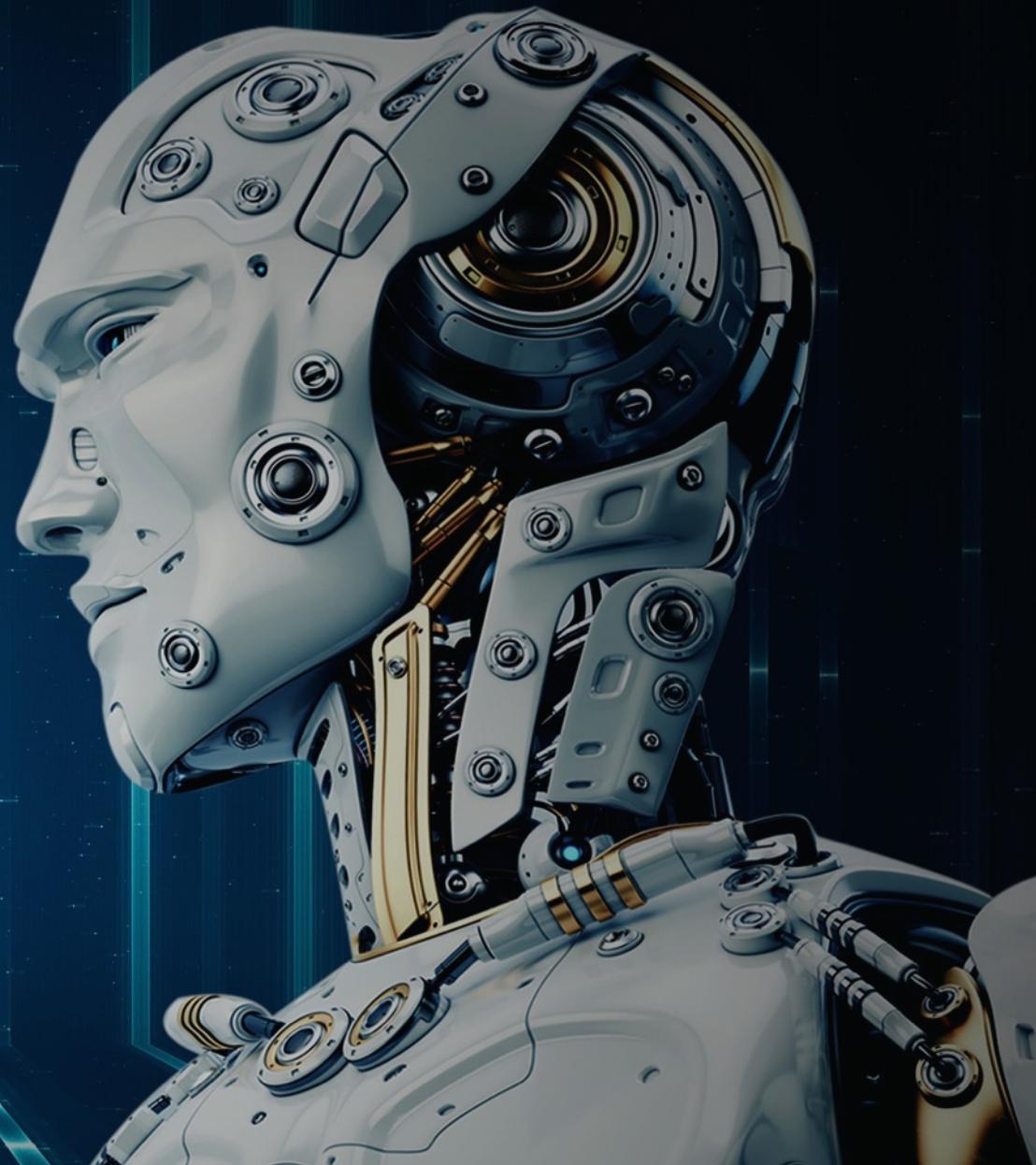


ORACLE®

Oracle Intelligent Bots Advanced Training

Bot Debugging and Error Handling

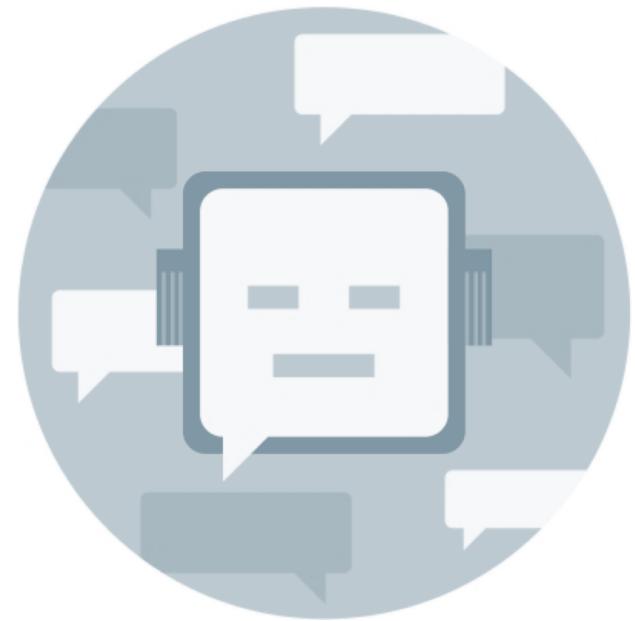


Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

The only source of knowledge
is **experience**.

- Albert Einstein



Topic Agenda

- 1 ➤ Frequent Development Mistakes
- 2 ➤ Finding Troublemakers at Design Time
- 3 ➤ Error Handling at Runtime
- 4 ➤ Custom Components
- 5 ➤ Custom Component Debugging
- 6 ➤ Dialog Flow Strategy

Debugging and Error Handling

Frequent Problems

Common Development Mistakes

- Forgetting to train the model
- Missing required component properties
- Invalid code indenting
 - Two blanks per hierarchy
- Hidden characters after copying and & pasting content
- Missing or duplicated quotes
- Use of '>' or '<' in an Apache FreeMarker expression
 - Use <, <=, >, >= instead
- Setting a semicolon (';') at the end of an expression

Errors

- Designtime
 - Misconfiguration
 - Formatting issues
- Runtime / Testing
 - Missing property values
 - Unresolved custom components
 - Component failure

How-to customize message?

Oops I'm encountering a spot of trouble.
Please try again later...

How-to get to the
bottom of the issue?

Debugging and Error Handling

Finding Troublemakers at Design Time

Approach

- Ensure model is trained if needed
 - Train All good
 - ! Train Model is not trained
- Does the intent tester work?
 - If so, then chances are that a System.Intent action references a state that doesn't exist or misses the nlpresult variable reference
- Do you use auto-translation?
 - If yes, is translation service configured?
- Exclude formatting errors
 - Use validation button
 - Use debugger



Approach

Use the **JSON bot response log**
in the embedded tester to find
exceptions



Check for Component Misconfiguration

```
        "orderearizza": null,  
        "cardsRangeStart": 0  
    },  
    "errorState": "Intent",  
    "errorMessage": "javaException:null",  
    "currentDialogState":  
"System.DefaultErrorHandler",  
    "executedStates": [  
        "System.DefaultErrorHandler"  
    ]  
}  
}  
},  
"userId": "8872725"  
}  
]
```

Test Reset

Bot Intent Q&A

I like to order pizza

Oops I'm encountering a spot of trouble.
Please try again later...

JSON

I like to order pizza 🔗 Send

The screenshot shows a bot testing interface with three tabs: Bot, Intent, and Q&A. The Bot tab is selected. On the left, there is a JSON log window containing a stack trace. A red box highlights the error state section of the log. On the right, there is a message input field with the text "I like to order pizza" and a "Send" button. Below the input field, a response message says "Oops I'm encountering a spot of trouble. Please try again later...". At the bottom, there is a "JSON" link.

Debugging and Error Handling

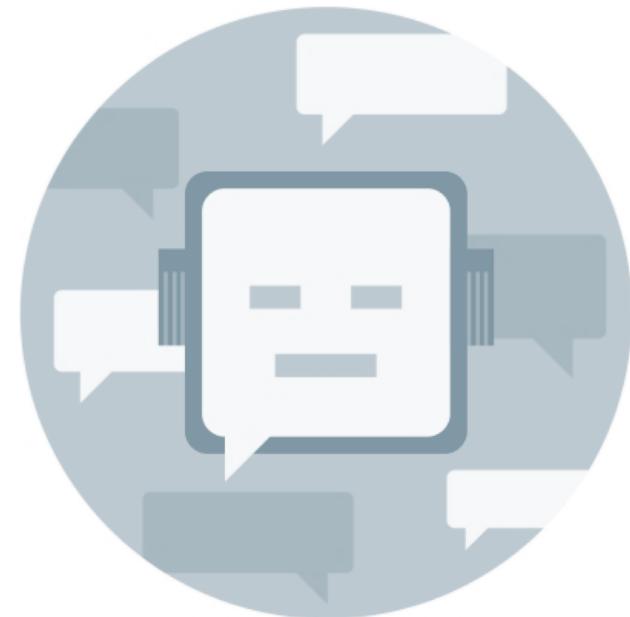
Runtime Error Handling

Error Handling in Oracle Intelligent Bots

- State Level
 - "error" transition
 - Use for errors you anticipate
- Bot level
 - Global error handler
 - Use for errors you don't anticipate
- System level
 - Last resort
 - Not user friendly

```
variables:  
...  
defaultTransitions:  
    error: "globalErrorHandler"  
  
states:  
...  
expenseState:  
    component: "expenseCreation"  
    properties:  
        amount: "${invoiceAmount.value}"  
transitions:  
    next: "attachReceipt"  
    error: "handleError"
```

**"Empty transitions" are a
recipe for failure.**



Infinite Loop

- Typically shows during design-time and testing
 - Reoccurring navigation without user input
 - Usually indicates
 - "keepTurn = true" or similar issue
 - Forgetting to reset variable states
 - Custom component problem
- Bot throws error after 100 responses sent without user interaction
 - Not configurable
 - Prevents bot from hanging itself



Your session appears to be in an infinite loop. Please try again later...

Unexpected User Actions

- Users are unpredictable
 - May enter free text where expected to select from a choice
 - May scroll up in the chat history and click an option that is no longer in focus
- **unexpectedAction**
 - Event raised when user selects out-of-scope action
- **textReceived**
 - Event raised when user provides free text input

Debugging and Error Handling

Custom Component

Custom Component Errors

- Misconfiguration
 - Missing component service access credentials
 - Usually happens when bot was imported
 - No bot response in JSON log provides a hint

JSON



```
[  
  {  
    "responses": [  
      {  
        "messagePayload": {  
          "text": "Oops I'm encountering a spot of  
trouble. Please try again later...",  
          "type": "text"  
        },  
        "userId": "3419070"  
      }  
    ]  
  }  
]
```

Custom Components Registration Errors

- http-500
 - Container could not get started
- Possible Reasons
 - You forgot to install Node dependencies
 - E.g. joi module
 - Use npm install from command line in directory containing package.json file
 - registry.js does not contain a correct component mapping
 - The component implementation file is invalid
 - Missing closing bracket
 - Missing comma in metadata JSON object
 - Component service main file does not properly reference shell.js

Custom Component Logging

- Logger
 - Custom Component SDK exposes function to access shared logger
 - conversation.logger().<log level>(' ... ');
 - Conversation.logger().info('...');
 - Log level
 - info, warn, error, fine, finest, config, trace
- Avoid over-logging
 - Choose a sensitive level for your log messages
- Use a message prefix to find messages faster in the Oracle Mobile Cloud diagnostic panel
 - E.g. component name

Setting Log Level in Oracle Mobile Cloud

The screenshot shows the Oracle Mobile Cloud Diagnostics interface. On the left, there's a sidebar with icons for Diagnostics, Settings, and Clients. The main area has tabs for Overview, Requests, Logs (which is selected), and Health. Below these are filters for API or message type (set to 5m) and Request History, APIs, and Connectors checkboxes. A modal window titled "Server settings for capturing log data" is open over the logs section. It shows a component named "Backend to Log roletester 1.0". Under "Log Level", the dropdown menu is open, showing options: FINE (selected), SEVERE, WARNING, INFO, CONFIG, FINER, and FINEST. The modal also contains a "Message" field with placeholder text "Message text" and a "Search" button. In the background, some log entries are visible, such as "The API invocation was successful" and "creating messageModel with payload: { text: 'This reply has been ...'".

Custom Components Runtime Errors

The screenshot shows the Oracle Diagnostics interface with the 'Logs' tab selected. The left sidebar lists various monitoring categories: Diagnostics, Settings, Clients, Security, APIs, Storage, and Notifications. The main area displays a log table with columns: Related, Timestamp, Message Type, Call, and Message. The 'Message' column contains error details, with one specific entry highlighted by a red box: 'TypeError: JSON.Stringify is not a function'. This indicates a runtime error in a custom component named 'bot_cc_examples'.

Related	Timestamp	Message Type	Call	Message
	04/07/2018 15:39:29.778 +02:00	SEVERE	orclBotsUtils 1.0 > custom/bot_cc_examples 1.0 >	
	04/07/2018 15:39:29.766 +02:00	SEVERE	orclBotsUtils 1.0 > custom/bot_cc_examples 1.0	TypeError: JSON.Stringify is not a function
	04/07/2018 15:39:29.765 +02:00	SEVERE	orclBotsUtils 1.0 > custom/bot_cc_examples 1.0	Error in component, details={}
	04/07/2018 15:39:29.765 +02:00	INFO	orclBotsUtils 1.0 > custom/bot_cc_examples 1.0	valid messageModel
	04/07/2018 15:39:29.761 +02:00	INFO	orclBotsUtils 1.0 > custom/bot_cc_examples 1.0	creating messageModel with payload: { text: 'Hello'

Custom Components Registration Errors

The screenshot shows the Oracle Diagnostics interface. On the left, there's a sidebar with icons for Diagnostics, Settings, Clients, Security, APIs, Storage, and Notifications. The main area has tabs for Overview, Requests, Logs (which is selected), and Health. In the Logs tab, there's a filter section and a timestamp range from 04/07/18 13:24 to 05/07/18 13. Below this is a Request History section with checkboxes for APIs and Connectors. A table lists log entries:

Related	Timestamp	Message
	04/07/2018 15:39:29.778 +02:00	SEVERE
	04/07/2018 15:39:29.766 +02:00	SEVERE
	04/07/2018 15:39:29.765 +02:00	SEVERE
	04/07/2018 15:39:29.765 +02:00	INFO
	04/07/2018 15:39:29.761 +02:00	INFO

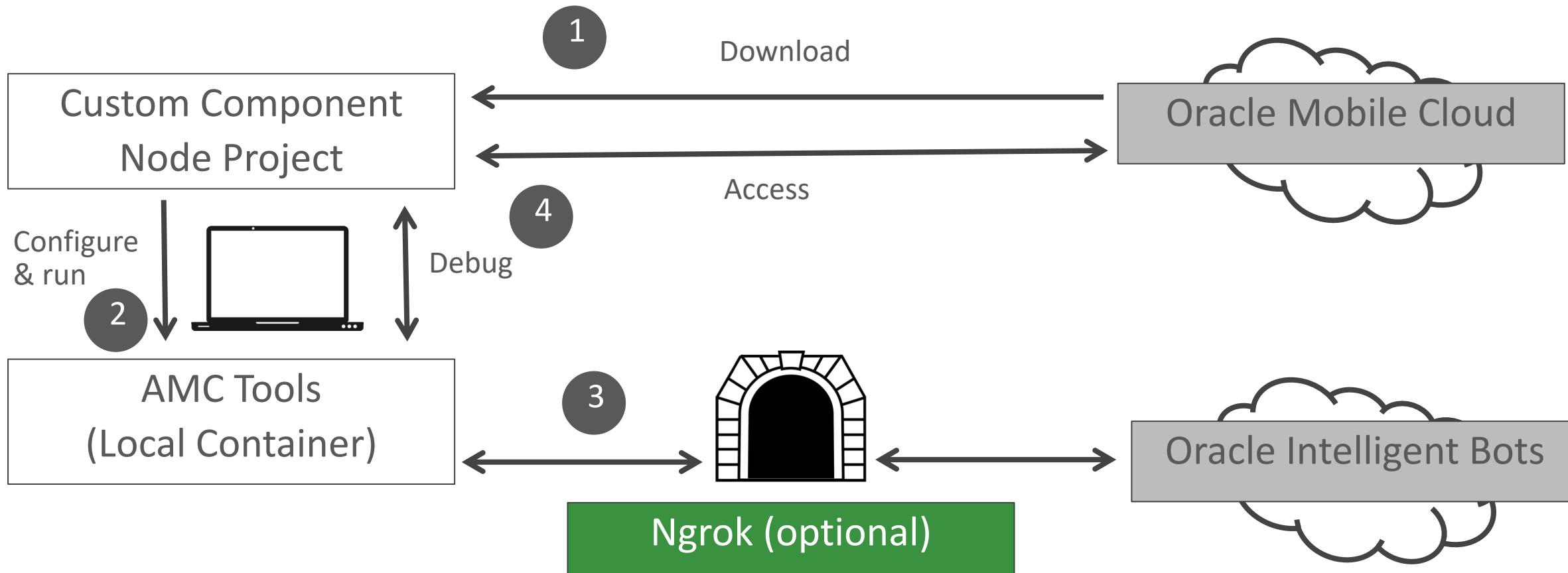
A modal window titled "Message Detail" is open, showing "Message 19 of 56". It has an "Overview" section with the text "Custom Code:custom/bot_cc_examples 1.0 - SEVERE" and "bots_OMCe_service_template 1.0.0". The "Message Text" section contains a stack trace:

```
TypeError: JSON.Stringify is not a function
at Object.invoke (/u01/app/oracle/mobile/mobile_ccc/custom_code_modules/cb65cb8b-d321-4084-9dd6-a56b20386435_6/sdk-mce/examples/say_hello.js:20:52)
at Object.invokeComponentByName (/u01/app/oracle/mobile/mobile_ccc/custom_code_modules/cb65cb8b-d321-4084-9dd6-a56b20386435_6/sdk-mce/shell.js:84:19)
at /u01/app/oracle/mobile/mobile_ccc/custom_code_modules/cb65cb8b-d321-4084-9dd6-a56b20386435_6/sdk-mce/bots.js:36:11
at Layer.handle [as handle_request] (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/layer.js:95:5)
at next (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/route.js:131:13)
at Route.dispatch (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/route.js:112:3)
at Layer.handle [as handle_request] (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/layer.js:95:5)
at /u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:277:22
at param (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:349:14)
at param (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:365:14)
at Function.process_params (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:410:3)
at next (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:271:10)
at mcsSdk (/u01/app/oracle/mobile/mobile_ccc/mcs-node-router/mcs-node-server.js:908:9)
at Layer.handle [as handle_request] (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/layer.js:95:5)
at trim_prefix (/u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:312:13)
at /u01/app/oracle/mobile/mobile_ccc/node-configurations/6.10/node_modules/express/lib/router/index.js:280:7 ['TypeError: JSON.Stringify is not a function']
```

Debugging and Error Handling

Custom Component Debugging

AMC / Bot Local Development & Debugging Architecture



Debugging with JavaScript IDE

The screenshot shows the Oracle Mobile Studio interface with the following details:

- Top Bar:** Code, File, Edit, Selection, View, Go, Debug, Tasks, Window, Help.
- Left Sidebar:** Includes icons for file operations (New, Open, Save, Find, Replace, Cut, Copy, Paste, Undo, Redo) and a search bar.
- Variables Panel:** Shows local variables for the current component. A red box highlights the variable `order`, which is currently undefined.
- Call Stack:** Shows the stack trace for the current invoke operation, with the top entry being `invoke SubmitOrder.js 97:9`.
- Code Editor:** Displays two files: `OrderSummary.js` and `SubmitOrder.js`. The `SubmitOrder.js` file contains the following code:

```
const deliveryType = conversation.properties().deliveryType;
const deliveryCost = conversation.properties().deliveryCost;
const recipientName = conversation.properties().recipientName;
const recipientAddress = conversation.properties().recipientAddress;
const recipientMobile = conversation.properties().recipientMobile;

//optional fields
const discountPercent = conversation.properties().discountPercent ? conversation.properties().discountPercent : 0;
const cardSenderName = conversation.properties().cardSenderName ? conversation.properties().cardSenderName : "Frank";
const cardMessage = conversation.properties().cardMessage ? conversation.properties().cardMessage : "Happy Birthday";

conversation.logger().fine('advt.24hrs.flowers.orders.SubmitOrder: Oracle Mobile SDK found? ' + conversation.oracleMobile);

if (conversation.oracleMobile != null) {
    //generate random order status. The options are:
    //delivered = order complete, active = in delivery, waiting = awaiting delivery, rejected = not delivered
    var orderStatusOptions = ['delivered', 'active', 'queued', 'none', 'rejected'];
    var orderStatus = orderStatusOptions[Math.floor(Math.random()*orderStatusOptions.length)];
    conversation.logger().fine('advt.24hrs.flowers.orders.SubmitOrder: Order status set to ' + orderStatus);
    //create order JSON
    var order = {
        "order": {
            "orderId": orderId,
            "orderStatus": orderStatus,
            "productName": productName,
            "pricePerUnit": pricePerUnit,
            "orderQuantity": orderQuantity,
            "discountPercent": discountPercent
        }
    };
}
```

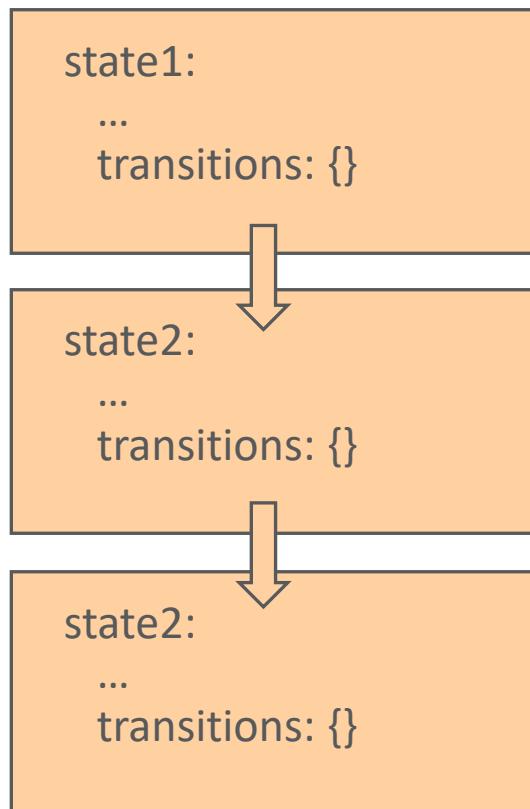
A red box highlights the condition `if (conversation.oracleMobile != null)`.

Debugging and Error Handling

Dialog Flow Strategy

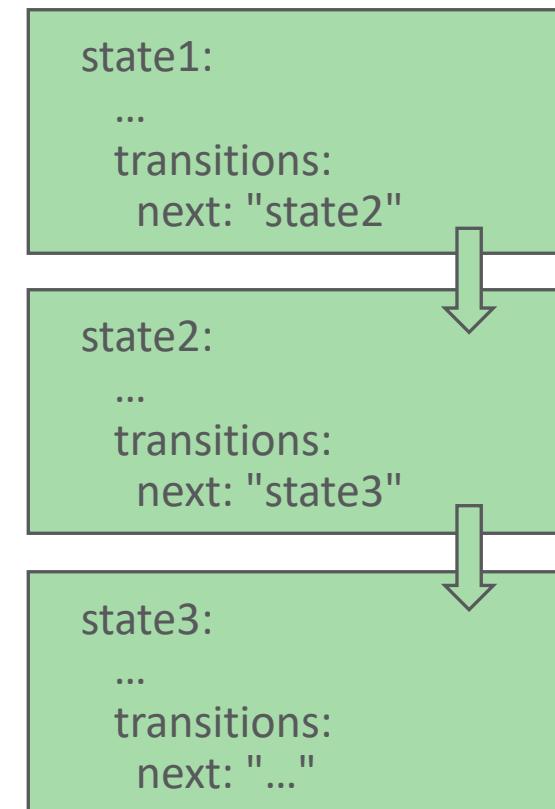
Empty Navigation vs. Next Transition

Empty Navigation



What's the difference?

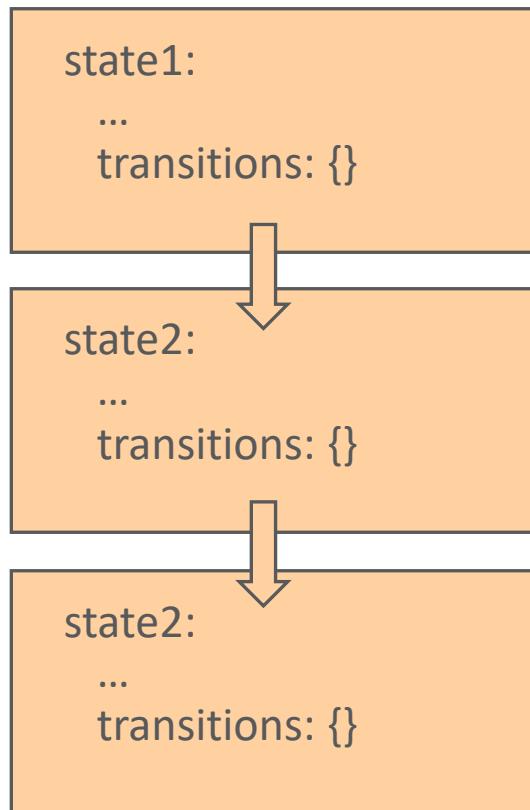
Next Transition



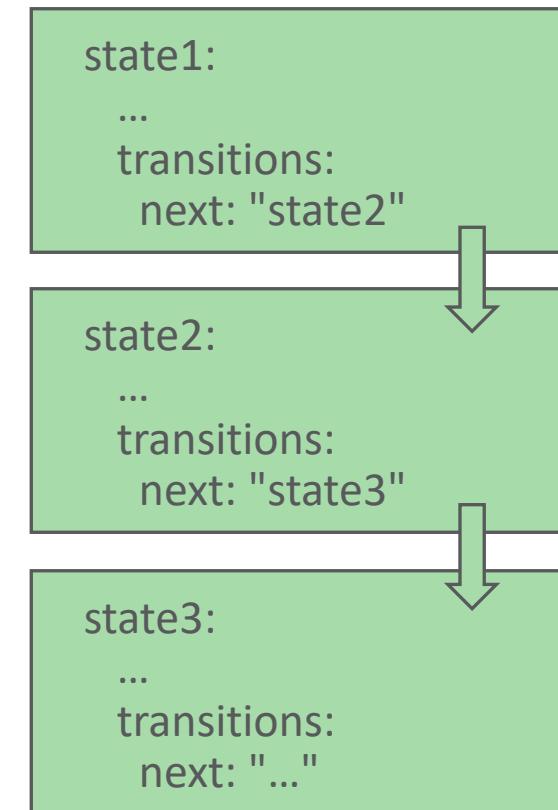
Empty Navigation vs. Next Transition

```
defaultTransitions:  
  next: "NavigationAlert"  
  error: "GlobalErrorHandler"  
actions:  
  unexpectedAction: "unexpAction"
```

Empty Navigation



Next Transition



What's the difference now?

What's the cost?

Discipline

And how about this?

Global error handler



Define default transition
for "next" navigation

Test

Reset

Bot

Intent

Q&A

Hi

Hi there! What would you like me to echo back?

Hello!

Ola!

Vannakam!

Namaste!

Namaste!

Your bot conversation uses an empty transition. Please correct. See the JSON log for help



```
context:  
variables:  
  iResult: "string"  
  greeting: "string"  
defaultTransitions:  
  next: "alertUseOfEmptyTransition"
```

```
states:  
  askGreeting:  
    component: "System.List"  
    properties:  
      options: "Hello!, Ola!, Vannakam!, Namast  
      prompt: "Hi there! What would you like me  
      variable: "greeting"
```

```
printGreeting:  
  component: "System.Output"  
  properties:  
    text: "${greeting.value}"  
transitions:  
  return: "done"
```

```
alertUseOfEmptyTransition:  
  component: "System.Output"  
  properties:  
    text: "Your bot conversation uses  
    transitions."
```

Test

Reset

Bot

Intent

Q&A

Hi

Hi there! What would you like me to echo back?

Hello!

Ola!

Vannakam!

Namaste!

Namaste!

Your bot conversation uses an empty transition. Please correct. See the JSON log for help



JSON

```
        "currentDialogState":  
"alertUseOfEmptyTransition",  
        "executedStates": [  
            "askGreeting",  
            "alertUseOfEmptyTransition"  
        ]  
    }  
}  
},
```

Integrated Cloud Applications & Platform Services

ORACLE®