

ORACLE®

Oracle Digital Assistant

The Complete Training

Digital Assistant Skills Routing

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.

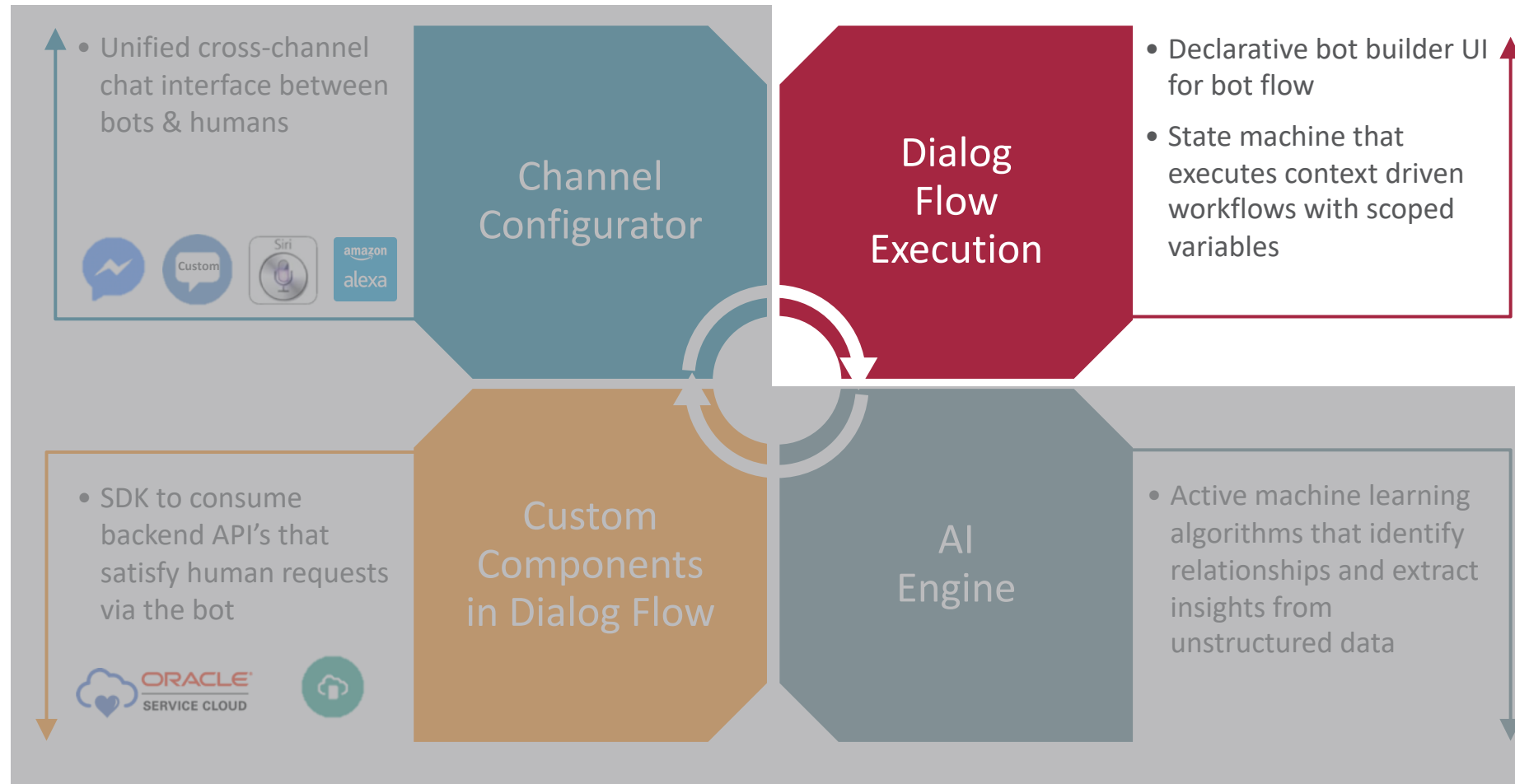
Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing
- 4 ➤ Tuning the routing model
- 5 ➤ Dealing with unresolvedIntent

Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing
- 4 ➤ Tuning the routing model
- 5 ➤ Dealing with unresolvedIntent

Intelligent chatbots: key components



What are skills?

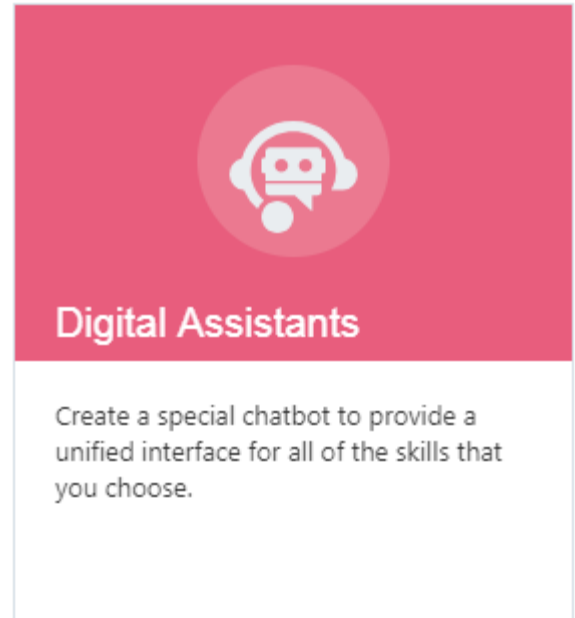
- Skills are ***individual chatbots*** designed to *fulfill specific tasks*, such as ordering food, making reservations, or changing contact information
- Each skill helps a user complete a task through a combination of text messages and simple UI elements like select lists

The benefits of skills

- Each skill can focus exclusively on its own domain
 - Improves intent classification within each skill
 - Modularizes functions and enables incremental development
 - Simplifies versioning and lifecycle management
- Dramatically simplifies dialog flow development
 - Conversational Designers need not worry about (and design for) skill disambiguation
 - Built-in Digital Assistant skills reduces code in each individual skill
- Improves non-sequitur/off topic handling
- Enables segmented authorization

What is a digital assistant?

- An AI-driven collection of skills
- Advantages
 - Assemble based on developed skills or from skill store
 - Automatically matches user input to most appropriate skill
 - Greets the user on access
 - Upon request, lists what skills and use cases it supports
 - Handles interruptions to flows
 - Handles disambiguation
 - Explicit invocation
 - Exit and help requests



What is routing?

- A key aspect of Oracle Digital Assistant
 - Enables richer, more human-like interactions
- A “conversational air traffic control”
 - Controls the overall “flow” of a conversation between and within skills
 - Necessary for the orchestration of skillbots



Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing
- 4 ➤ Tuning the routing model
- 5 ➤ Dealing with unresolvedIntent

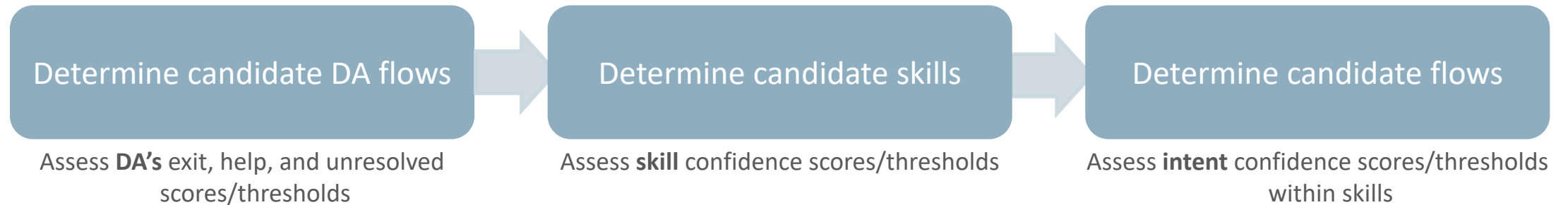
Routing terminology

- Implicit routing
 - Routing based on content in input (user utterance)
 - DA routing rules decide which bot will handle input
 - Inputs are automatically disambiguated (via “Smart Dialogs”), if needed
- Explicit routing
 - Occurs when a skill name is explicitly stated in the user’s input
 - The dialog flow within the skill determines how the input is handled
- Candidate skills
 - Skills that have matching intents for a user input message
- Candidate flow
 - Intent in a skill bot matching the user input message
- System intent
 - Built-in DA intents (e.g., exit, help, unresolved)

The digital assistant routing model – what it does

- The Digital Assistant evaluates each input (user utterance) to determine “where it belongs” and thus decides how to respond
- The options for routing an input are:
 - To a built-in Digital Assistant intent
 - To a new skill
 - To a different intent (state) within the current skill

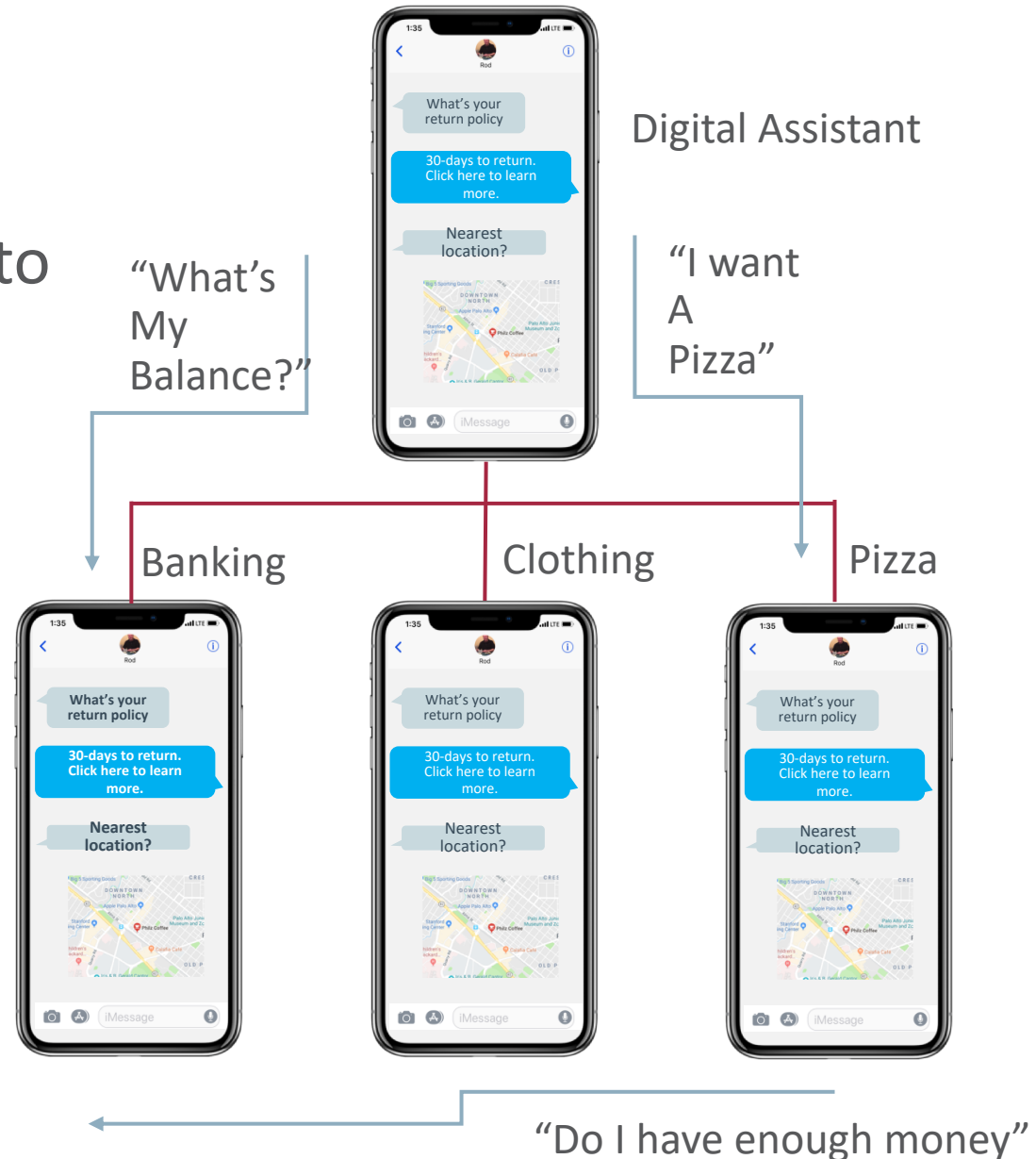
The base routing model layers



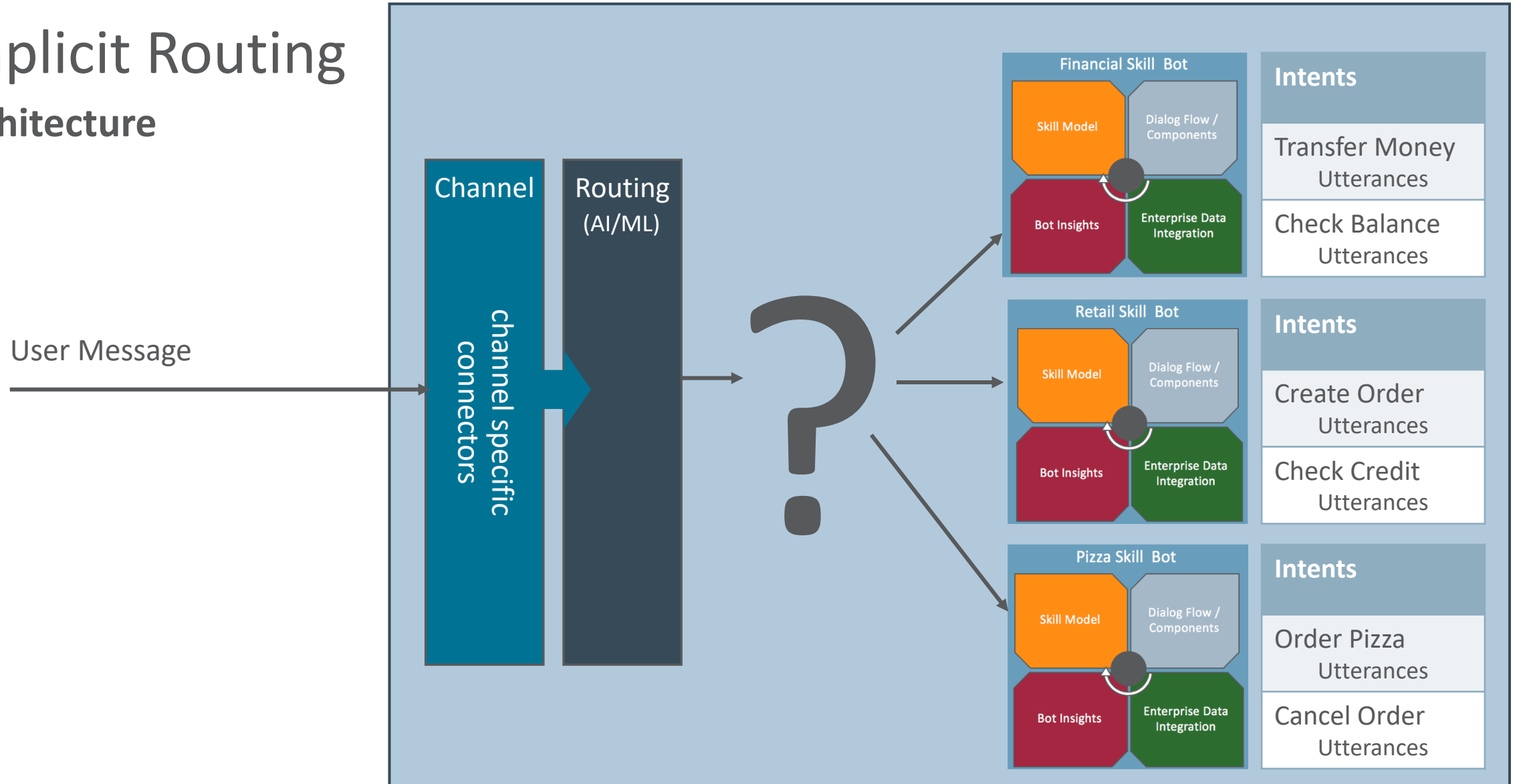
- **NOTE:** *There are special cases that impact the base routing model. (We'll cover them shortly)*

Skill routing example

- Digital Assistant routes requests to the correct skill bots



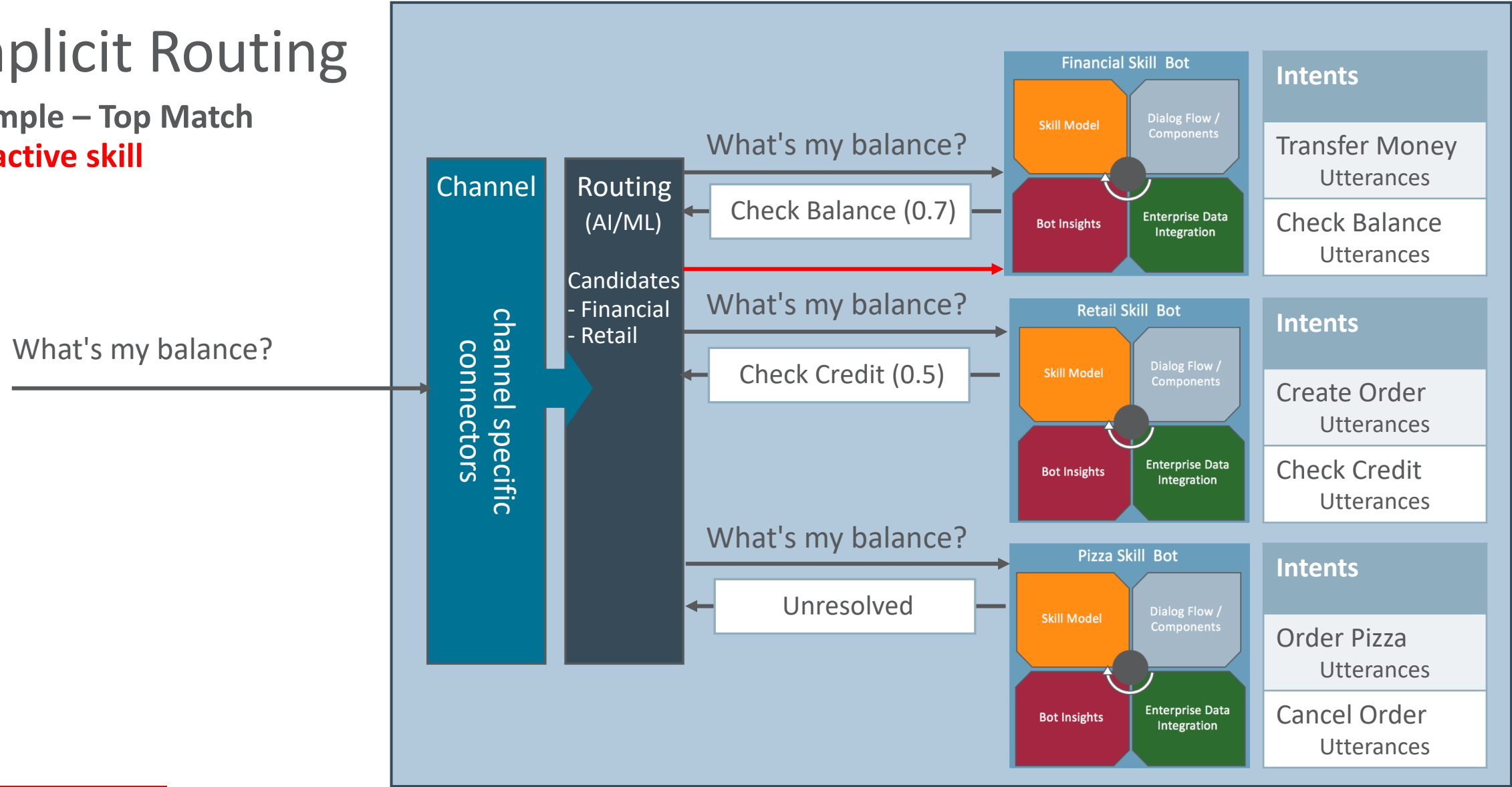
Implicit Routing Architecture



Implicit Routing

Example – Top Match

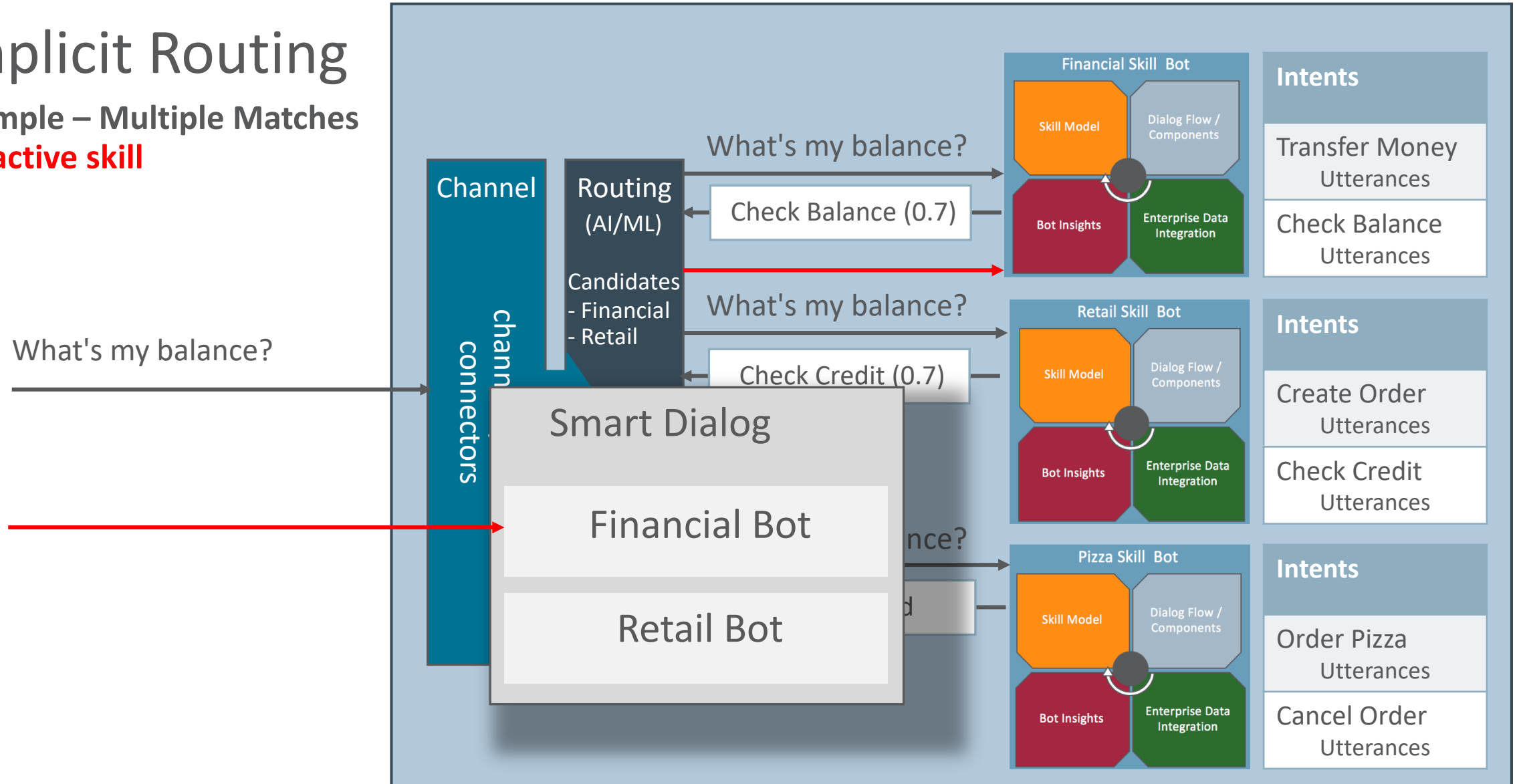
No active skill



Implicit Routing

Example – Multiple Matches

No active skill

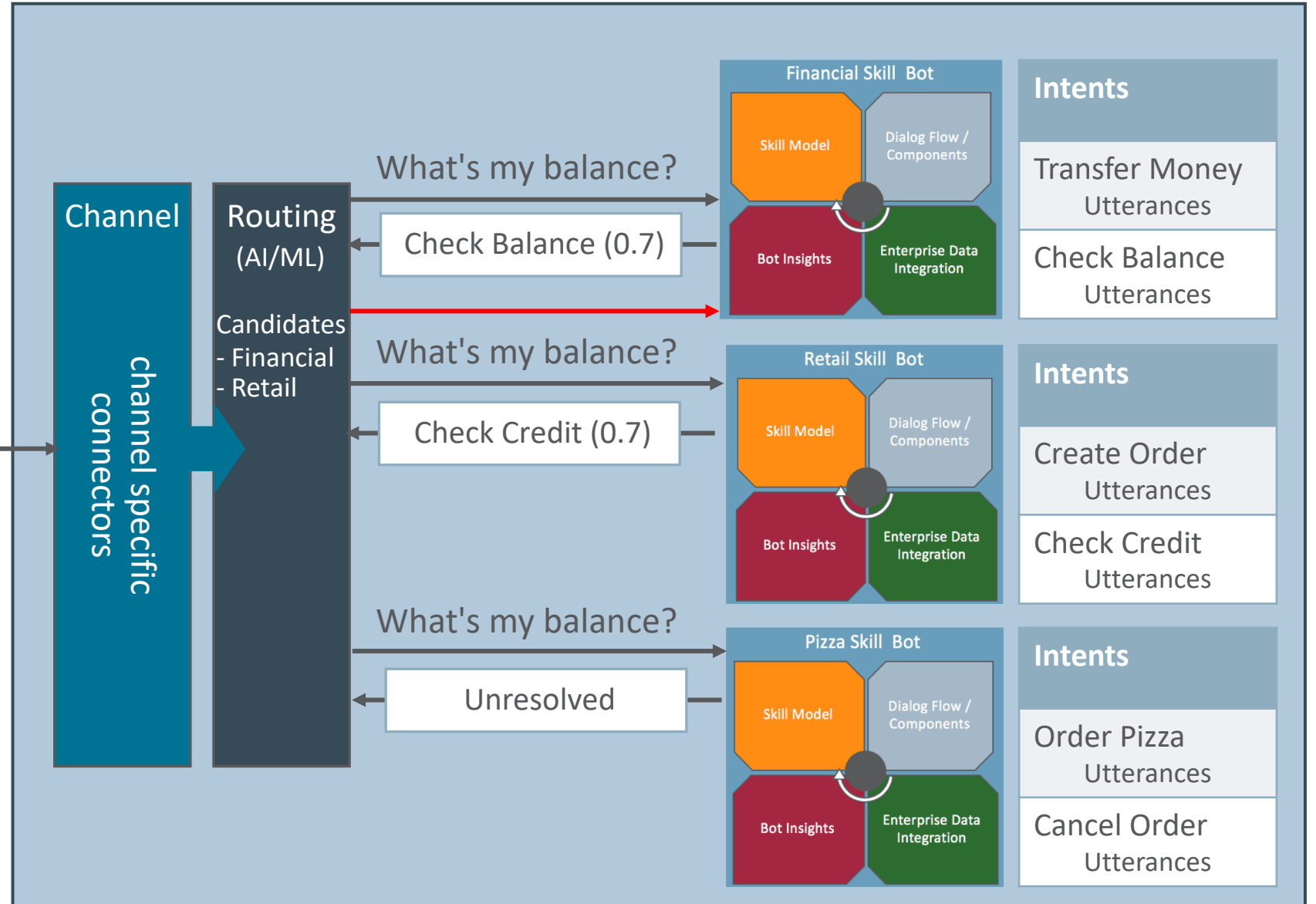


Implicit Routing

Example – Multiple Matches

Financial = active skill

What's my balance?

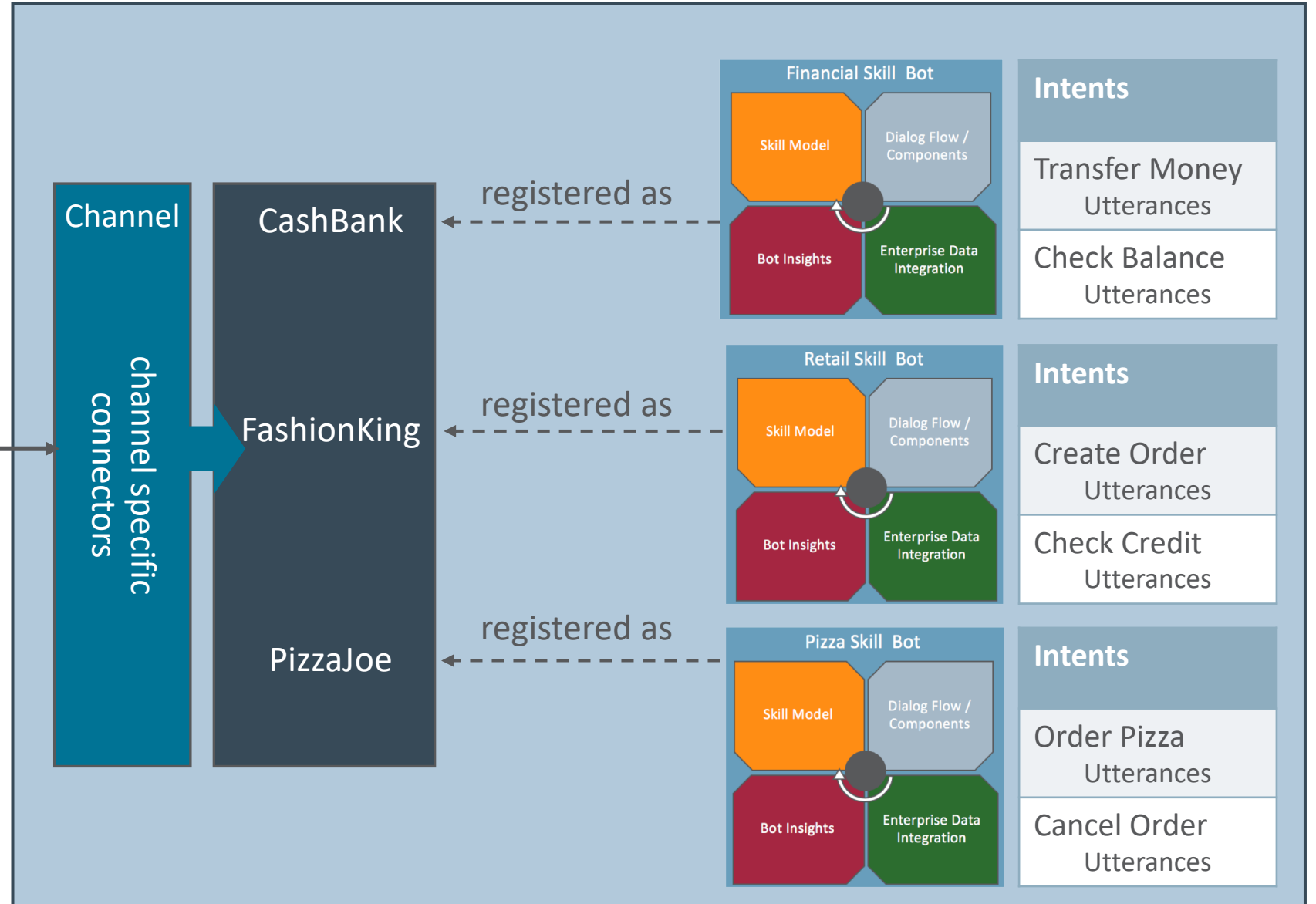


Explicit invocation patterns

- [Phrase] + skill bot name + [utterance]
 - Pizzajoe
 - Ask pizzajoe when my pizza will be delivered
 - Pizzajoe ,I want to order a pizza
- [Utterance] + skill bot name
 - Order a pizza from pizzajoe
 - Can I place an order with pizzajoe?

Explicit routing Architecture

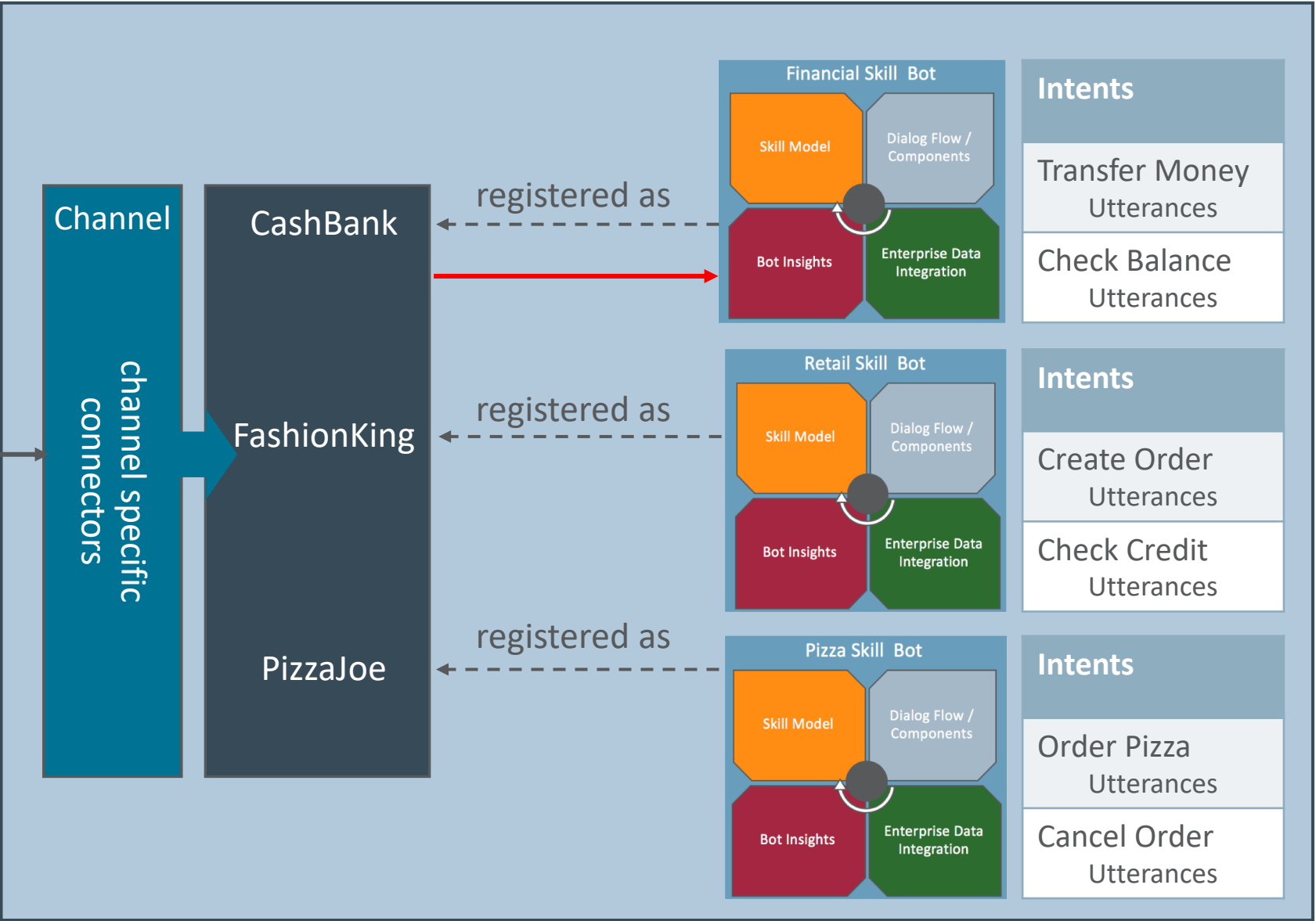
User Message
<addressing> <utterance>



Explicit routing

Example

Ask **CashBank** to check my balance

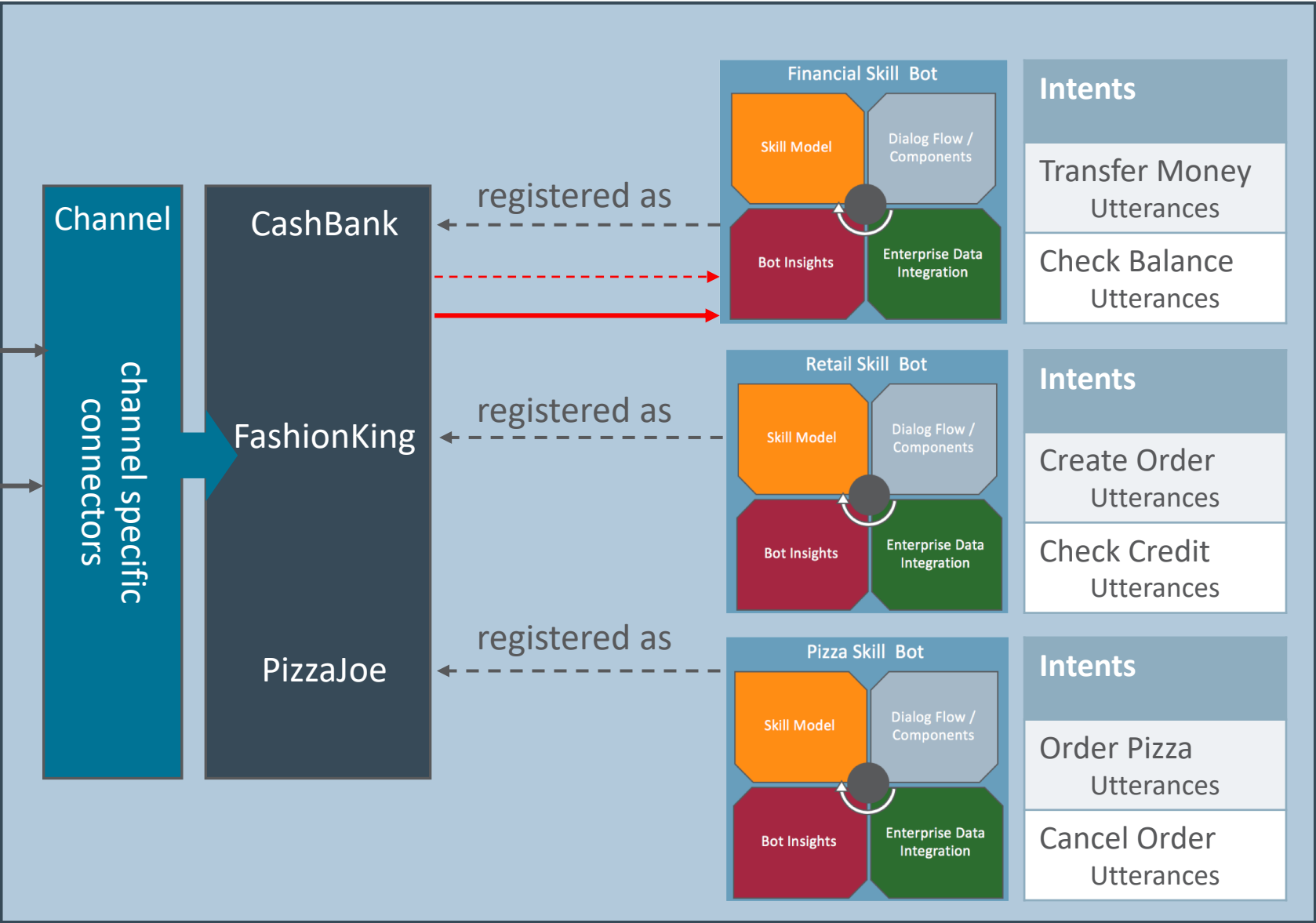


Explicit routing

Example

CashBank

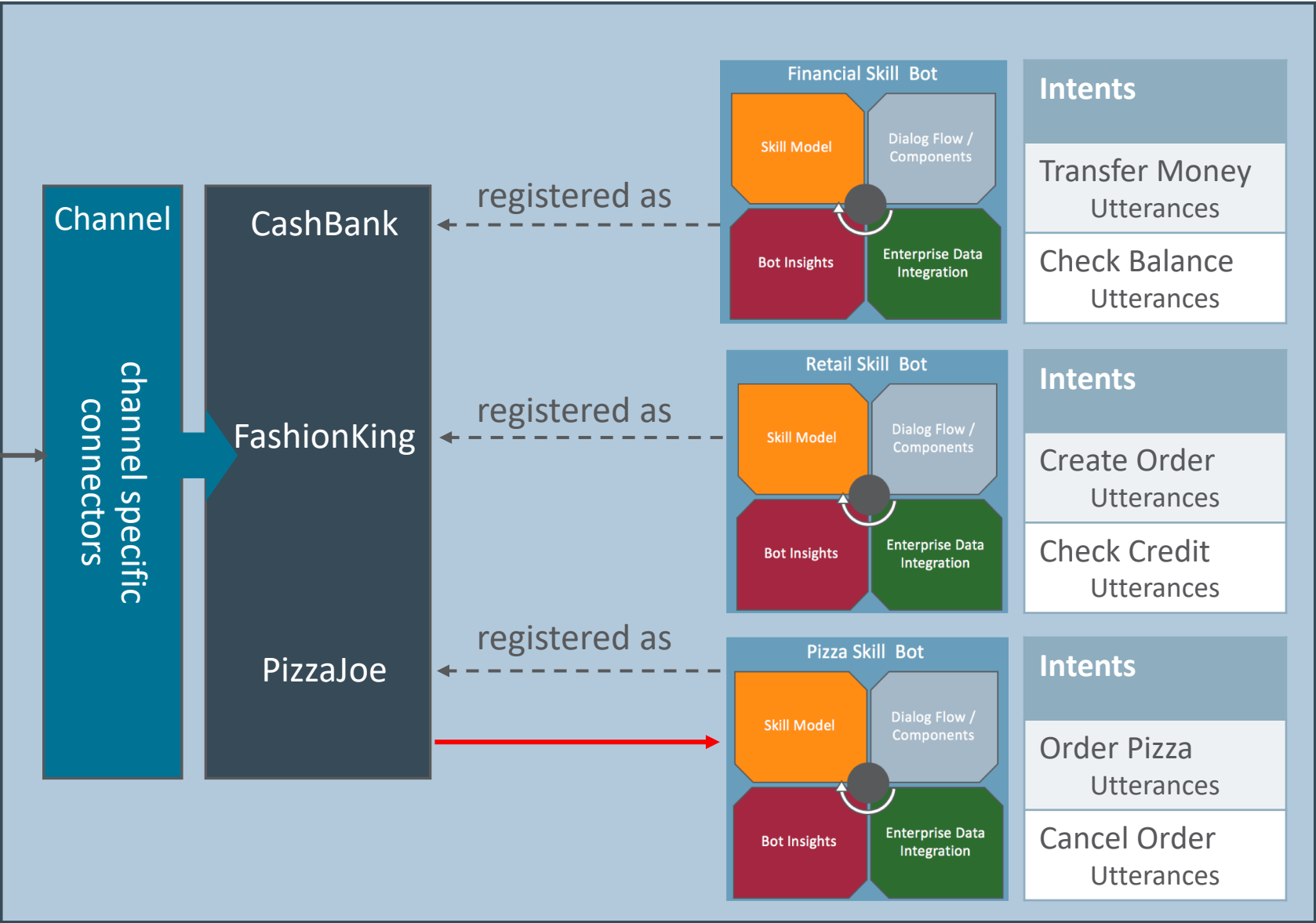
What's my balance?



Explicit routing

Example

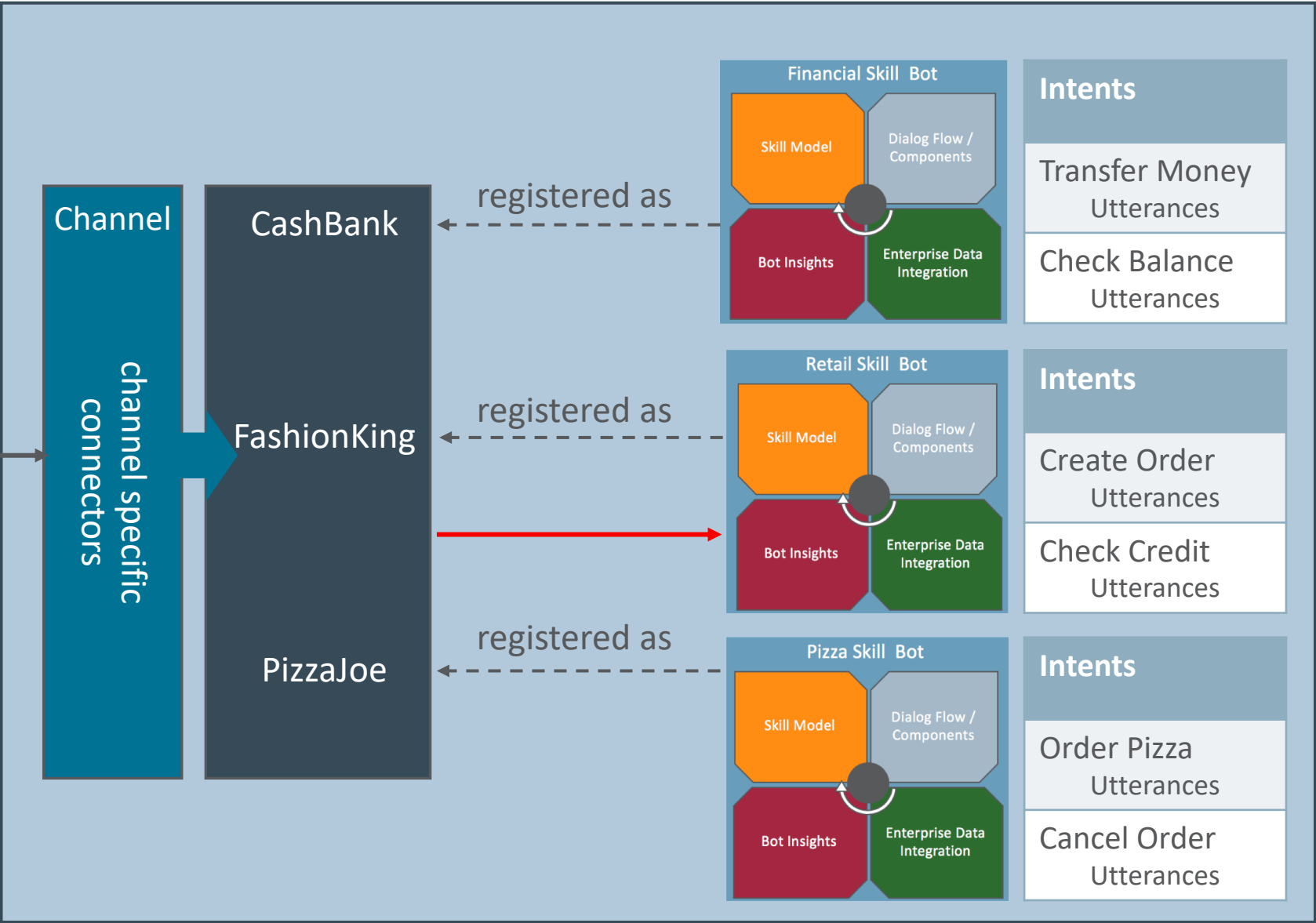
PizzaJoe, I want to order food



Explicit routing

Example

check **FashionKing**, how much money do I have?

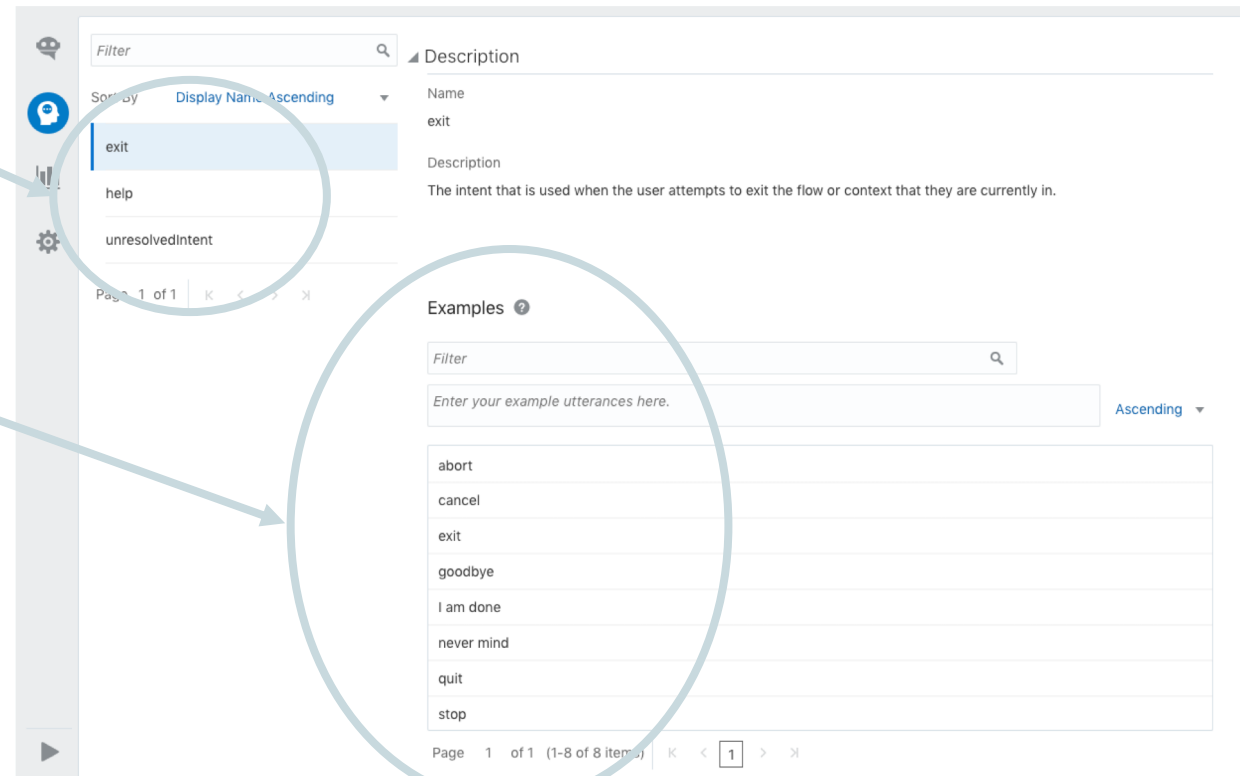


Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing**
- 4 ➤ Tuning the routing model
- 5 ➤ Dealing with unresolvedIntent

Built-in Digital Assistant intents

- Wouldn't it be good if DA could globally handle common use cases?
- The DA has *built-in intents* to control the behavior for the following:
 - Exit
 - Help
 - Unresolved
- Each intent may be trained to suit specific needs



Built-in skills – exit

- Exit state applies when the intent engine determines user asking to exit
- If in a flow, DA will ask to confirm exit (via the “Exit Prompt” setting)

exit

Do you want to exit Balances in Finance Master now?

Yes

No

Exit Prompt

Do you want to exit `${system.routingFromIntent}` in `${system.routingFromSkill}` now?

The prompt to display before exiting a flow

- If not in a flow, DA will confirm once it has exited (via the “Exit Skill Confirmation” setting)

exit

Exited Finance Master.

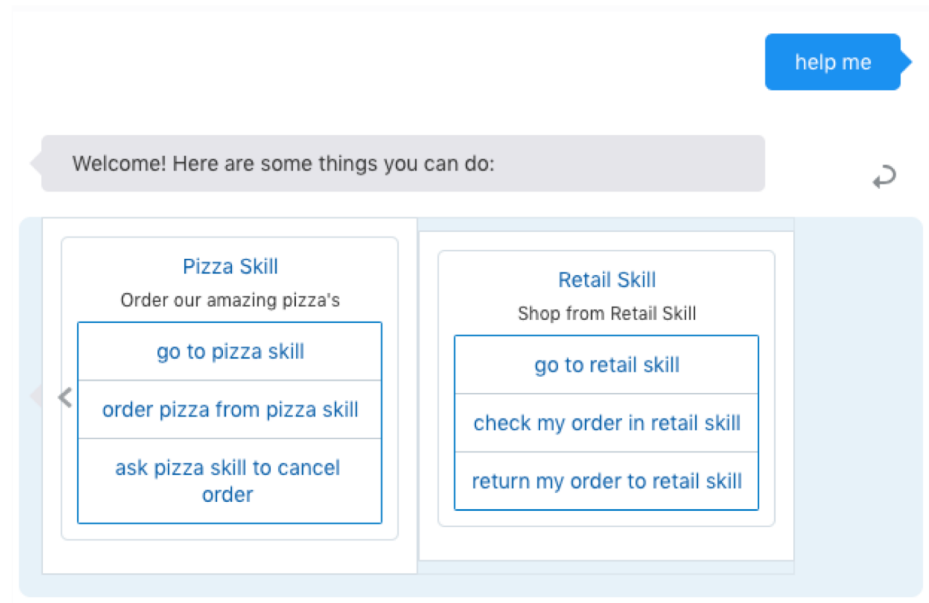
Exit Skill Confirmation

Exited `${system.routingFromSkill}`.

The confirmation to display when exiting a skill

Built-in skills – help (user not in skill)

- Help state applies when the intent engine determines user asking for help
- When not in a flow/skill, the digital assistant will DA will offer a help prompt and a carousel of available skills



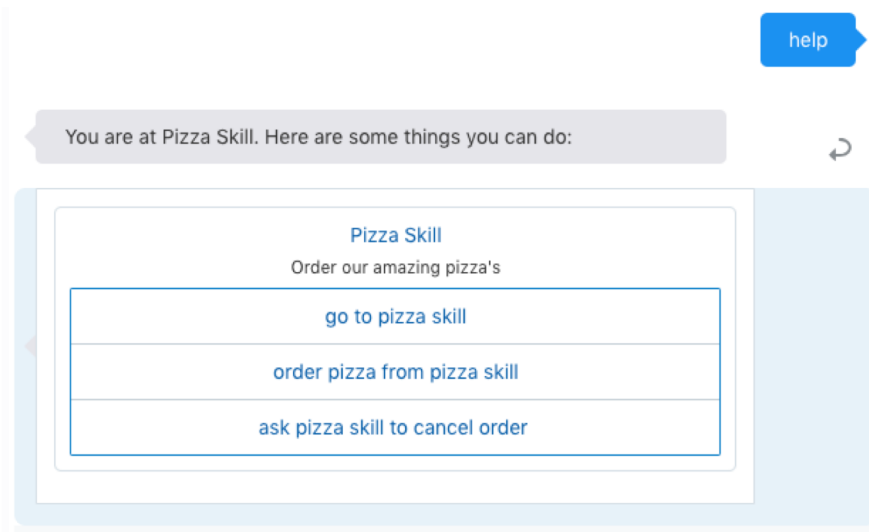
Digital Assistants Help Prompt

Welcome! Here are some things you can do:

The help message to display before help card

Built-in skills – help (user in skill)

- Help state applies when the intent engine determines user asking for help
- When in a flow/skill, the DA will first try to route to the skill's help state (as specified in the skill's configuration)
- If no help state has been defined at the skill level, the DA displays a prompt and a card



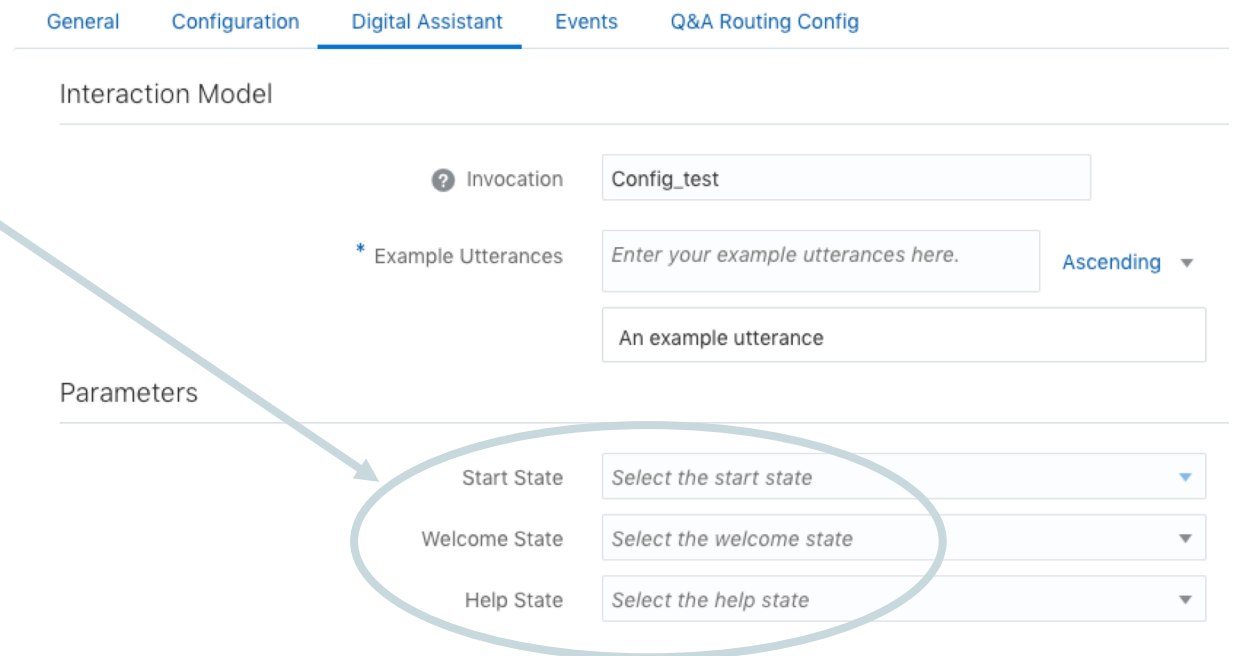
Skill Bot Help Prompt

You are at \${system.routingFromSkill}. Here are some things you can do:

The help message to display for a skill bot above its help card

Skill-level settings and DA default responses

- Individual skills may contain specific Start, Welcome, and Help states
 - Skill-level settings determine which state in a skill's dialog flow handles each state

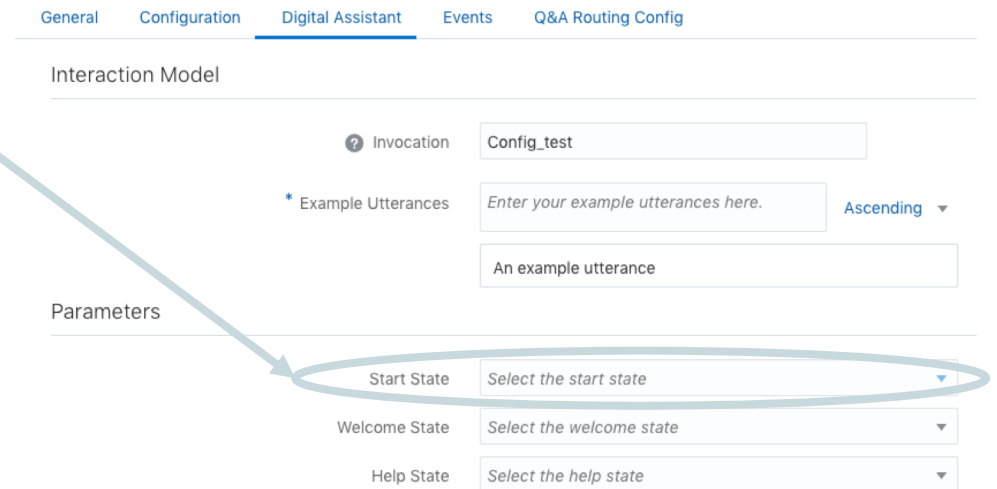


The screenshot displays the 'Digital Assistant' configuration tab. It includes sections for 'Interaction Model' and 'Parameters'. The 'Parameters' section is highlighted with a blue circle, and a blue arrow points from the text 'Skill-level settings determine which state in a skill's dialog flow handles each state' to this section.

| General | Configuration | Digital Assistant | Events | Q&A Routing Config |
|--------------------------|---------------|--------------------|--|--------------------|
| Interaction Model | | | | |
| | | Invocation | Config_test | |
| | | Example Utterances | Enter your example utterances here. Ascending ▼ | |
| | | | An example utterance | |
| Parameters | | | | |
| | | Start State | Select the start state ▼ | |
| | | Welcome State | Select the welcome state ▼ | |
| | | Help State | Select the help state ▼ | |

Default responses - start

- The Start state occurs when the intent engine determines that the user wants to start using a given skill
 - Generally when the user expresses an intent that is related to a skill
- If a start state hasn't been specified in the skill, the DA invokes the first state in the skill (typically the System.Intent component)



The screenshot displays the 'Digital Assistant' configuration tab in the Oracle Digital Assistant interface. It shows the 'Interaction Model' section with fields for 'Invocation' (set to 'Config_test'), 'Example Utterances' (with a text input and a dropdown set to 'Ascending'), and 'Parameters'. The 'Parameters' section contains three dropdown menus: 'Start State' (highlighted with a blue oval and an arrow pointing to it from the text 'typically the System.Intent component'), 'Welcome State', and 'Help State'. All three dropdown menus currently display the placeholder text 'Select the start state', 'Select the welcome state', and 'Select the help state' respectively.

Default responses - welcome

- Welcome applies when user enters the invocation name without an intent.
- If a welcome state is not defined for a skill, DA provides one automatically.
 - Default welcome is a prompt and card showing the skill's display name, one-sentence description, and a few of its sample utterances.

The screenshot displays the Oracle DA console interface for configuring a skill's welcome response. At the top, a blue button labeled "go to finance master" is visible. Below it, a grey prompt box contains the text "Welcome to Finance Master." To the right of this prompt is a circular refresh icon. The main area shows a card for the "Finance Master" skill, with the subtitle "Banking services from Fin Skill". Inside the card, there is a list of three sample utterances: "go to Finance Master", "ask Finance Master to check my balance", and "send money using Finance Master". To the right of the card, the "Skill Bot Welcome Prompt" is shown as "Welcome to \${system.routingFromSkill}." with a description "The welcome message to display for a skill bot". Below this, the "Skill Flow Selection Limit" is set to "5", with a description "Maximum skill flows to be offered for selection." and up/down arrow controls.

go to finance master

Welcome to Finance Master.

Finance Master
Banking services from Fin Skill

go to Finance Master

ask Finance Master to check my balance

send money using Finance Master

Skill Bot Welcome Prompt

Welcome to \${system.routingFromSkill}.

The welcome message to display for a skill bot

Skill Flow Selection Limit

5

Maximum skill flows to be offered for selection.

Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing
- 4 ➤ Tuning the routing model**
- 5 ➤ Dealing with unresolvedIntent

DA routing configurations overview

- Important terminology:
 - *Confidence level* is the intent engine’s “score” for utterance classification
 - Range is 0 – 1.0
 - *Threshold* is a value that’s compared to the confidence level to define next action
 - *Margin* is the difference between confidence levels
- Routing behavior is tuned by adjusting the thresholds of routing parameters

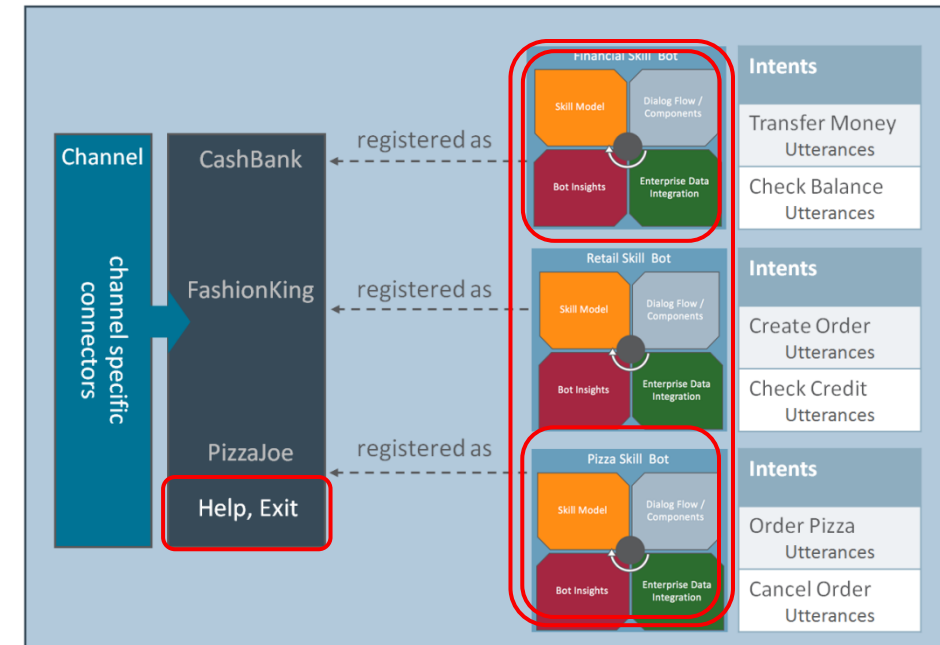
[General](#) [Configurations](#) [Events](#)

Routing Parameters

| | | | |
|---|-----|-----|--|
| Built-In System Intent Confidence Threshold | 0.6 | ▼ ▲ | Threshold for identifying built-in system intents, like help and exit. (Minimum value 0, maximum value 1) |
| Candidate Skills Confidence Threshold | 0.4 | ▼ ▲ | The minimum confidence score required to a match a candidate skill. (Minimum value 0, maximum value 1) |
| Confidence Win Margin | 0.1 | ▼ ▲ | Used to help determine which candidate skills and candidate built-in system intents are matched with user input. Only the top candidate that exceeds the confidence threshold is matched if its confidence score exceeds that of other candidates by this value or more. If other candidates that exceed the confidence threshold have scores that are within that of the top candidate by less than the win margin, these candidates are also matched. (Minimum value 0, maximum value 1) |
| Consider All Threshold | 0.8 | ▼ ▲ | Threshold above which a candidate intent or flow will match, regardless of the win margin and whether there are higher scoring candidates. (Minimum value 0, maximum value 1) |
| Consider Only Current Context Threshold | 0.8 | ▼ ▲ | If the confidence score for an intent in the current context exceeds this threshold, matches from other intent resolution models are not considered. (Minimum value 0, maximum value 1) |
| Explicit Invocation Confidence Threshold | 0.8 | ▼ ▲ | Confidence threshold that an utterance has to reach to be considered an explicit invocation. (Minimum value 0, maximum value 1) |

DA routing configuration details

- Built-In System Intent Confidence Threshold
 - Threshold for identifying built-in system intents (e.g. help and exit)
- Candidate Skills Confidence Threshold
 - The minimum confidence score required to a match a candidate skill
- Confidence Win Margin
 - If a candidate's confidence score exceeds that of other candidates by this value or more, it will be matched



Topic agenda

- 1 ➤ Recap of chatbot anatomy, skills, routing intro
- 2 ➤ Routing terms and concepts, Implicit/Explicit routing
- 3 ➤ Built-in Digital Assistant intents/routing
- 4 ➤ Tuning the routing model
- 5 ➤ Dealing with unresolvedIntent

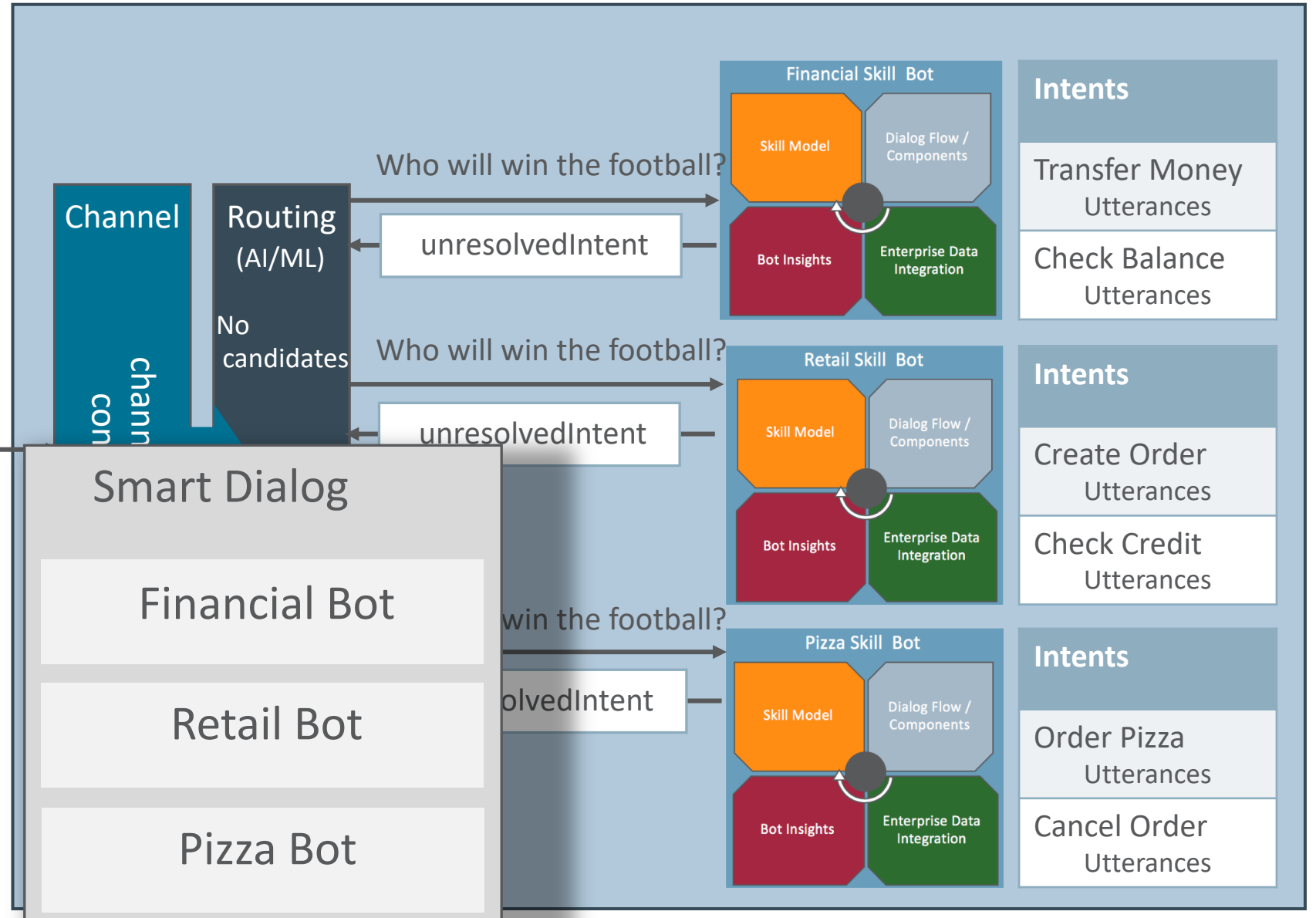
Dealing with unresolvedIntent

- unresolvedIntent is a good practice to aid successful intent resolution
 - Each skill should have one defined
 - unresolvedIntent on DA level only helps disambiguate exit and help
- You don't want user being told unresolvedIntent is an option
 - unresolvedIntent is the only intent we "hide" in smart dialog
 - If skill is already active, dealt with by that skills System.Intent unresolvedIntent

```
states:  
  getIntent:  
    component: "System.Intent"  
    properties:  
      variable: "userInput"  
    transitions:  
      actions:  
        TransferMoney: "startMoneyTransfer"  
        Balance: "startBalance"  
        TrackSpending: "startTrackSpending"  
        unresolvedIntent: "handleUnresolved"
```

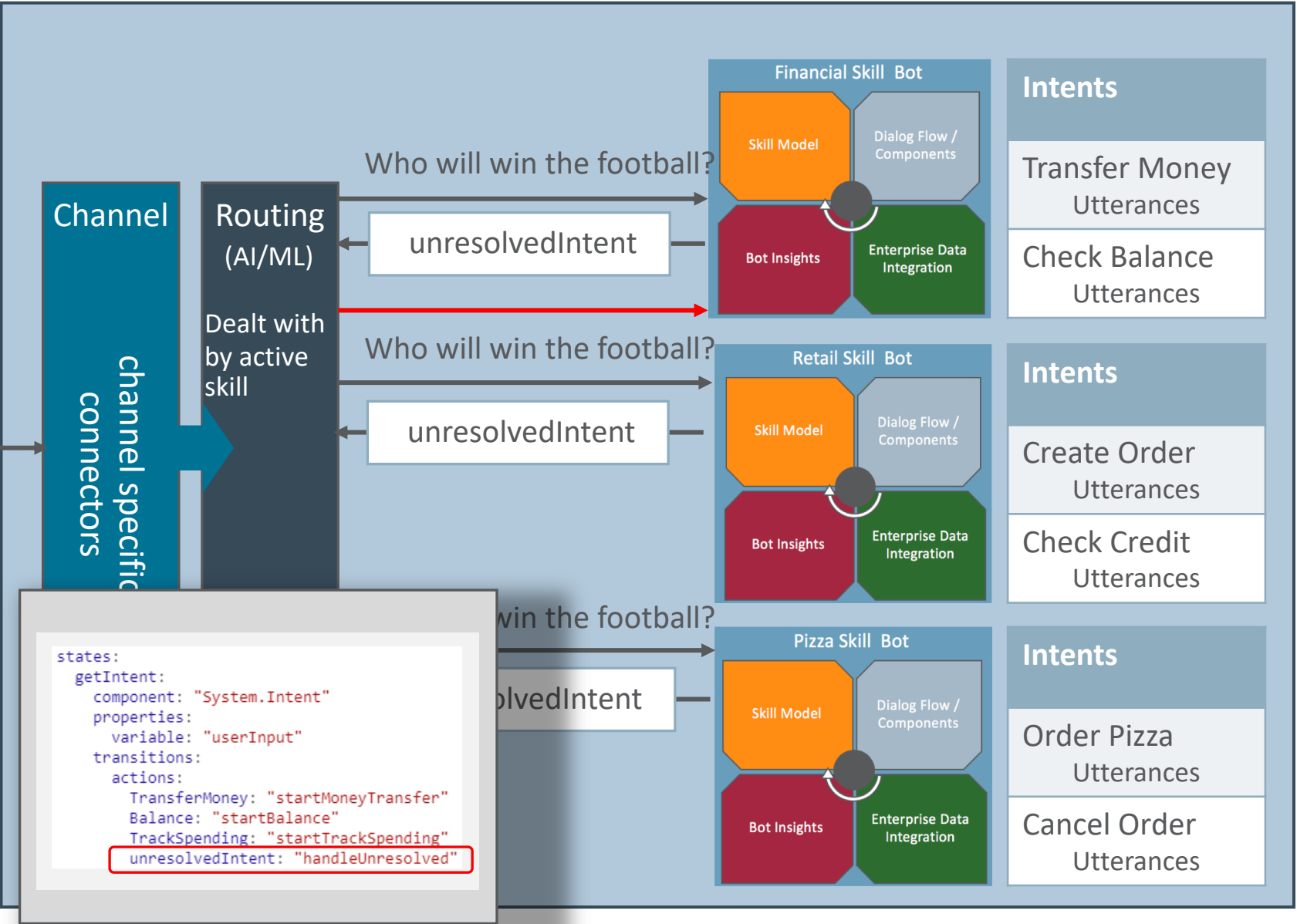
No active skill

Who will win the football?



Example – unresolvedIntent
Financial = active skill

Who will win the football?



Integrated Cloud

Applications & Platform Services

ORACLE®