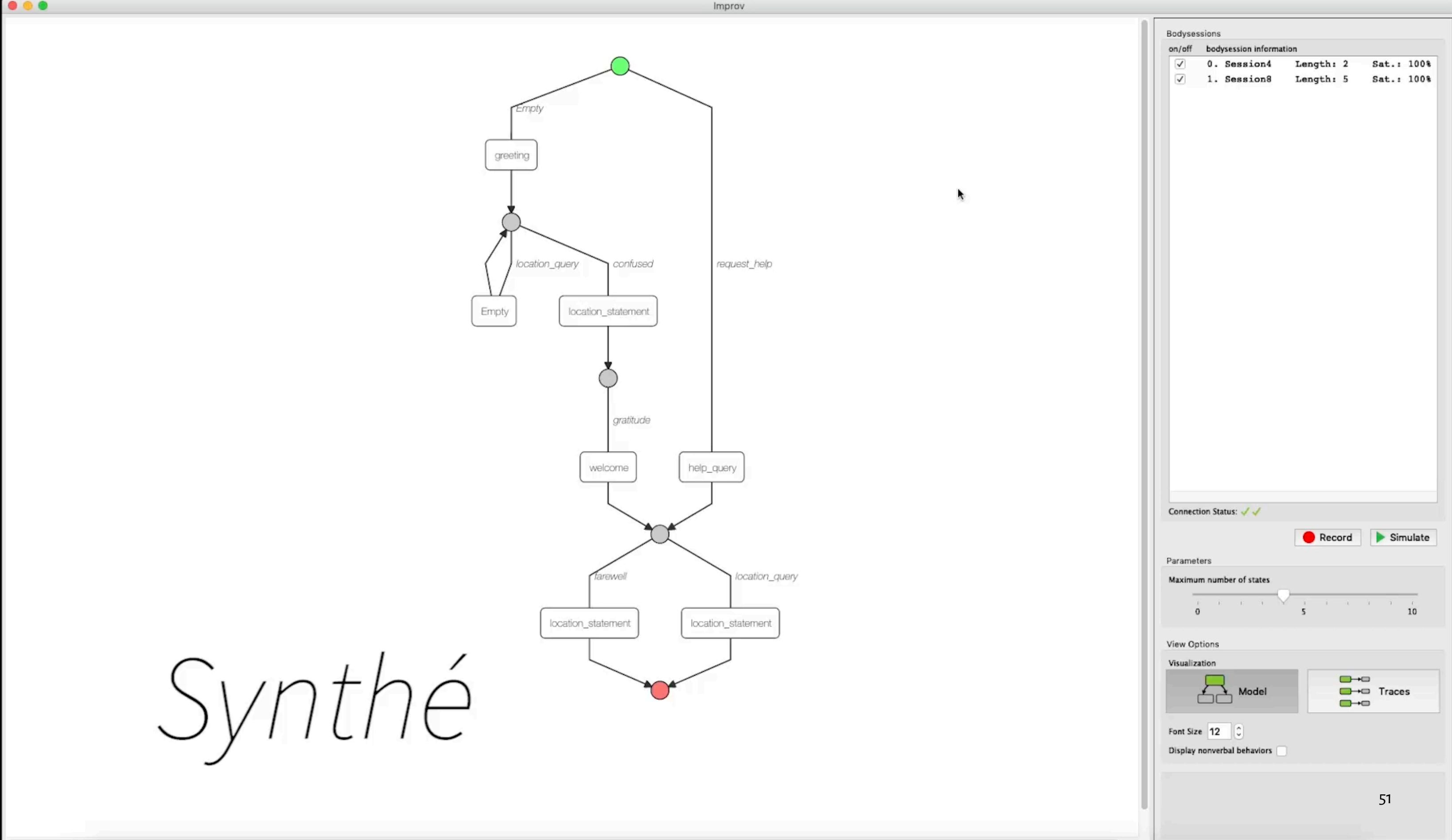
Pro Tip: Experience prototyping is *very* awkward. Get the awkwardness out of your system so that you can focus on using the method for design.

Example Use of Bodystorming²²

Supporting design teams in ideating and acting out human-robot interactions using a system called Synth .

²² Porfirio et al., 2019, Bodystorming Human-Robot Interactions

Synthé



Additional resources

- Seminar paper on experience prototyping
- Case studies in bodystorming
- "Universal Methods of Design": sections on *experience prototyping* and *bodystorming*

Questions?