

ORACLE

# Building better action menus

---

# Agenda

---

- 1 About action menus
- 2 Building a basic action menu declaratively
- 3 Building an entity-based action menu
- 4 Best practices

# Agenda

---

- 1 **About action menus**
- 2 Building a basic action menu declaratively
- 3 Building an entity-based action menu
- 4 Best practices

# About action menus

Like value lists, just for navigation

Each choice has a label and an associated action

- Action is mapped to a next dialog flow state using a transition

Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?

Book holiday

Cancel booking

View catalog

Book holiday

Consider it done. I've sent you a confirmation to your email address. Enjoy your holidays.

---

”

User errors that originate at design time are pilot errors

---

Assuming users always  
press buttons

Not handling user choices  
that are not in the list

Thinking that users can  
guess keywords

---

# About action menus

Like value lists, just for navigation

Each choice has a label and an associated action

- Action is mapped to a next dialog flow state using a transition

Action menus don't need to understand every user message, but shouldn't fail if they don't

Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?

Book holiday

Cancel booking

View catalog

Yes, please

No, cancel

Catalog, please

Sure, go ahead

Book

Cancel

Nope

Book holiday

Yes

Show me the catalog

View the catalog

Catalog

Show me the catalog

Ok

2

Cancel booking

View catalog



# What you should consider

---

Users are users and behave like users

- Users are not robotic but conversational
  - Not good in using single keywords
  - Mess up with case sensitivity
  - Send ambiguous messages

Menus should guide users and not confuse them

- Help users to make the right choice
- Be forgiving
  - Allow surrounding text with keywords
  - Handle user input case insensitive

Expect users using speech



# Agenda

---





- 1 About action menus
- 2 **Building a basic action menu declaratively**
- 3 Building an entity-based action menu
- 4 Best practices



# Creating a basic action menu in Visual Flow Designer

- ...
- Add state
- Make start state
- Delete

### Select Template

-  **Send Message**  
Send a simple message
-  **Ask Question**  
Ask a question
-  **Resolve Composite Bag**  
Resolve a composite bag entity
-  **User Messaging** >  
Send messages to users

### < User Messaging

- Resolve Entities** >  
Resolve system and custom entities
- Create Text Menu** >  
Display a simple menu
- Display Multimedia Messages** >  
Display text and multimedia messages

### < Create Text Menu

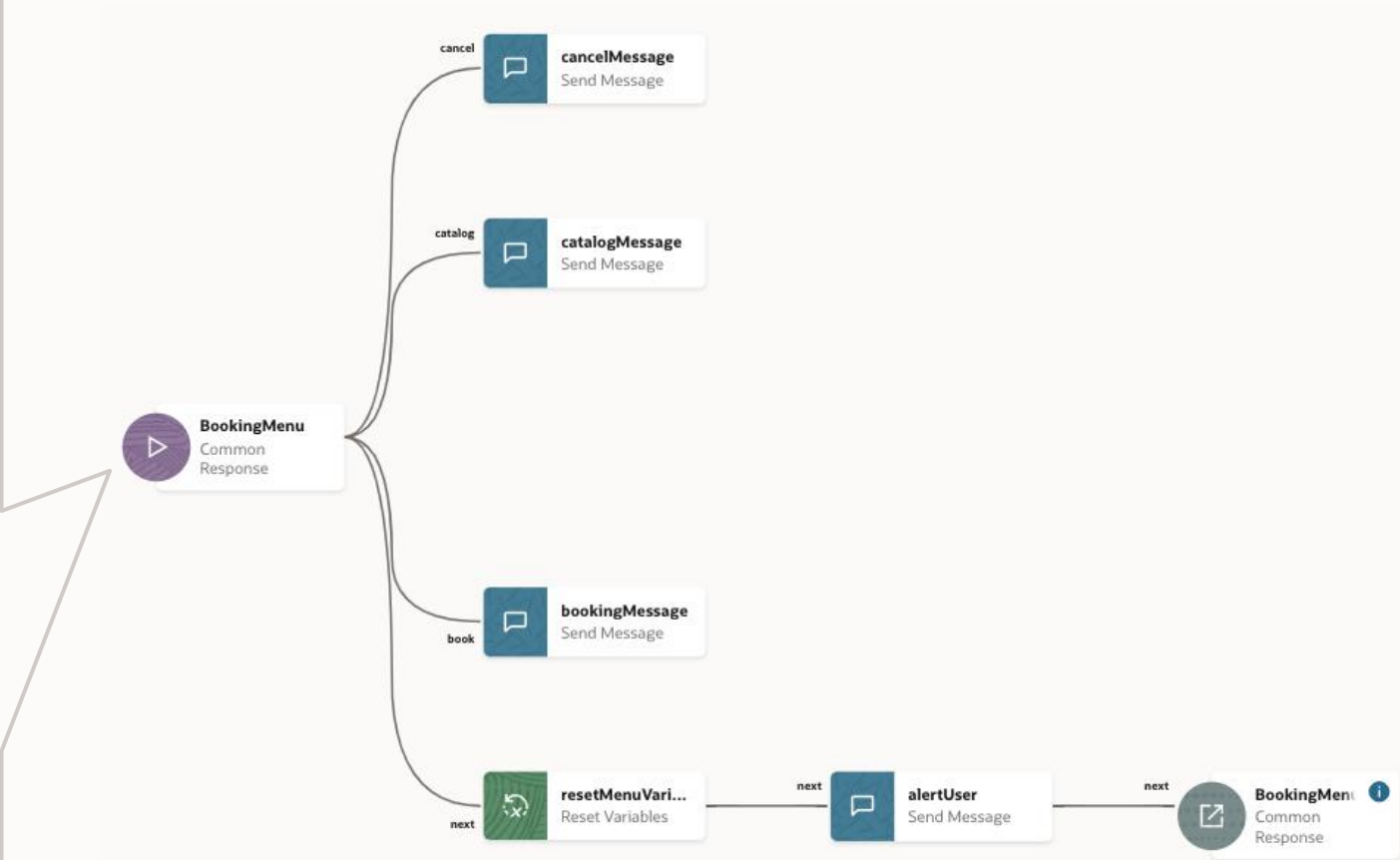
- Create Action Menu**  
Display a menu with action buttons
- Create Cards Menu**  
Create a menu using cards



# Menu definition and transitions

```
1 responseItems:  
2   - text: Seems you are ready for a vaca  
10, 2023. Do you want me to go ahead and  
3   type: text  
4   actions:  
5     - payload:  
6       variables:  
7         menuAction: book  
8         action: book  
9         label: Book holiday  
10        type: postback  
11        keyword: "1,book"  
12     - payload:  
13       variables:  
14         menuAction: cancel  
15         action: cancel  
16         label: Cancel booking  
17         type: postback  
18         keyword: "2, cancel"  
19     - payload:  
20       variables:  
21         menuAction: catalog  
22         action: catalog  
23         label: View catalog  
24         type: postback  
25         keyword: "3,view,catalog"
```

1 Enter valid YAML.



# What does work well

List rendered using buttons

Keyword property allow synonyms to be used

- The keyword property can reference a resource bundle with keywords separated by commas to support multilingual digital assistants

The screenshot displays a chatbot conversation with three distinct messages and their corresponding user responses:

- Message 1:** A dark grey bubble containing the text: "Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?". Below it is a white button list with three options: "Book holiday", "Cancel booking", and "View catalog".
- User Response 1:** A white bubble with the text "Book holiday".
- Message 2:** A dark grey bubble containing the text: "Holiday booked".
- Message 3:** A dark grey bubble containing the text: "Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?". Below it is a white button list with three options: "Book holiday", "Cancel booking", and "View catalog".
- User Response 2:** A white bubble with the text "2".
- Message 4:** A dark grey bubble containing the text: "Booking cancelled".

Small circular icons with arrows are visible to the right of each message bubble, indicating a sequence or refresh action.

# What does not work so well

Synonyms must be used as defined using the keyword property. Any use within a sentence fails

Because action menu is not validated, the navigation continues to the next dialog flow state

- If you set next transition to point to the action menu state without resetting the variable, the conversation enters into an infinite loop

Non-sequitur routing, which is the digital assistant's ability to interrupt a conversation for another, does not work with this setup

- A validation marked as failed is required

The screenshot illustrates a chatbot conversation with three distinct states:

- State 1:** A dark message bubble from the bot asks, "Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?". Below it is a white action menu with three buttons: "Book holiday", "Cancel booking", and "View catalog". A user icon is visible to the left of the menu.
- State 2:** A white message bubble from the user says, "yes, please book". A bot icon is visible to the right of the bubble.
- State 3:** A dark message bubble from the bot says, "I am sorry, it appears I did not understand what you wanted me to do. Please select from the menu". A bot icon is visible to the left of the bubble.
- State 4:** A dark message bubble from the bot repeats the question from State 1: "Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?". Below it is the same white action menu as in State 1. A bot icon is visible to the left of the menu.

Small circular arrows on the right side of each state indicate the flow of the conversation.

# Agenda

---

- 1 About action menus
- 2 Building a basic action menu declaratively
- 3 **Building an entity-based action menu**
- 4 Best practices

---

”

Define action menus as value list entities

---

Alternating prompts

User input disambiguation

Use synonyms as keywords

User input error handling

Non-sequitur navigation

---

# How to create entity-based action menus: The value list entity

Create a value-list entity

- Define value as the action to trigger
- Use synonyms as keywords

Define prompts and messages

- Error, disambiguation, prompt
- Use resource bundles

Optional: define list values as metadata and use resource bundle entries for each value

- Allow you to determine order in which items are displayed

Important: don't associate the entity with an intent

## list.bookingActionMenu

### General Information

Name

list.bookingActionMenu

Description

displays and validates the booking action menu

### Configuration

Type

Value list

#### English <sup>Ⓢ</sup>

+ Value

Sort By  
Primary Language Value

Value (Primary Language)

Synonyms

bookMenu.1.book

**book , yes , sure , do it , go for it , doit**

bookMenu.2.cancelBooking

**cancel , no , nope , don't , dont , quit**

bookMenu.3.catalog

**catalog**

# Resource bundles

Bundles defined for all messages and prompts

List values defined as bundles if value cannot be displayed as an action button label

- In case of artificial value used like in previous screen shot

## Resource Bundles

User-Defined   Intents   Q&A   Configuration

### Keys (6)

+ Add Key

- bookMenu.1.book**
- bookMenu.2.cancelBooking
- bookMenu.3.catalog
- list.bookingActionMenu.Disambig...
- list.bookingActionMenu.ErrorMes...
- list.bookingActionMenu.prompt1

### bookMenu.1.book

+ Add Language

Language	Message
default	Book Holiday





# How to create entity-based action menus: The dialog flow state

Create a dialog flow state

- User Messaging | Resolve Entities | Resolve Declarative Entity

Edit dialog flow state properties

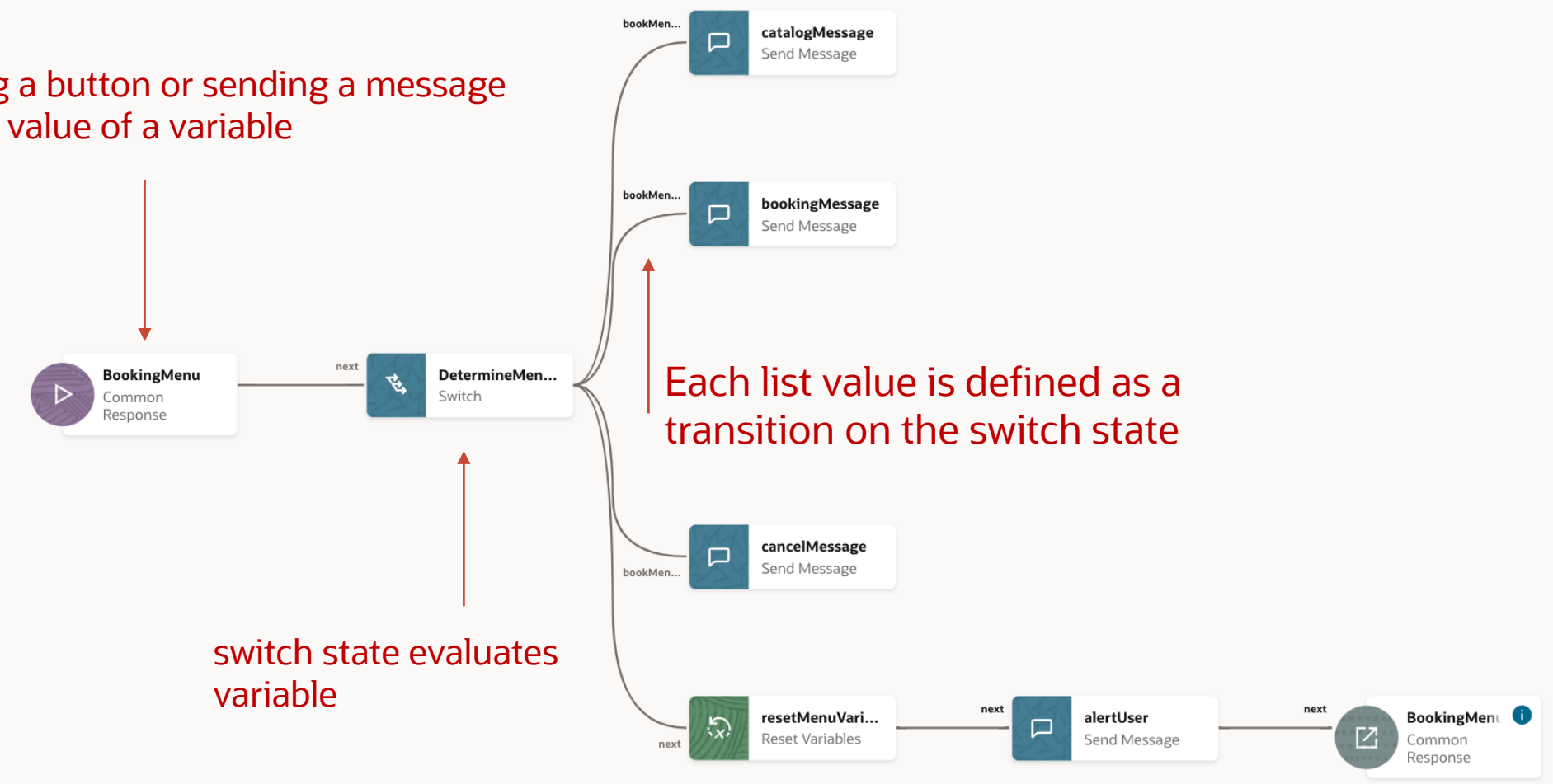
- Create flow variable pointing to list entity
- Edit response items property to use primaryLanguageValue

Edit Response Items

```
actions:  
  - iteratorVariable: enumValue  
    iteratorExpression: "${system.entityToResolve.value.enumValues}"  
    keyword: "${(enumValue?index?number+1)}"  
    payload:  
      variables:  
        "${system.entityToResolve.value.variable}": "${enumValue.primaryLanguageValue}"  
    label: "${rb(enumValue.primaryLanguageValue)}"  
    type: postback
```

# Action menu flow

pressing a button or sending a message sets the value of a variable



switch state evaluates variable

Each list value is defined as a transition on the switch state

fallback in case of a value not mapped to a state



# Entity-based action menus at runtime

Action menu displays as before

Value list entity value determines order in which action buttons show in the list, e.g.

- bookMenu.1.book
- bookMenu.2.cancelBooking

Synonyms make keywords and work even if used as part of a longer user message

User input is validated against values and synonyms.

- If user message does not match, then the error message defined on the value list is displayed and the user is re-prompted

The screenshot illustrates a chat interaction. At the top, a dark grey message bubble contains the text: "Seems you are ready for a vacation for two in Hawaii from July 10, 2023 to August 10, 2023. Do you want me to go ahead and book this for you?". Below this, a white action menu is displayed with three buttons: "Book Holiday", "Cancel Booking", and "View Catalog". To the right of the menu is a circular refresh icon. Below the menu, a user's response is shown in a white bubble: "yes, please book". At the bottom, a dark grey confirmation message bubble says "Holiday booked", with a circular refresh icon to its right. A small user icon is visible to the left of the action menu.

# Agenda

---

- 1 About action menus
- 2 Building a basic action menu declaratively
- 3 Building an entity-based action menu
- 4 **Best practices**

# Best practices

---

Use action menus that are resolved from value list entities or composite bag entities

- All messages should be read from resource bundles
  - Define multiple entities prompts for displaying alternating messages
  - Define error messages and disambiguation messages
- Define entity values as metadata values to control the order in which a value appears in the menu and to ensure unique values

If not using entities

- Prompt, labels and keywords from resource bundles
- In case of an invalid user input, navigate to a dialog flow state to print a user message, reset the action menu variable and redirect to the action menu



ORACLE