AWS re:Invent

SVR304

bots + serverless = 

Vishal Singh, Senior Product Manager,
AWS Lambda

December 1, 2016

amazon web services

What to Expect from the Session

- Introduction to Serverless Chatbots
- Frameworks and integrations for serverless chatbots
- AWS Slack chatbot hackathon summary
- Amazon Lex
    - Architecture & benefits
    - Demo – building a Facebook chatbot using Lex + Lambda
- Mobile Hub enterprise SaaS connectors

Introduction to Serverless Chatbots

Chatbots

Chatbot is a service designed to simulate conversation with human users, accessible from a messaging interface.

Key characteristics:

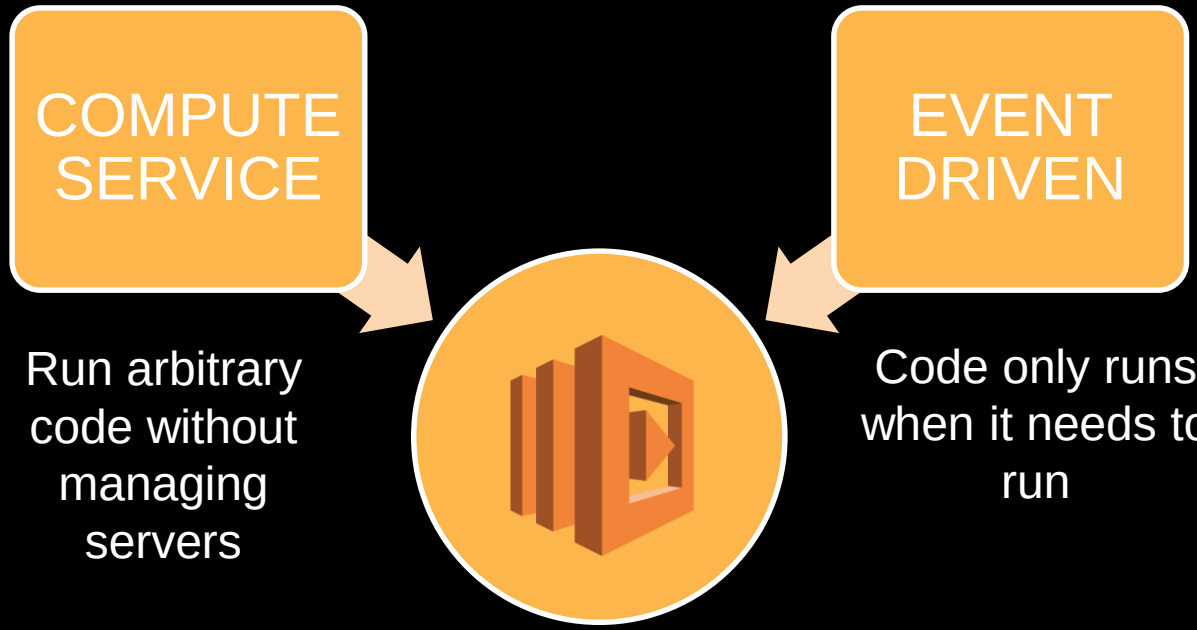
Natural

Accessible

On-demand

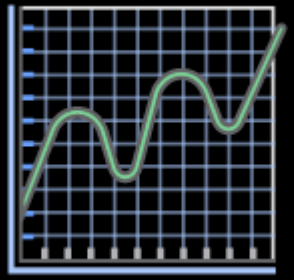
Efficient

AWS Lambda: Run code in response to events



Benefits of AWS Lambda

Code is all you need

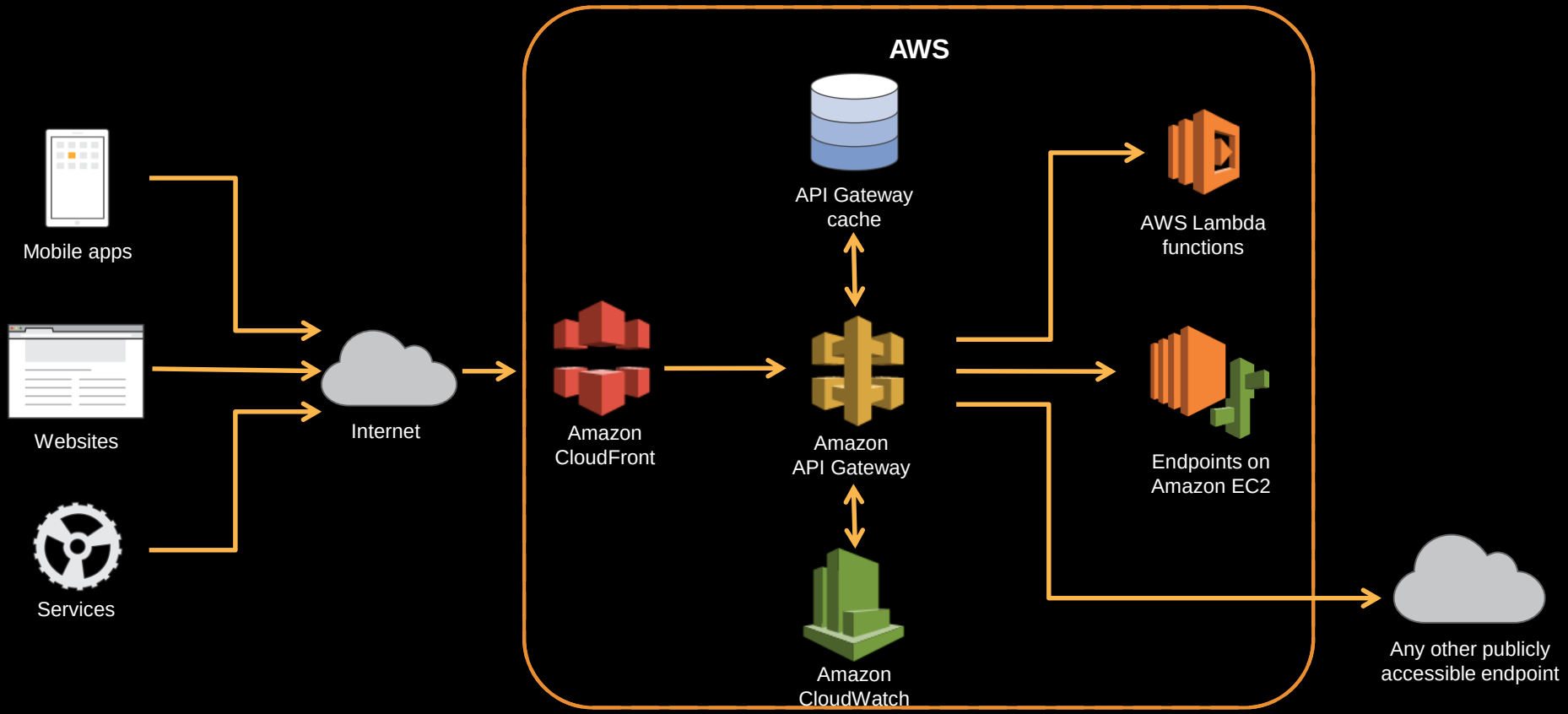


Event-driven scaling

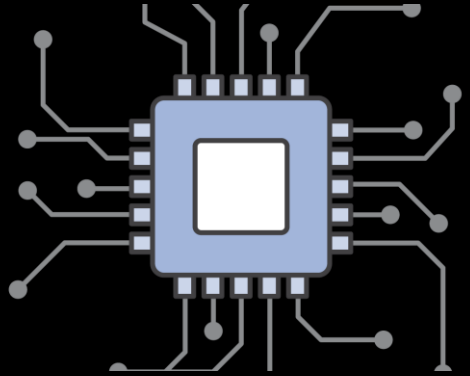
Never pay for idle

Availability and fault tolerance built in

Amazon API Gateway: Serverless APIs



Benefits of Amazon API Gateway



Create a unified API frontend for multiple microservices

DDoS protection and throttling for your back end

Authenticate and authorize requests to a back end

Serverless Apps

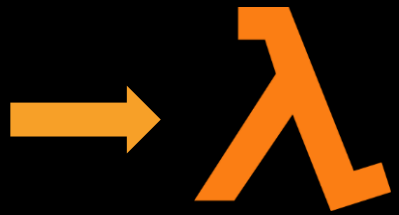
EVENT SOURCE

Changes in data state

Requests to endpoints

Changes in resource state

FUNCTION

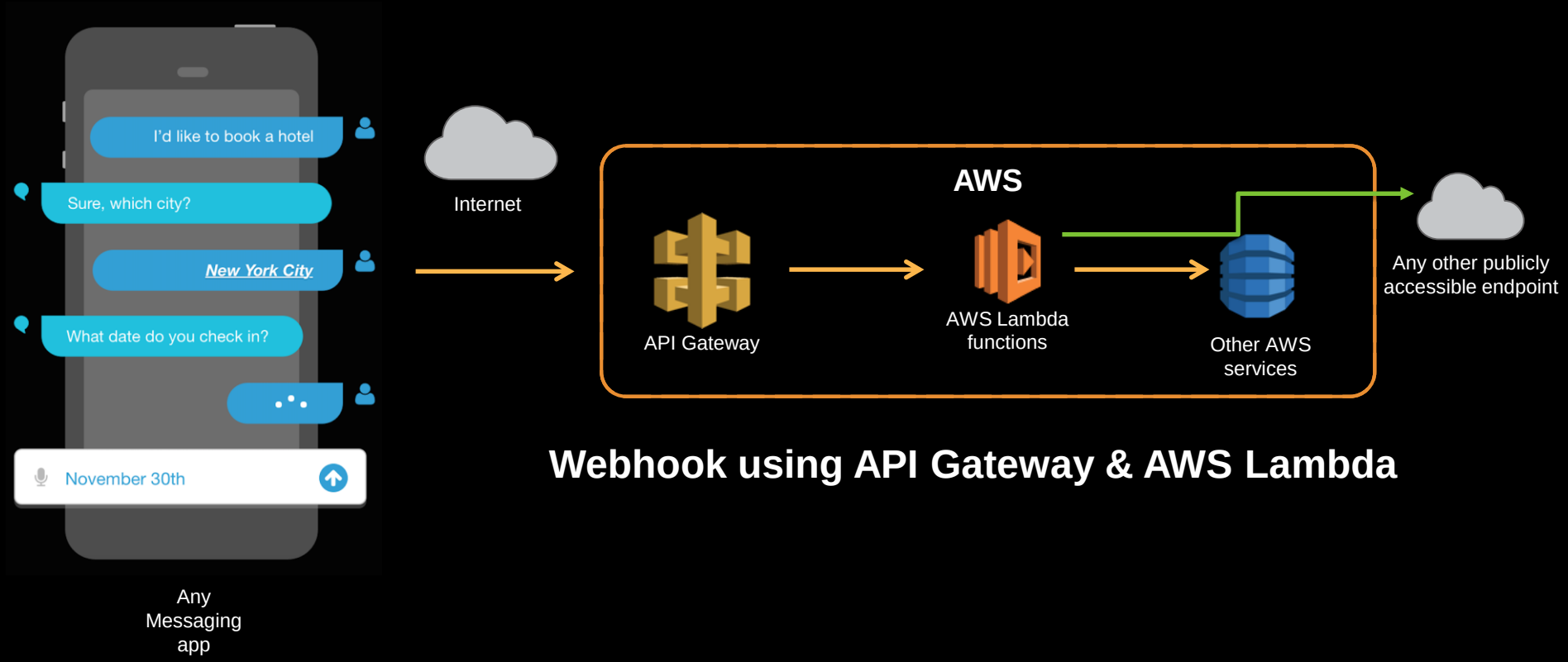


Node
Python
Java
C#

NEW!

SERVICES (ANYTHING)

The Serverless Chatbot



Webhook using API Gateway & AWS Lambda

Things to Note

Intent resolution: Language understanding is handled by the Lambda function using NLP libraries

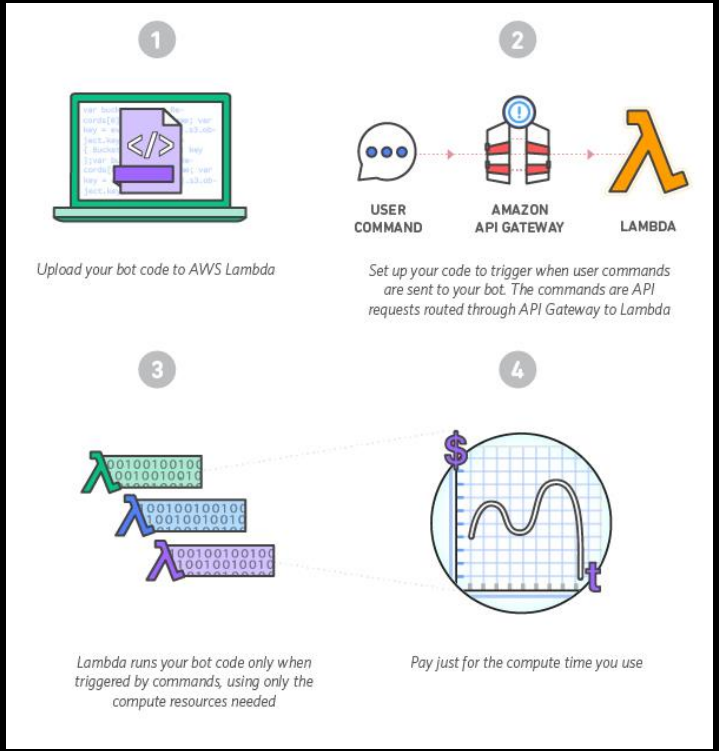
Authentication: API Gateway authenticates the incoming requests before passing to Lambda.

Partner integrations:

- Slack (blueprints available!)
- Twilio
- Hipchat

Building Chatbots with Serverless

1. Upload Bot code to AWS Lambda
2. Set code to trigger when user interacts via API Gateway webhook
3. Integrate with Messaging service
4. Test your chatbots



Frameworks and Integrations for Chatbots

Frameworks and Integrations for Serverless Chatbots

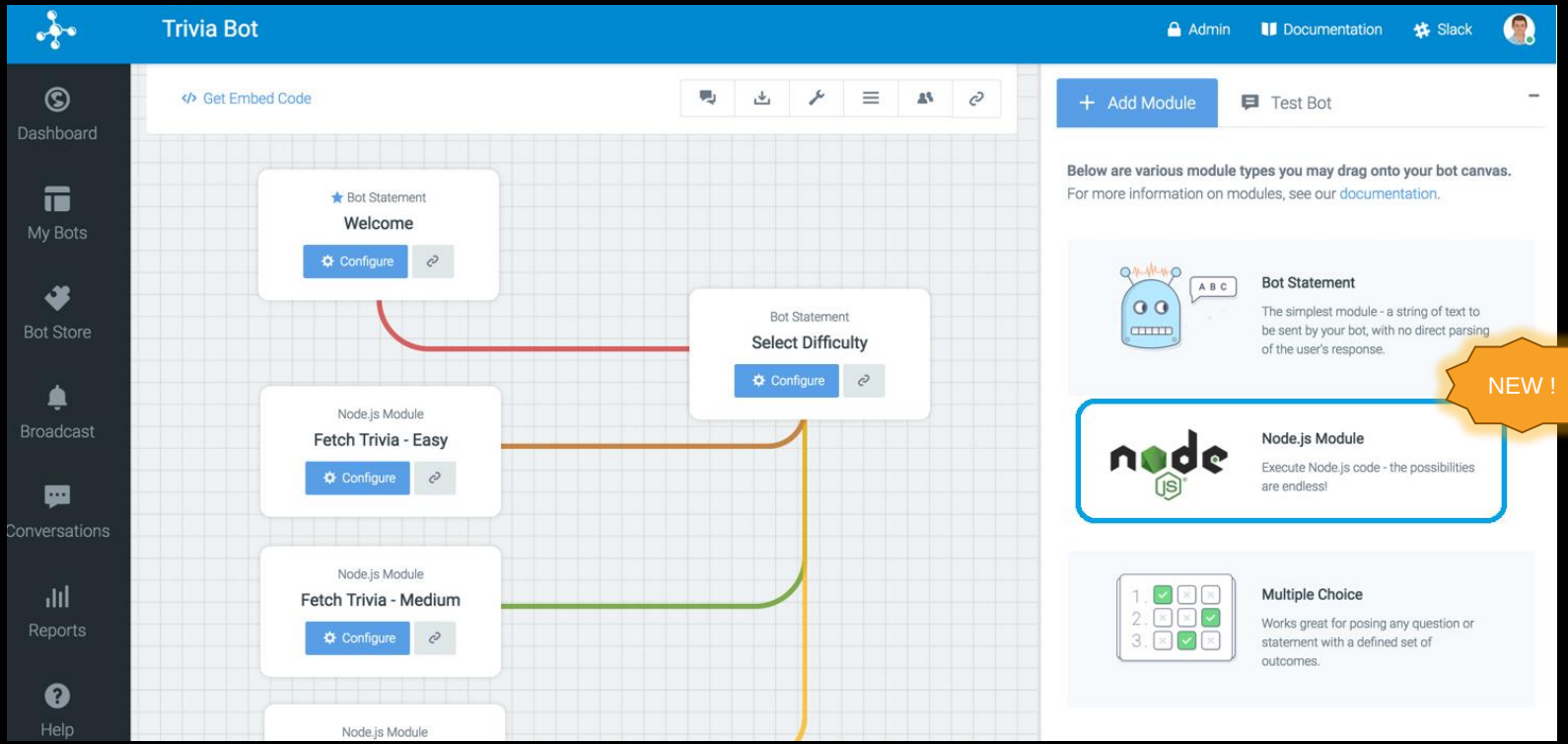
AWS SAM – AWS Serverless Application Model

- Specification to express a serverless chatbot application
- Helps in quickly deploying your application via CloudFormation

Other community frameworks:

- Claudia-bot-builder – Node open source framework
- Serverless.com – Serverless framework

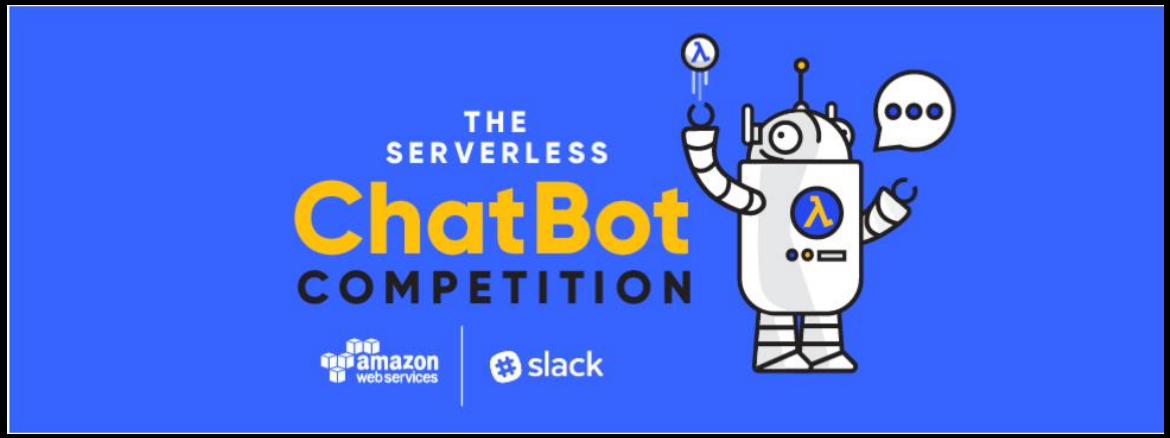
motion.ai + NEW !



AWS Slack Chatbot Hackathon Summary

AWS Slack Chatbot Hackathon Summary

- 2 months virtual hackathon (Aug10–Sept29 ‘16)
- 650+ registrations
- ~70 submissions
- Using various frameworks, libraries, open source and more

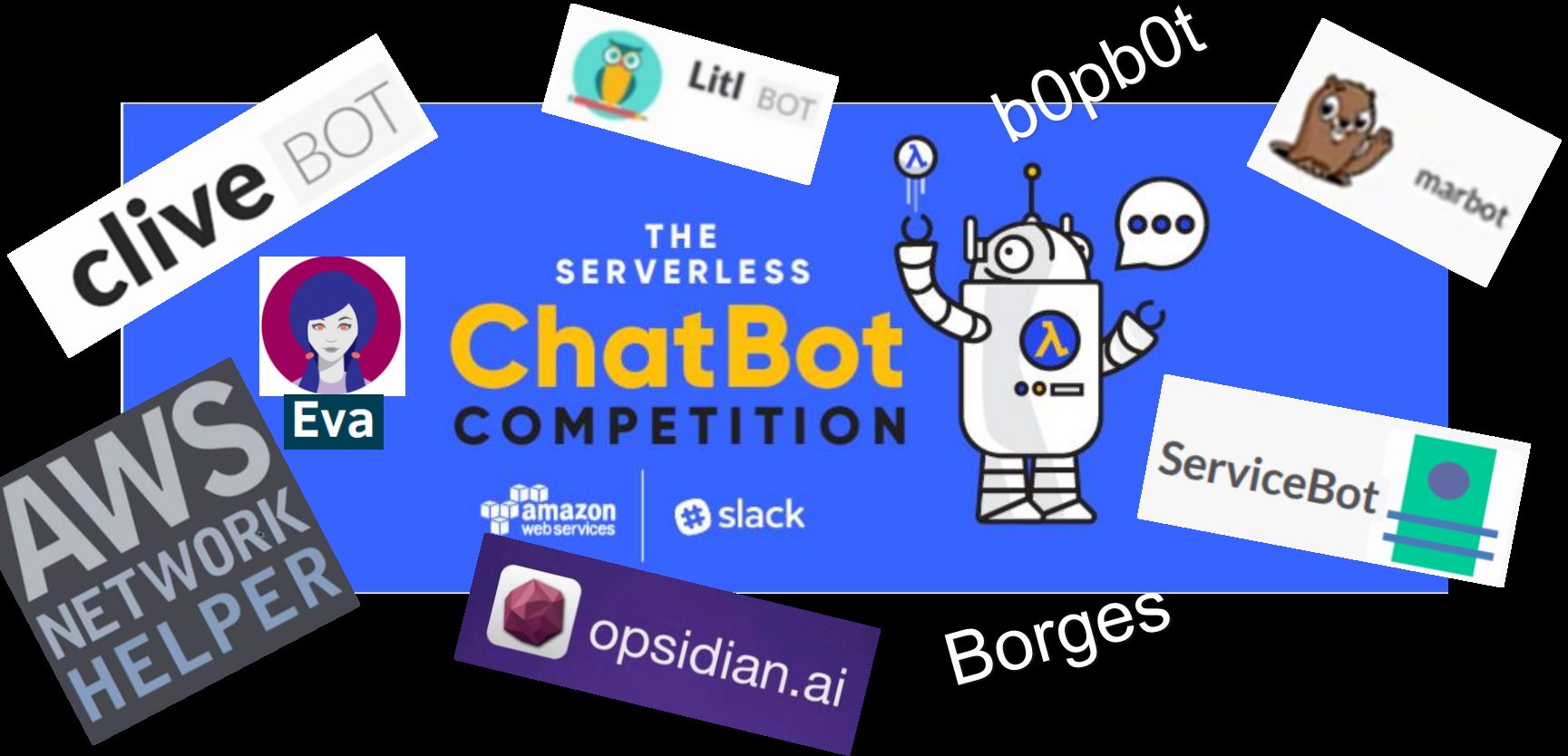


https://awschatbot.devpost.com/submissions

Hackathon Popular Chatbot Use Cases

- ChatOps
- Education
- Social
- Translation
- Travel
- Trivia
- Polls/quiz
- Knowledge Base
- Many more…

AWS Slack Hackathon Winners - Congratulations!

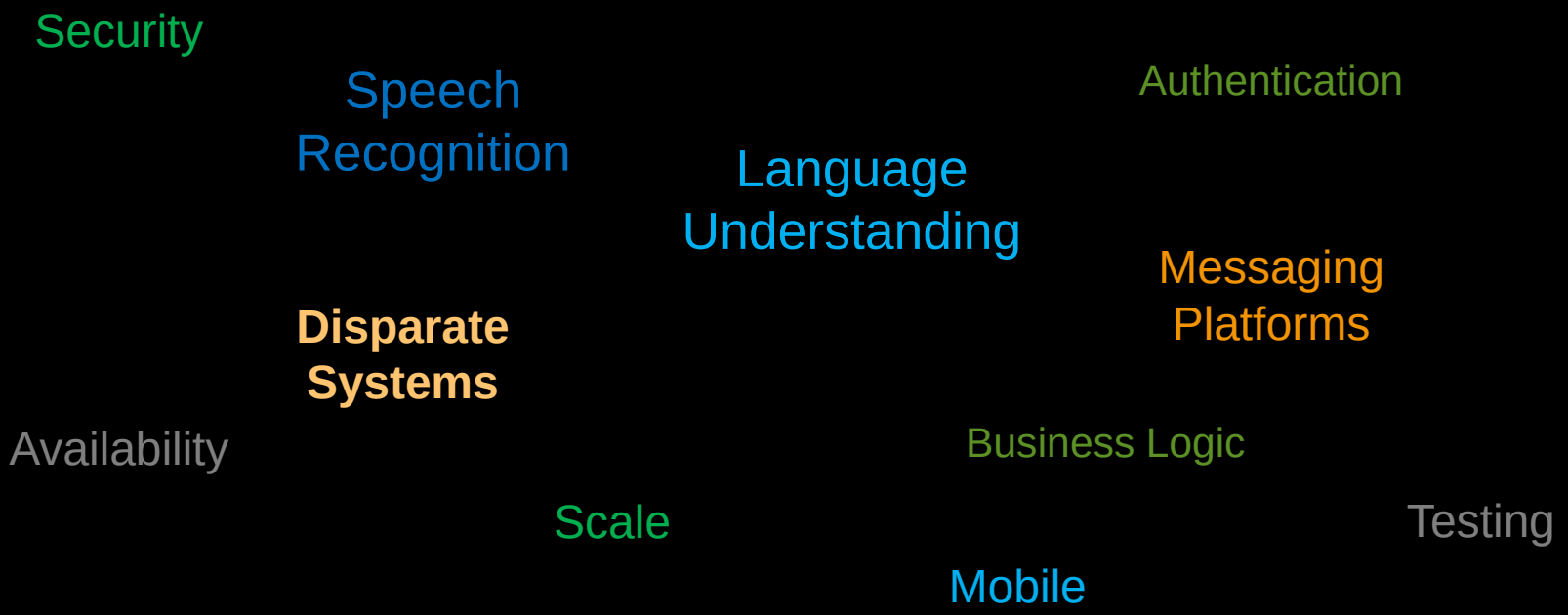


Chatbots Architectural Recommendations

- Authentication and authorization
  - Sigv4, Amazon Cognito, custom authorizers
  - Lambda execution role
- Environment variables 

  - Encrypt your secrets
- Persistence of state & sessions
  - DDB, S3, or any data store
- Reuse bot logic across messaging platforms

Developer Challenges

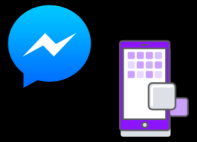


Conversational interfaces need to combine a large number of sophisticated algorithms and technologies

Amazon Lex

Amazon Lex Features

Text and speech language understanding: powered by the same technology as Alexa



Deployment to chat services

Designed for builders: efficient and intuitive tools to build conversations; scales automatically

Versioning and alias support

Enterprise SaaS connectors: connect to enterprise systems

Text and Speech Language Understanding

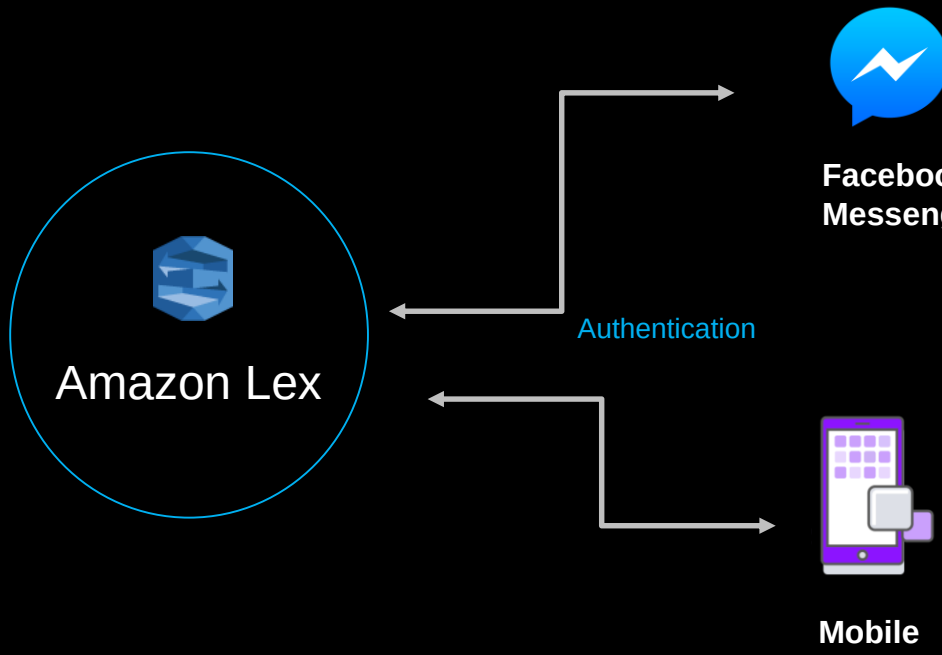
Speech Recognition

Natural Language Understanding

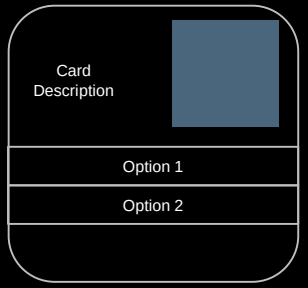
Powered by the same Deep Learning technology as Alexa

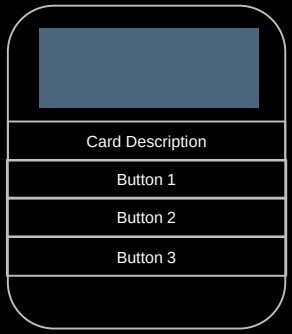
Deployment to Chat Services

One-Click Deployment

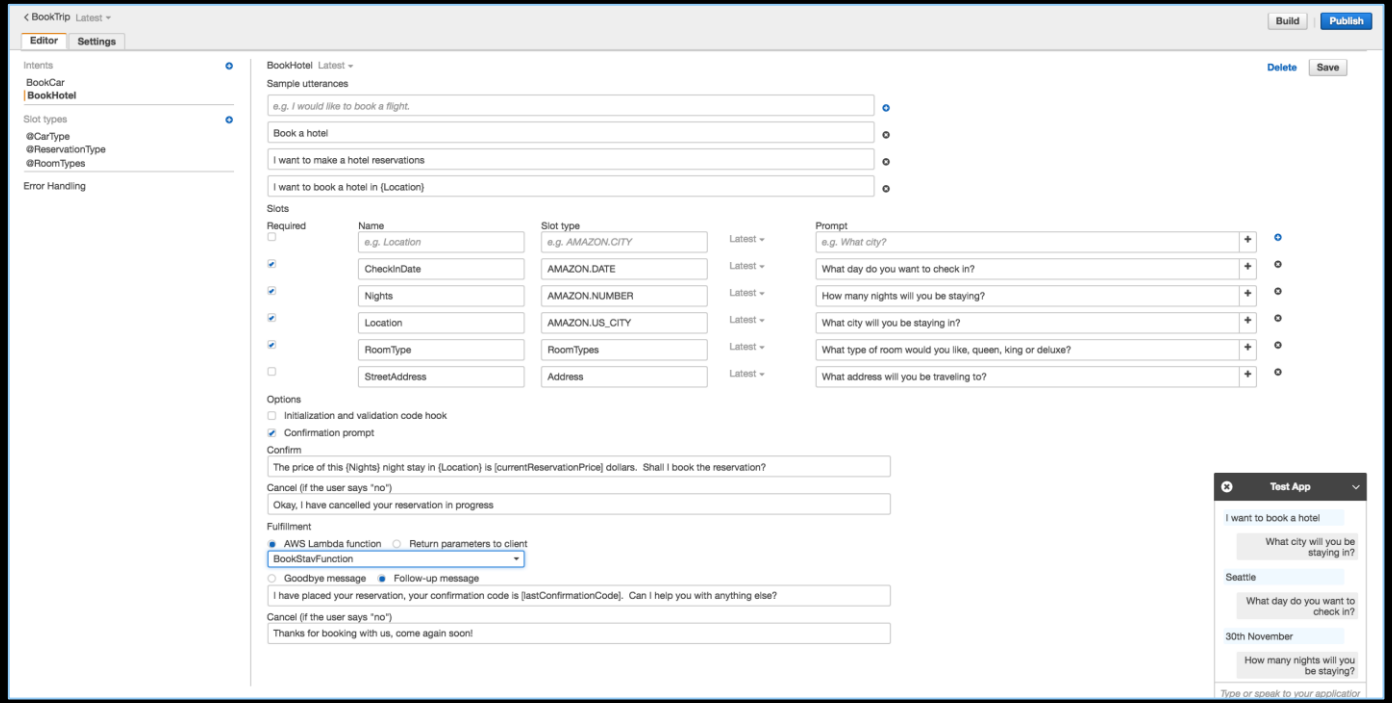


Rich Formatting



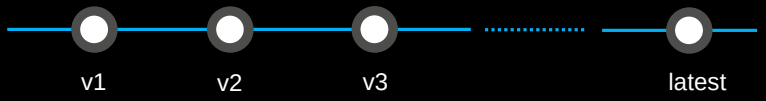


Designed for Builders



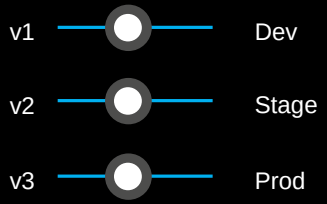
Efficient and intuitive tools to build conversations

Versioning and Alias Support



- Supported for Intents, Slots, and Bots
- Enables multi-developer environment
- Roll back to previous versions

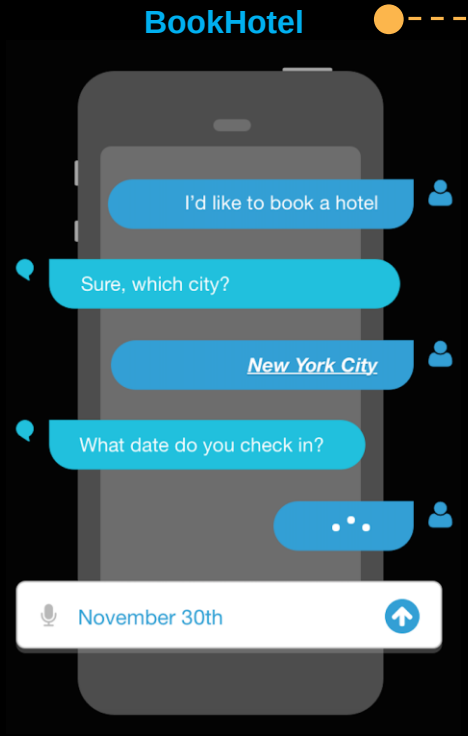
Versioning



- Deploy different aliases to different platforms
- Run different stacks for dev, stage, and prod environments
- Target different user groups with different aliases

Alias

Lex Bot Structure

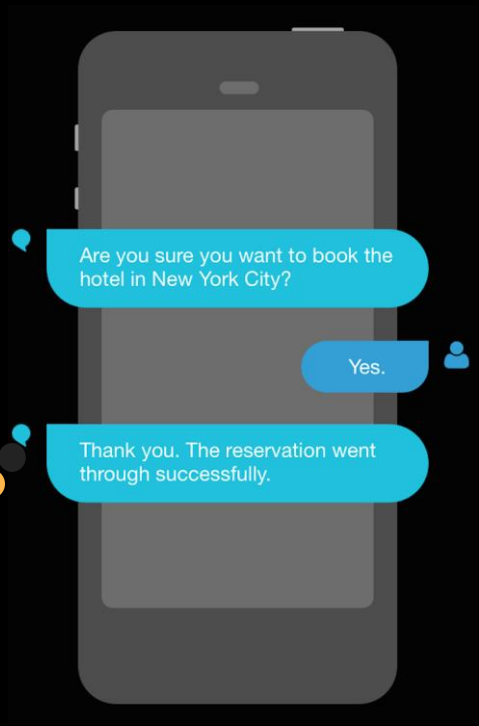


Intents
An intent performs an action in response to natural language user input

Utterances
Spoken or typed phrases that invoke your intent

Slots
Slots are input data required to fulfill the intent

Fulfillment
Fulfillment mechanism for your intent



Utterances

I’d like to book a hotel

I want to make my hotel reservations

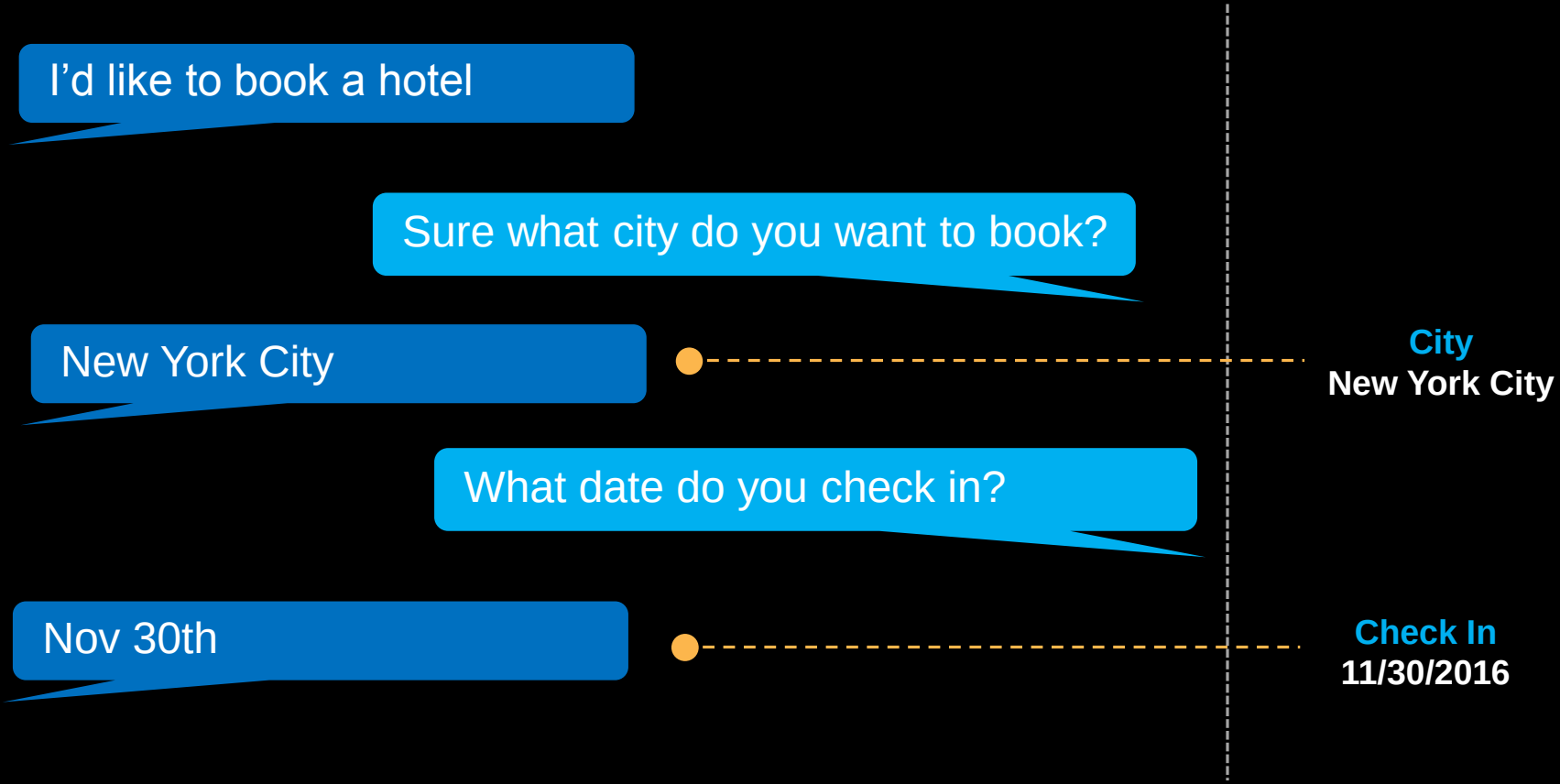
Can you help me book my hotel?

I want to book a hotel in New York City

Slots

Slot	Type	Values
destination	City	New York City, Seattle, London, …
Check In	Date	Valid dates
Check Out	Date	Valid dates

Slot Elicitation



Fulfillment

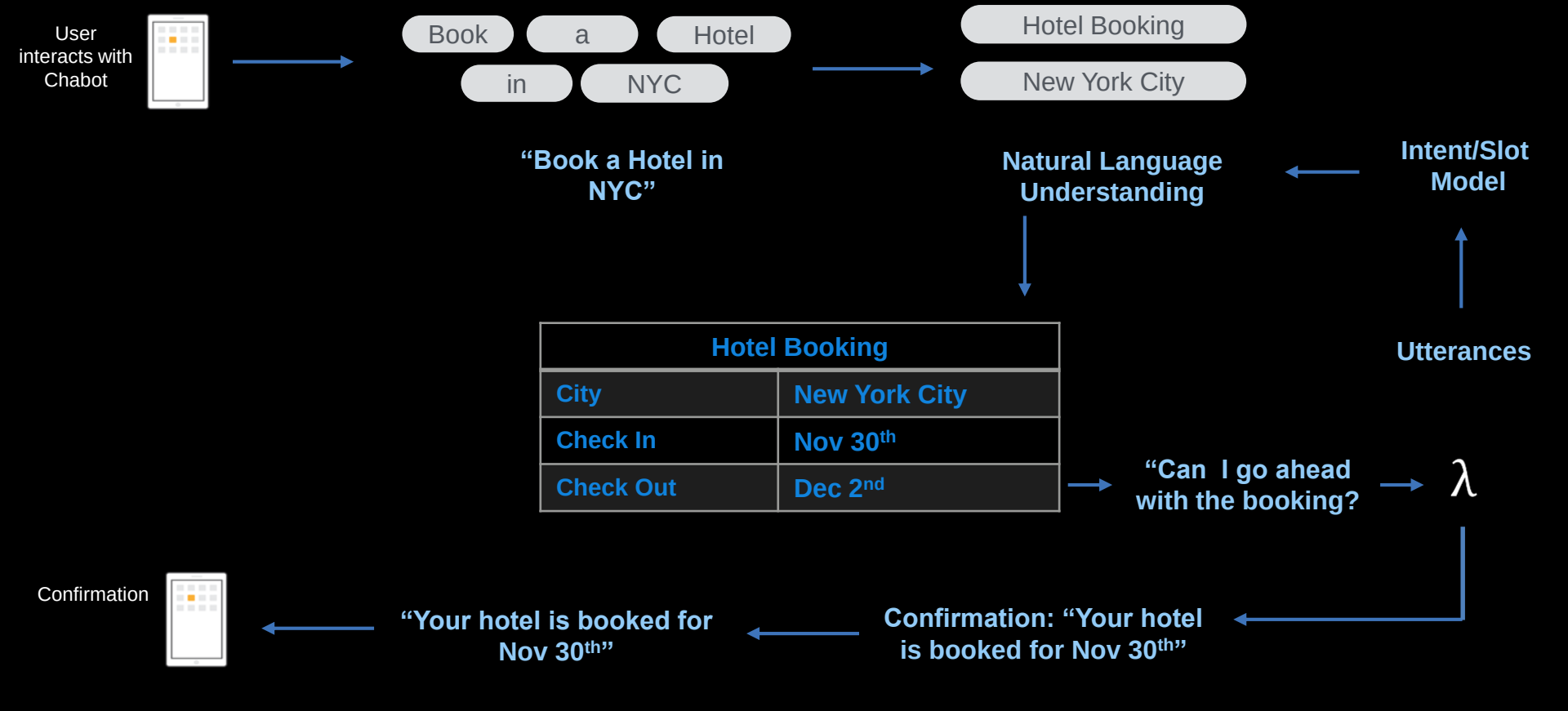
Intents and slots passed to AWS Lambda function for business logic implementation.

AWS Lambda Integration

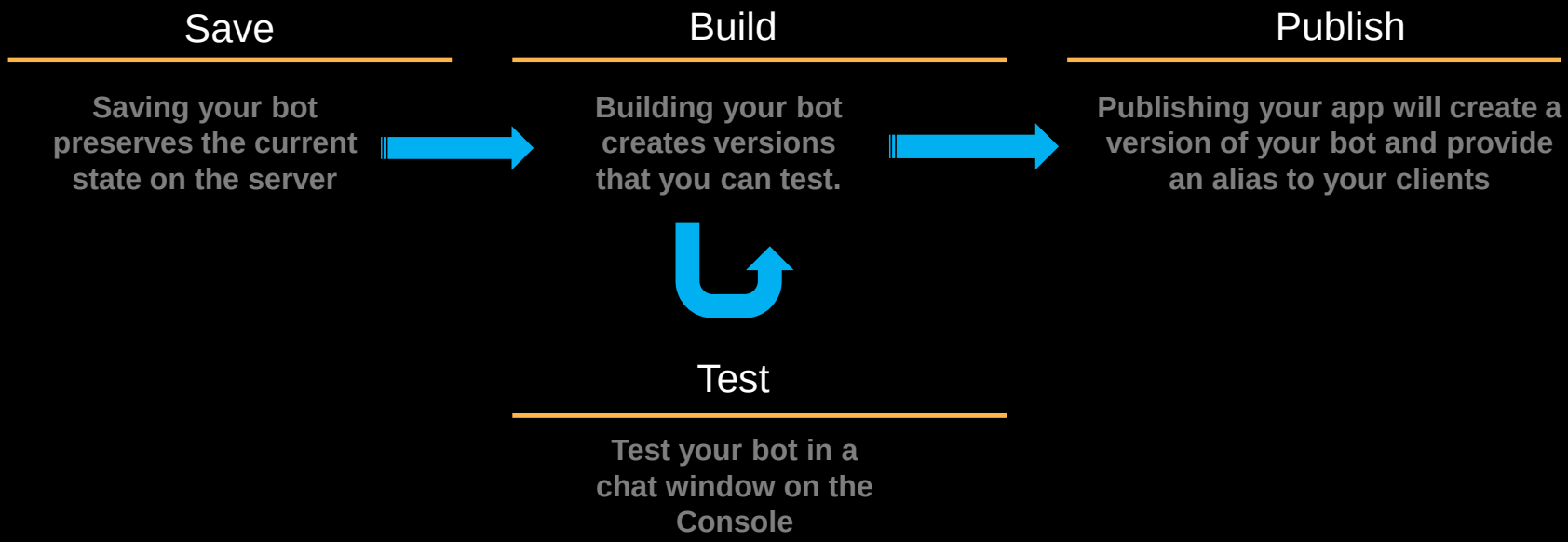
User input parsed to derive intents and slot values. Output returned to client for further processing.

Return to Client

“Book a Hotel”


User interacts with Chabot
Book
a
Hotel
in
NYC
Hotel Booking
New York City
“Book a Hotel in NYC”
Natural Language Understanding
Intent/Slot Model
Utterances
Hotel Booking
City
New York City
Check In
Nov 30th
Check Out
Dec 2nd
“Can I go ahead with the booking?
λ
Confirmation
“Your hotel is booked for Nov 30th”
Confirmation: “Your hotel is booked for Nov 30th”

Save, Build, and Publish



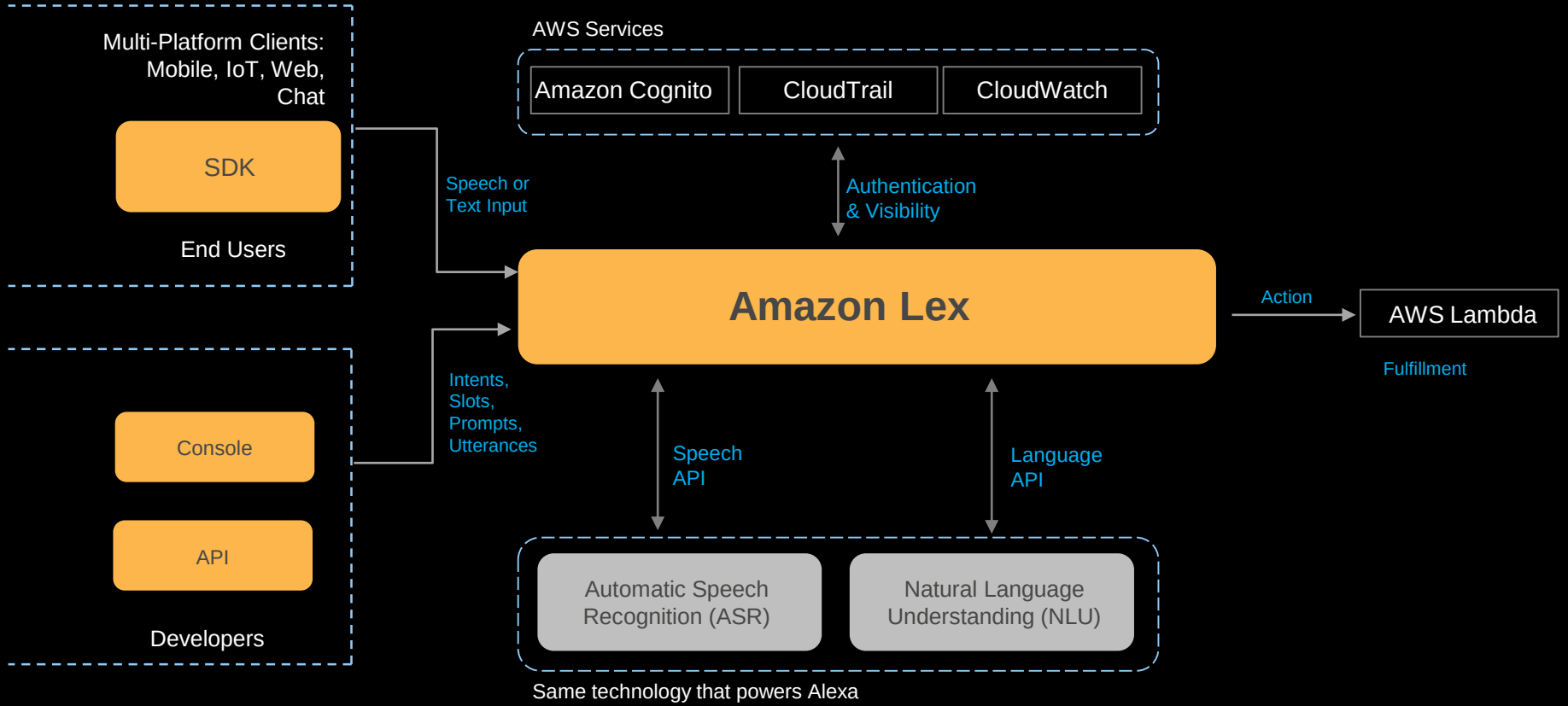
Monitoring

Missed Utterance Count

Request Latency

Track your bot

Amazon Lex Technology



Demo

Build a FB “Moviepedia” chatbot using Lex + Lambda

AWS services

Find a service by name (for example, EC2, S3, Elastic Beanstalk).

Recently visited services

- Lambda
- CloudWatch
- IAM
- Mobile Hub

All services

Build a solution

Get started with simple wizards and automated workflows.

- Launch a virtual machine — With EC2, ~1 minutes
- Build a web app — With Elastic Beanstalk, ~6 minutes
- Deploy a serverless microservice — With Lambda, API Gateway, ~2 minutes
- Host a static website — With S3, CloudFront, Route 53, ~5 minutes
- Create a backend for your mobile app — With Mobile Hub, ~5 minutes
- Register a domain — With Route 53, ~3 minutes

Learn to build

See all

Learn to deploy your solutions through step-by-step guides, labs, and videos.

Featured next steps

Manage your spend

Get real-time billing alerts based on your cost and usage budgets. Start now

Get best practices

Use AWS Trusted Advisor for security, performance, cost and availability best practices. Start now

Announcements

AWS Marketplace

Amazon Lex Use Cases

Informational Bots

Chatbots for everyday consumer requests

- News updates
- Weather information
- Game scores ….

Application Bots

Build powerful interfaces to mobile applications

- Book tickets
- Order food
- Manage bank accounts ….

Enterprise Productivity Bots

Streamline enterprise work activities and improve efficiencies

- Check sales numbers
- Marketing performance
- Inventory status ….

Internet of Things (IoT) Bots

Enable conversational interfaces for device interactions

- Wearables
- Appliances
- Auto ….

Amazon Lex Benefits

Easy to use

High quality text and speech language understanding

Seamlessly deploy and scale

Built-in integration with the AWS platform

Cost-effective

Mobile Hub + Enterprise SaaS Connectors

AWS Mobile Hub Integration

Authenticate users

Synchronize data

Analyze user behavior

Track retention

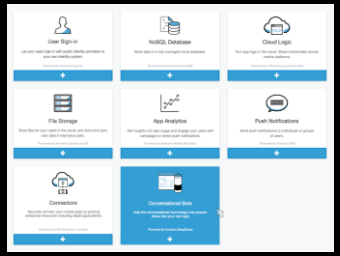
AWS Mobile SDKs

AWS Mobile Hub

Store and share media

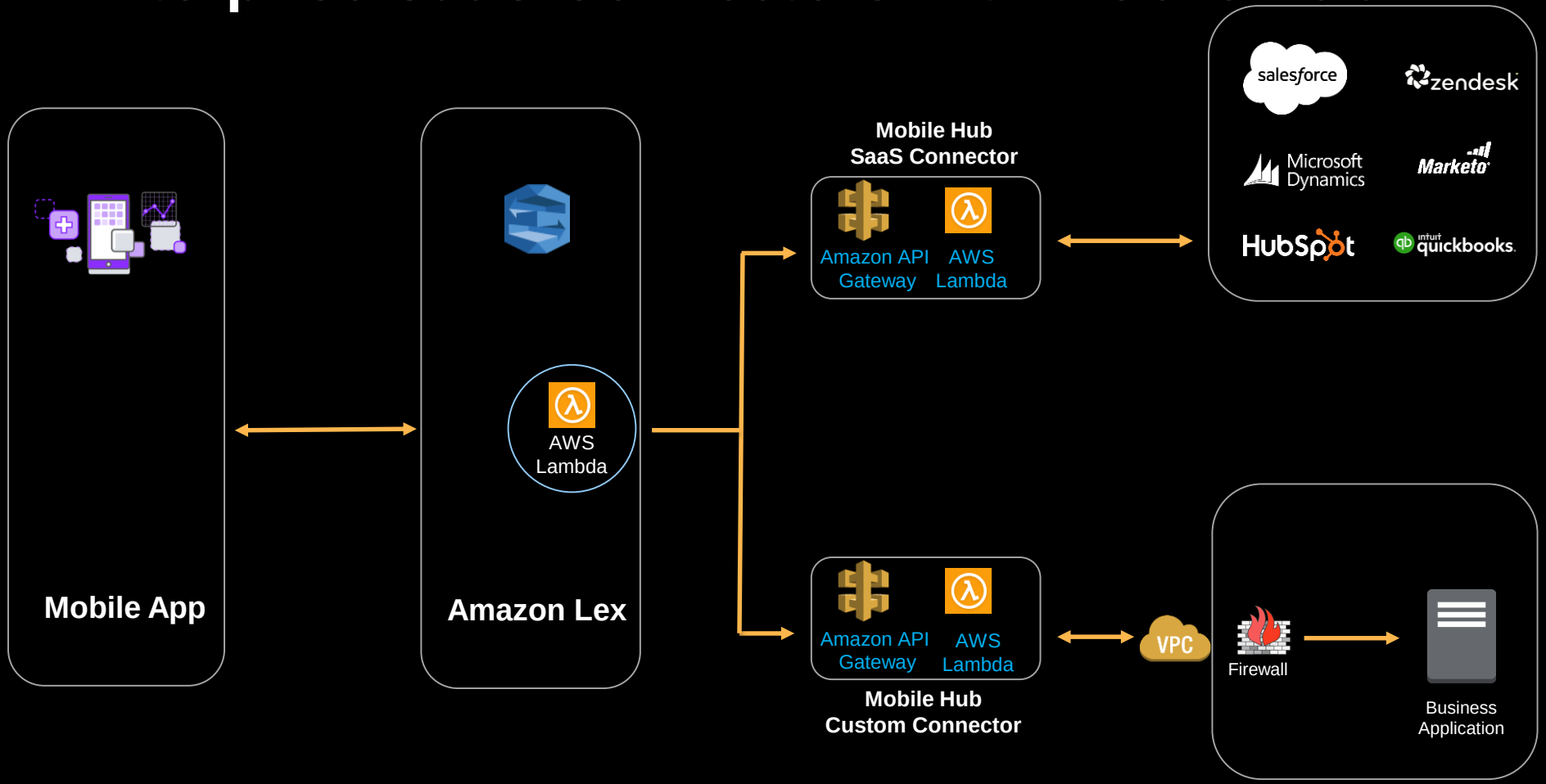
Conversational Bots

Lex

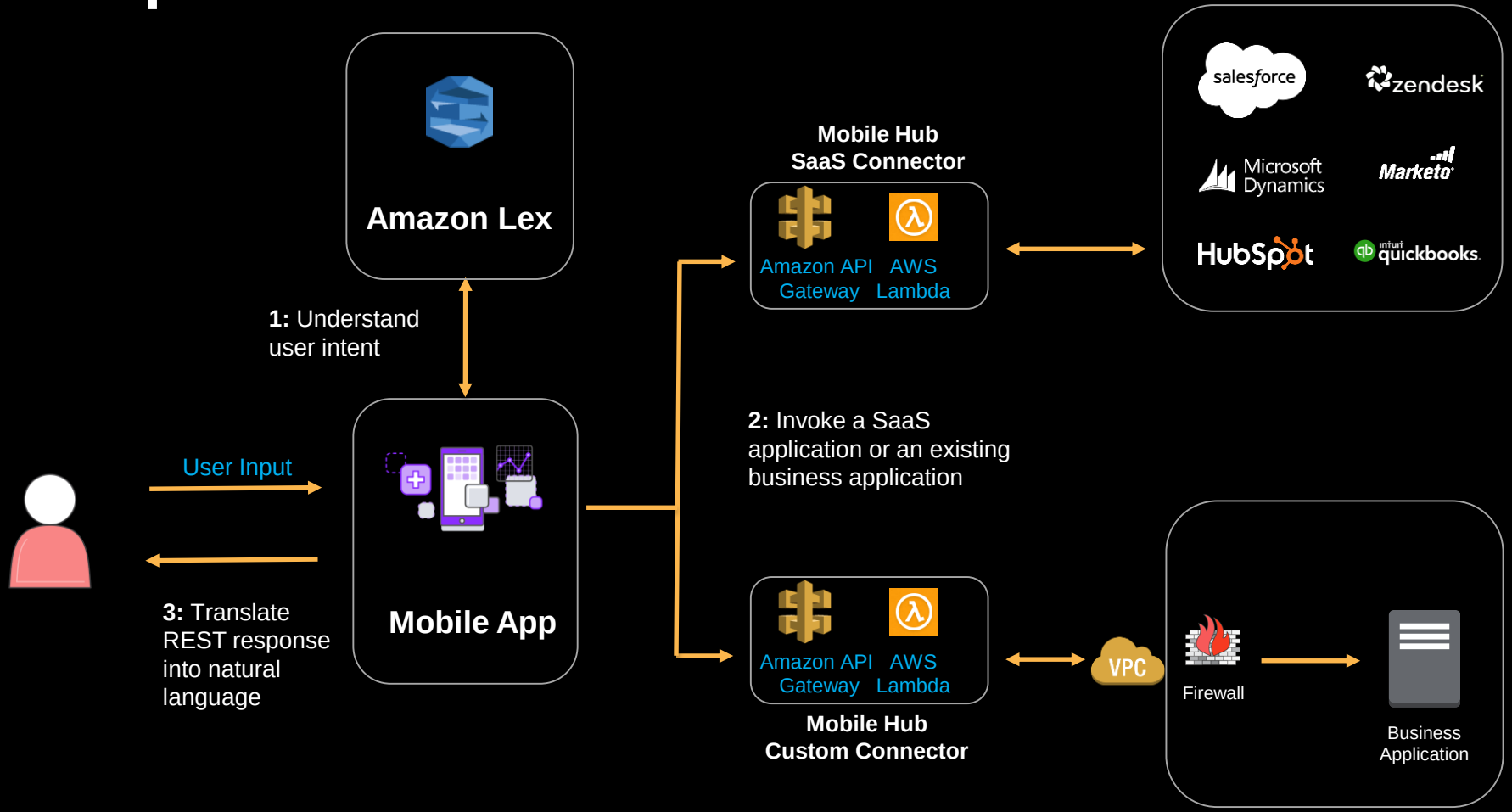


More ….

Enterprise SaaS Connectors with Mobile Hub



Enterprise SaaS Connectors with Mobile Hub

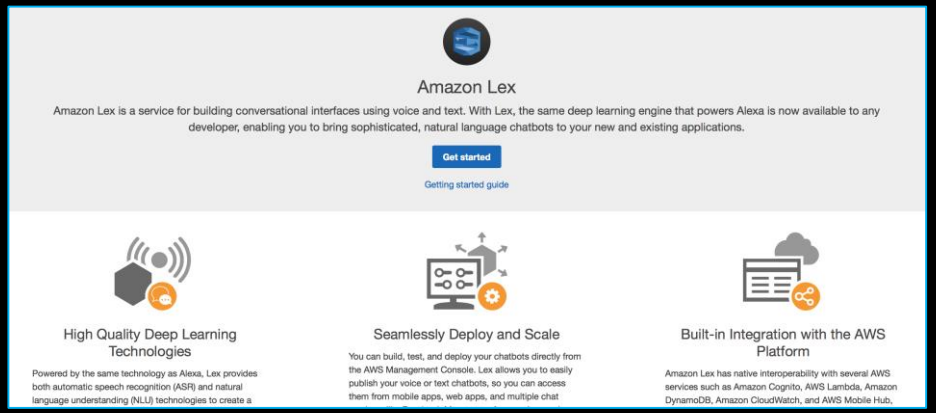


Recap

- First, we looked into creating chatbots as serverless webhooks using AWS Lambda and API Gateway
- Next, we added natural language intelligence to your chatbots using Amazon Lex and AWS Lambda
- Finally, we looked at mobile hub integration and enterprise SaaS connectors to enrich your chatbot with enterprise data

Key Takeaway: Register for the Preview @ aws.amazon.com/lex

Sign-Up & Whitelist



Build your first serverless chatbot using samples in Lex and Lambda blueprints!

Thank you!

Remember to complete your evaluations!