

SDC19

Where Now Meets Next

Building Bixby Conversational Experiences For Hands-Free Use

Alex Ren

Research Engineer, Viv Labs/Samsung Electronics





Alex Ren

Research Engineer, Viv Labs/Samsung Electronics

Alex's Bio

Table of Contents

- ✗ Recap: MovieAgent capsule
- ✗ Introduction to Hands-free
- ✗ Navigating lists for Result moments
- ✗ Navigating lists for Input moments
- ✗ Summary



Recap: MovieAgent capsule

Demo: Finding a movie to watch

"Hi Bixby, help me find a movie to watch."

Which genre are you in the mood for?

Movie Agent

Today's highlight

Action

More results

Adventure

Animation

Comedy

Crime

Documentary

Drama

None

Help me choose

Movie Agent

Answer these questions and I'll find your perfect movie genre. We can get you first class seats! The cabin crew would like to know how you prefer your steak.

A. Well-done

B. Medium

C. Rare

D. No meat please

Movie Agent

Here are a few animation movies.

Top recommendation

Cars 2006

More results

Toy Story 4

The Lion King

Spider-Man: Into the Spider-Verse

The Addams Family

Movie Agent

Cars

Year 2006

Runtime 117 minutes

Budget

Revenue

Movie Agent

Cars

Year 2006

Runtime 117 minutes

Budget

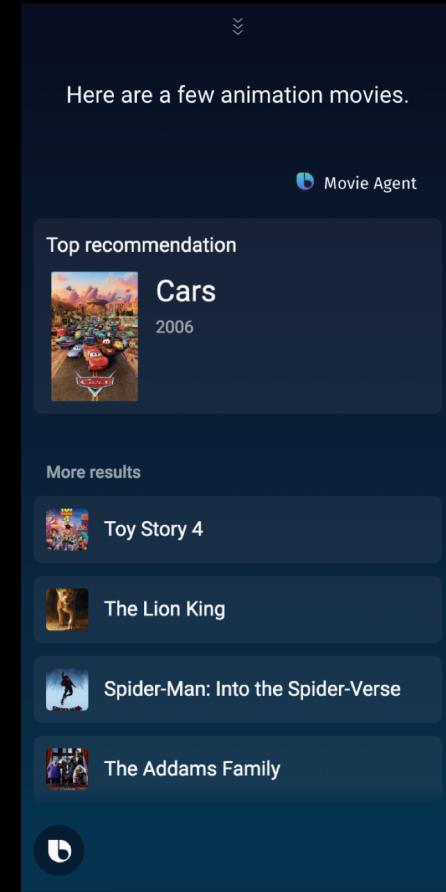
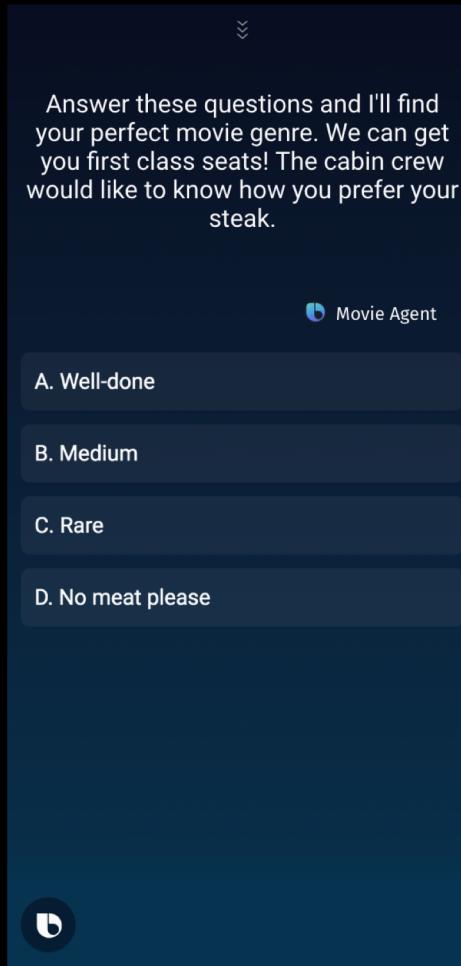
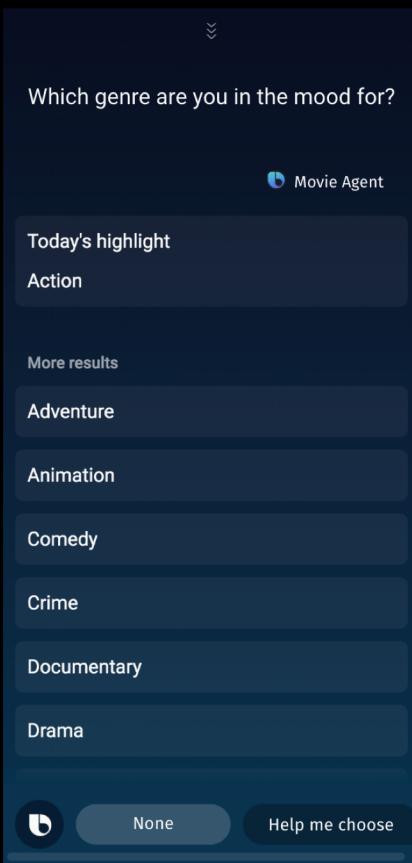
Revenue

#SDC19

Samsung Developer Conference

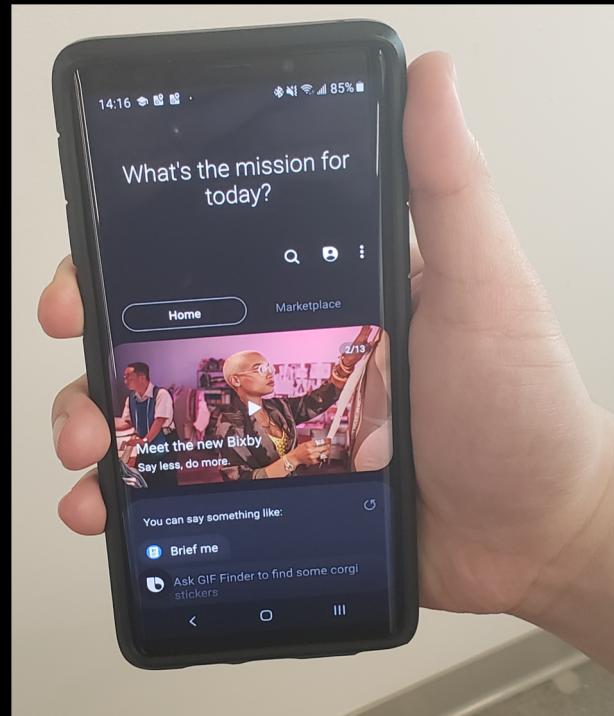
MovieAgent has many lists

- MovieAgent is built up of lists of selections that users can make



Works great with hands-on!

- Users can see results and touch with items on the screen
- They can even continue the conversation with Natural Language if they wish



But what if you can't touch?

- A visual experience can give a ton of information – text, pictures, buttons
- But, it might not always be possible (or safe!) to hold/look at the device.
- So, we next to read out what's on the screen and help the user navigate, hands-free



Key Question

- What can we do to convert MovieAgent into a hands-free experience?



Introduction to Hands-free

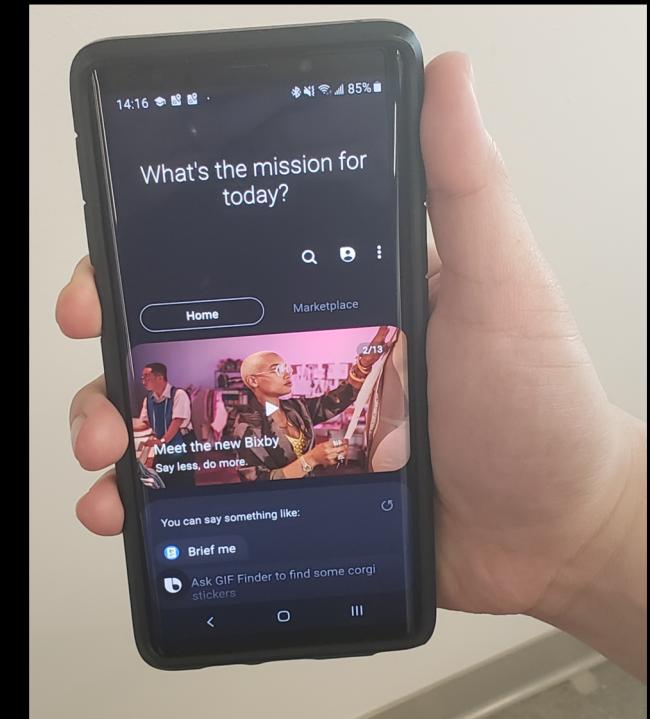
Hands-on vs. Hands-free

We'll go over these concepts in detail

	Hands-on	Hands-free
Woken by	Bixby Button	Wake phrase (Hi Bixby)
Spoken summary is read	no	yes
Mic turns on for user response	no	yes
\$handsFree variable	false	true

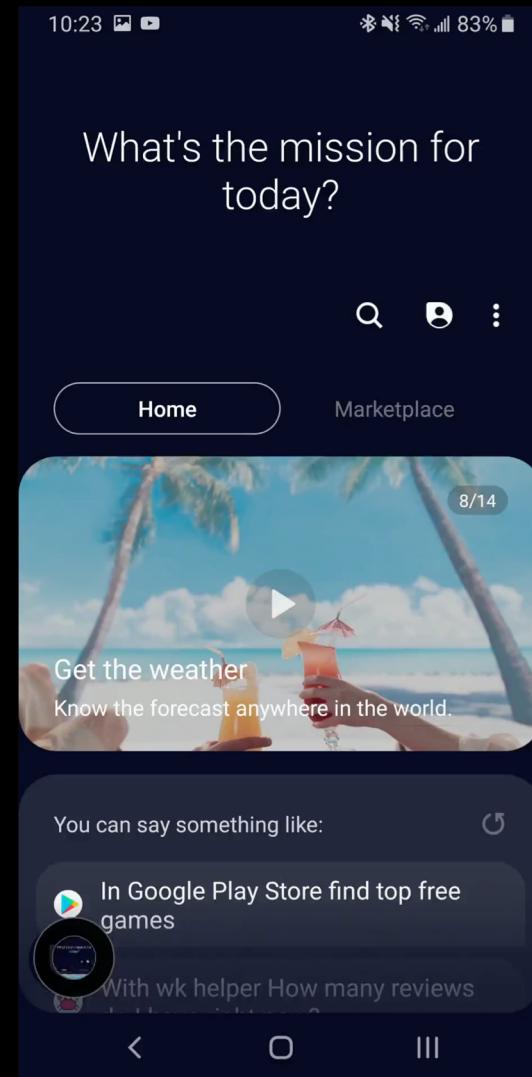
Hands-on Mode

- User is holding the phone, and wake Bixby using the hardware/software button for Bixby
- Implications:
 - User can see and touch the screen
 - User can use the button to wake Bixby to continue interacting with it
- In the code, **\$handsFree** (a variable) will be false



Mic behavior in Hands-on Mode

- User holds button to wake Bixby, then says “Help me pick a movie to watch”
- Bixby won’t start to listen again until user presses Bixby Button.



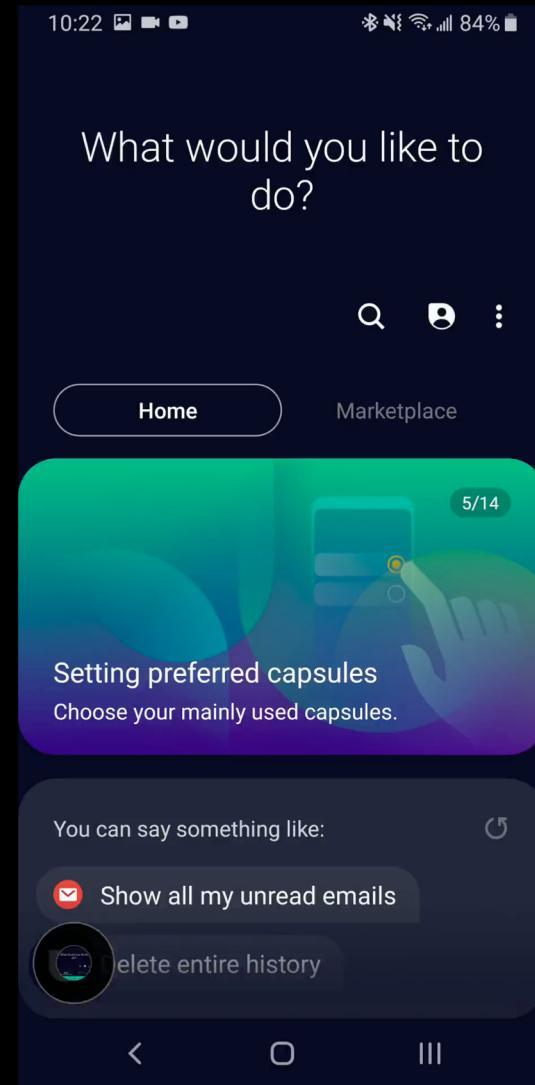
Hands-free Mode

- User says “Hi Bixby” – imagine a user driving, or their phone is not in their hands
- Implications:
 - User might not be able to see the screen, or it might not be safe to look at the screen.
 - User can't reach the device to press the Bixby button.
- In the code, **\$handsFree** will be true



Mic behavior in Hands-free Mode

- User wakes Bixby with “Hi Bixby”
 - Then says, “Help me pick a movie to watch”
- Mic automatically turns on after dialog is finished for the user’s response

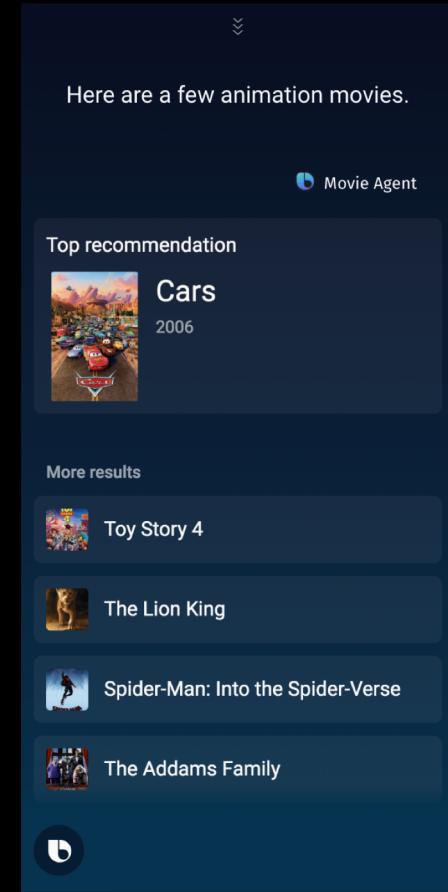
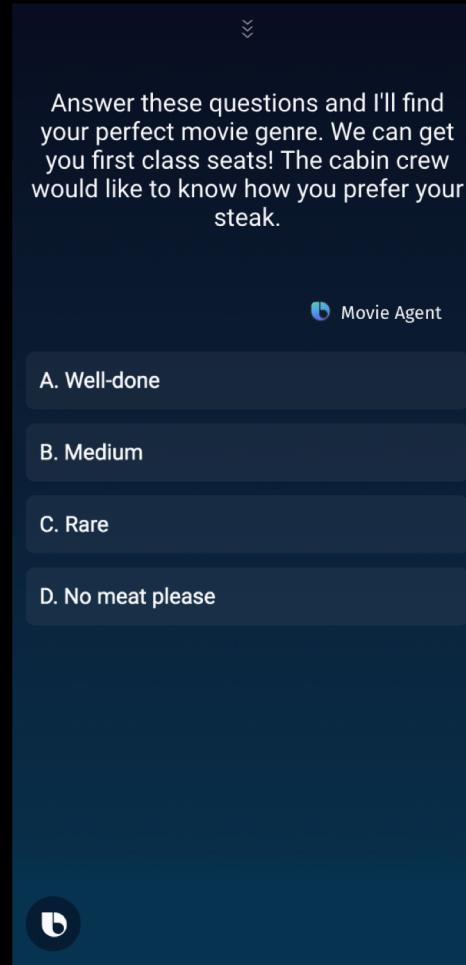
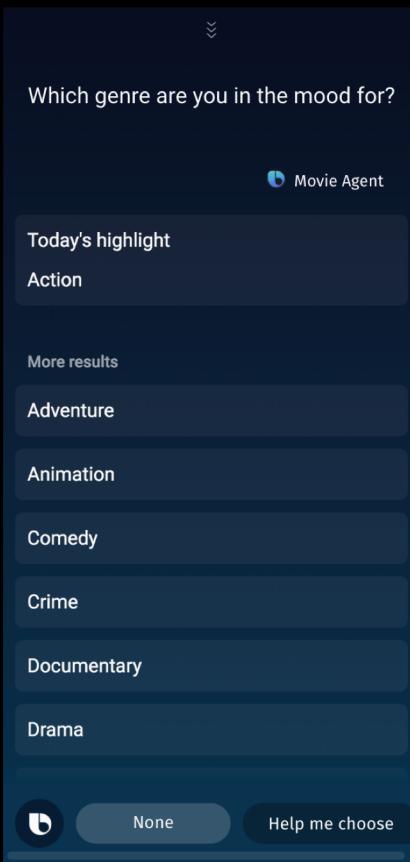




Building out Hands-free

RECAP: MovieAgent has many lists

- MovieAgent is built up of lists of selections that users can make



Introducing navigation-mode

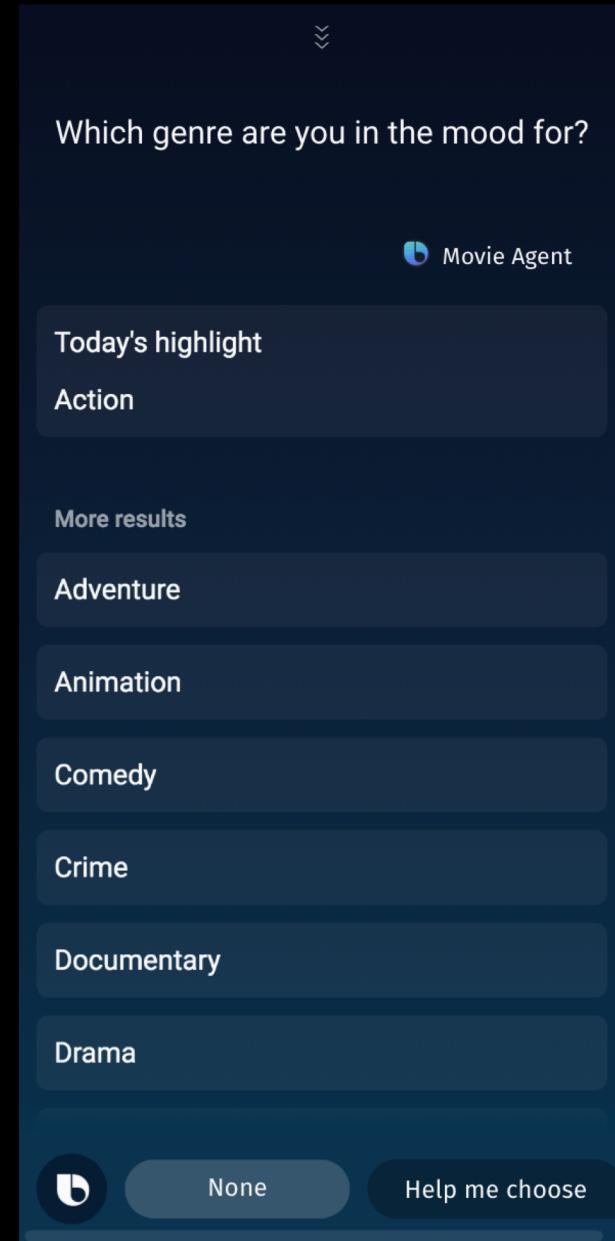
- navigation-mode lets Bixby read out a list of inputs or results
- You define which navigate-mode to use, and the dialog you want at key moments – the Bixby system does the rest.
 - No new models or code required
- Bixby automatically gives you navigation commands – next, previous, selection

Which navigation-mode to use?

- Which mode to use depends on your content

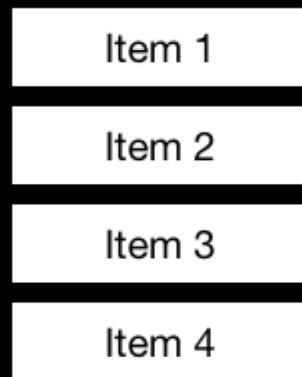
read-none

- Consider - do we need to read out all the genres?
 - No – users asking for a movie are probably already familiar with movie genres
- This is a good use case for “read-none” - or simply, not implementing a navigation-mode.
 - Not everything needs a change when we’re creating hands-free experiences

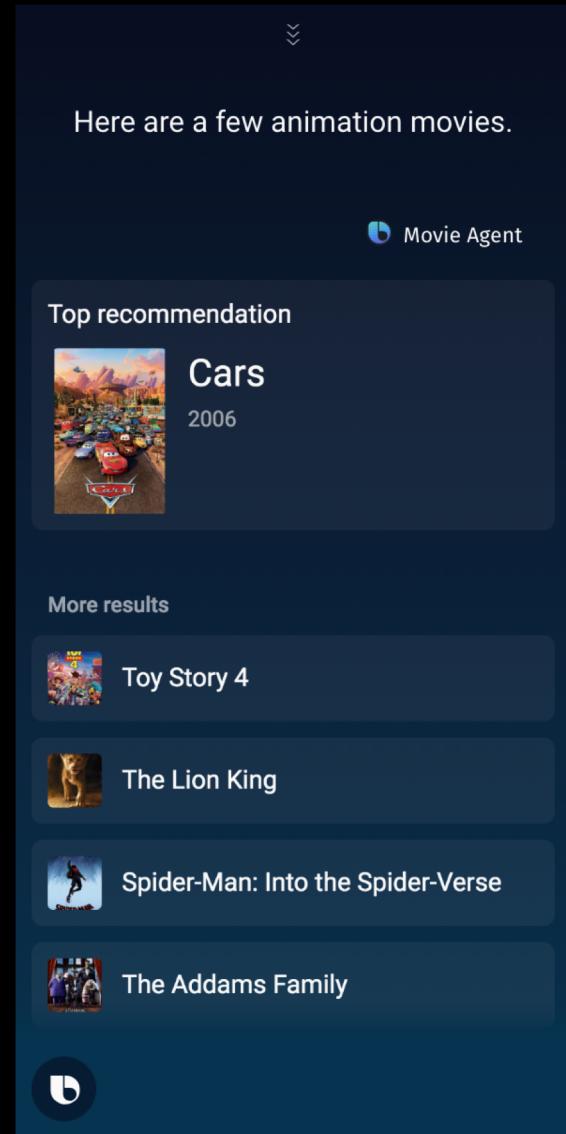


read-one

- Consider – what's the best way to read out each movie title result?
- Should we read-none? No, the user needs to hear the results
- Should we read-many? No, we shouldn't overwhelm the user with too many movie titles
- This is a good use case for “read-one”

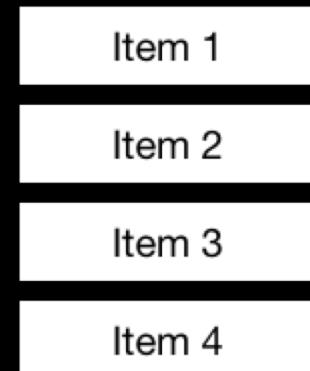


Read one

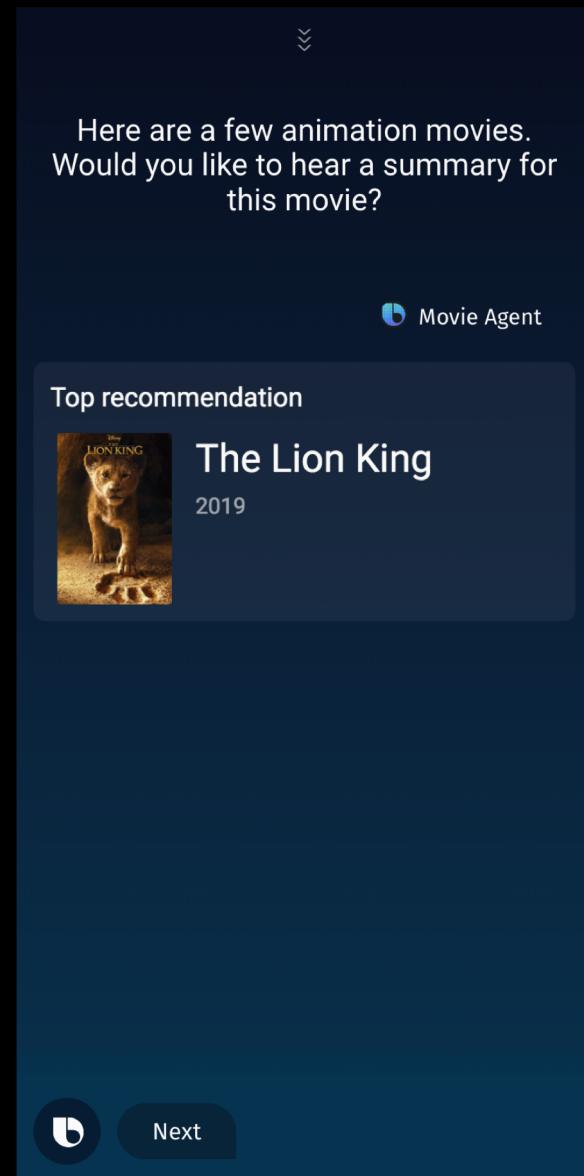


read-one: view changes

- Comes with a visual change – in case the user glances at the screen – only one result at a time.
 - Use \$handsFree to show a different layout in this mode
 - “spoken summary” will be read out loud



Read one



read-one: dialog changes

Here are a few keys that help shape the dialog

list-summary: Here are a few Action movies.

summary: My top recommendation is Spider-Man: Far from Home.

item-selection-question: Would you like to hear a summary for this movie?

(mic turns on for the user to answer)

Item 1

Item 2

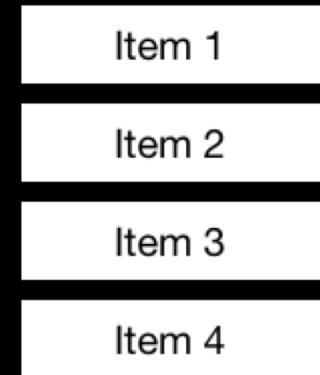
Item 3

Item 4

Read one

read-one: navigation commands

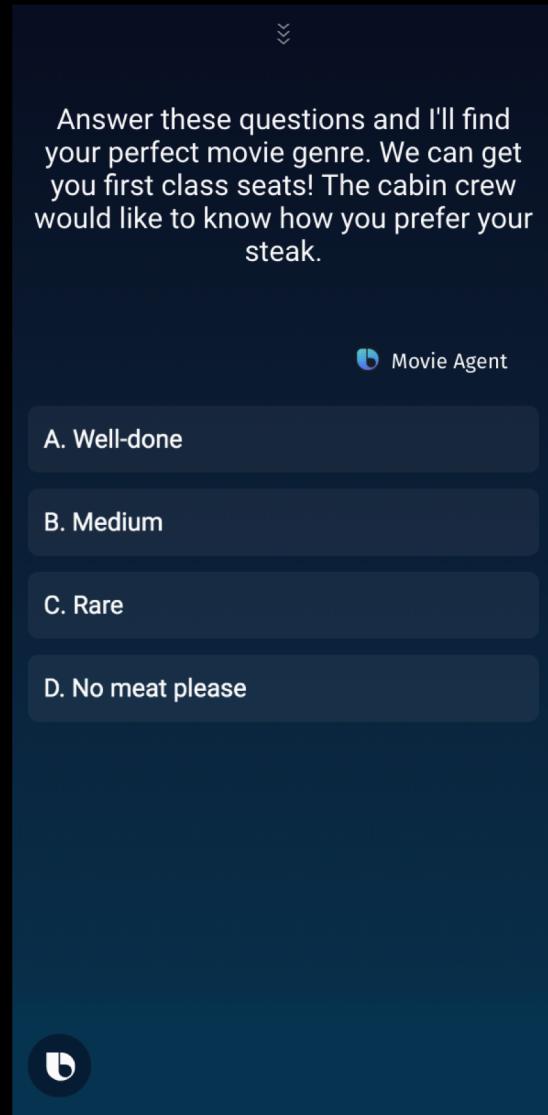
- navigation-mode brings support here for saying “next”, “previous” and such to navigate the results hands-free.
- What happens if the user navigates off of the list?
 - If they go backwards on the first item, that results in an underflow dialog
 - If they go forwards on the last item, that results in an overflow dialog
 - Same principles as above apply to other navigation modes



Read one

read-many

- Consider – how should we read out the list of possible quiz answers?
- Should we read-none? No, the user needs to hear the options.
- Should we read one? No, are different answers we want the user to hear, so they can pick the best answer.
- This is a good use case for read-many



read-many: dialog changes

We can construct this dialog by just adding a few keys.

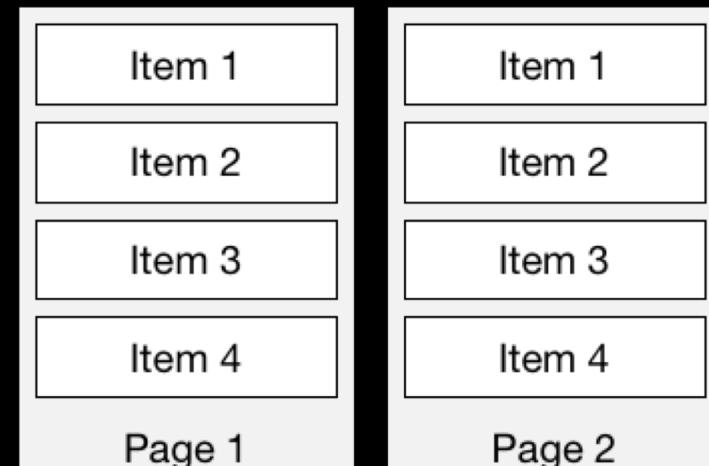
Here are a few keys that help shape the dialog

list-summary: Answer these questions and I'll find your perfect movie genre. What snack would you like for the trip?

summary: For each item, we generate a summary and all are read out loud
A ... Popcorn. B ... Chocolate. C ... Fresh fruits & veggies. D ... Leftover pizza.

item-selection-question: "" (left blank)

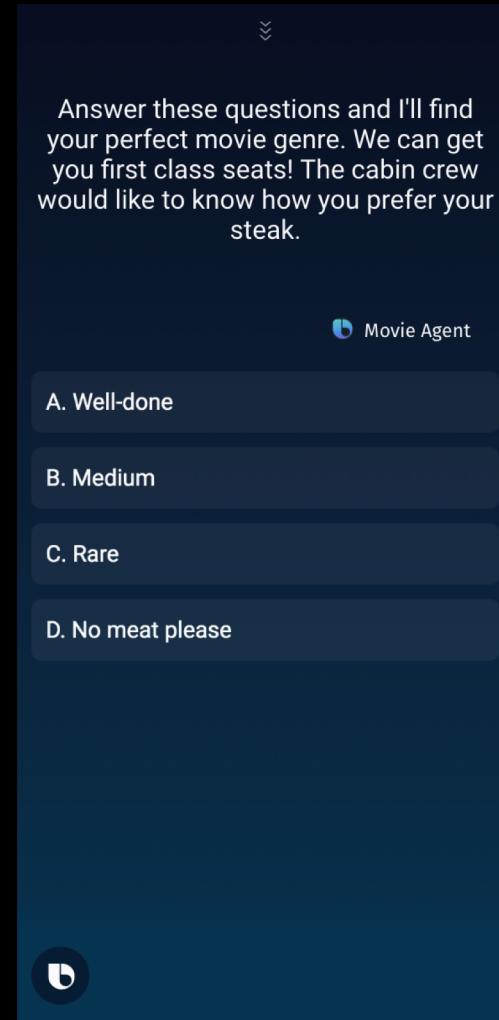
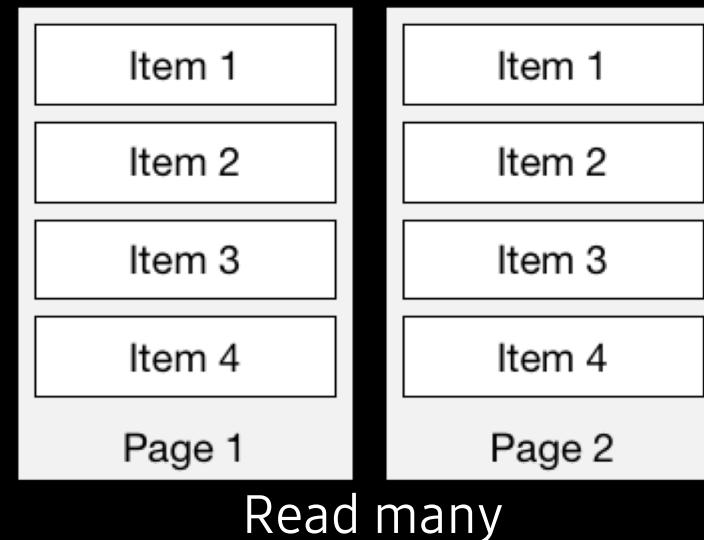
(mic turns on for the user to answer)



Read many

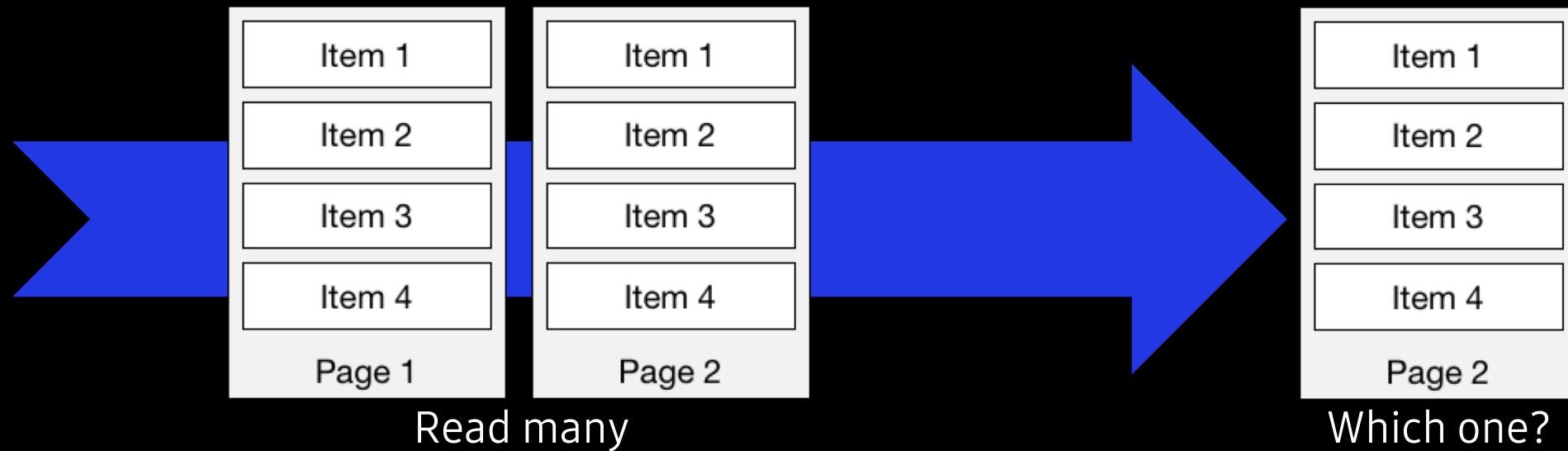
read-many: navigation commands

- In a read many list, ordinal selection comes built-in
 - “first one”, “second one”, etc
- To support any other type of selection, you should implement a filtering action
 - “well-done”, “no meat please”, etc



read-many: advanced pagination

- In read many, there is a “page-size” parameter
- You can have Bixby read items as pages of items
 - The user then confirms the item that they want is on a page, and then selects from that page.



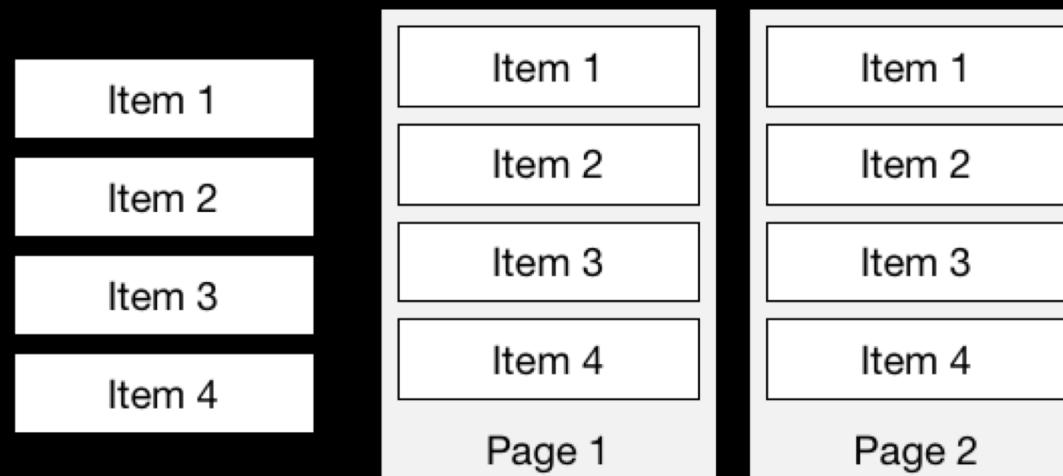


Summary

Use navigation-mode for hands-free

- navigation-mode tells Bixby to read out the results and allow user to navigate a list
- Pagination and item selection comes as part of the system
- Will turn on the mic after dialog is finished to help drive the conversation

Some examples of navigation-modes



Read one

Read many

Hands-on vs. Hands-free

We'll go over these concepts in detail

	Hands-on	Hands-free
Woken by	Bixby Button	Wake phrase (Hi Bixby)
\$handsFree variable	false	true
Mic turns on for user response	no	yes
Spoken summary	no	yes

Overview of Sessions

Tuesday

Bixby 101: Getting Started

12:30PM – Room 210A

Game Changing Voice Assistant: Industry Perspectives (Panel)

1:30PM – Room 210D

Debugging and Testing Capsules in Bixby Developer Studio

3:30PM – Room 210A

Natural Language Training: Best Practices for Bixby

4:30PM – Room 210A

Bixby Views and User Experience Best Practices

5:30PM – Room 210A

Wednesday

Building Bixby Conversational Experiences for Hands-Free Use

12:30PM – Room 210A

New Bixby Devices: No Longer Limited to Phones (Demo)

1:30PM – Room 210D

Bixby Developer Center and Marketplace

2:30PM – Room 210A

Bixby Building Blocks

3:30PM – Room 210A