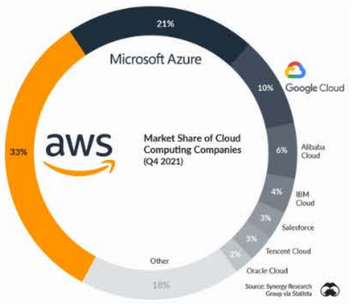
Northeastern University
Khoury College of Computer Sciences

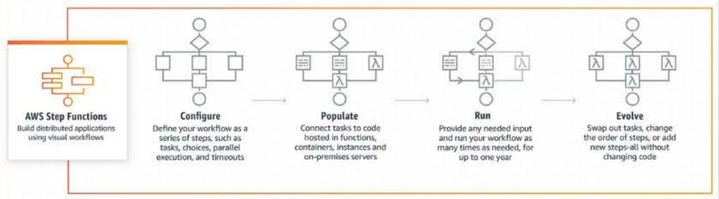
SIYANG ZHANG, HUAXING WANG, QIU CHEN,
IMDADUL AL IMRAAN,MICHAL AIBIN

EXPERIMENTAL STUDY ON MICROSERVICES ORCHESTRATION WITH AMAZON STEP FUNCTION

Background



- Cloud computing has become an immensely growing market in 2022.
- Amazon Web services (AWS) is dominating the cloud market by almost 34%



- AWS Step Functions is a serverless orchestration service that lets you integrate with AWS Lambda functions and other AWS services to build business-critical applications.
- Through Step Functions' graphical console, you see your application's workflow as a series of event-driven steps.

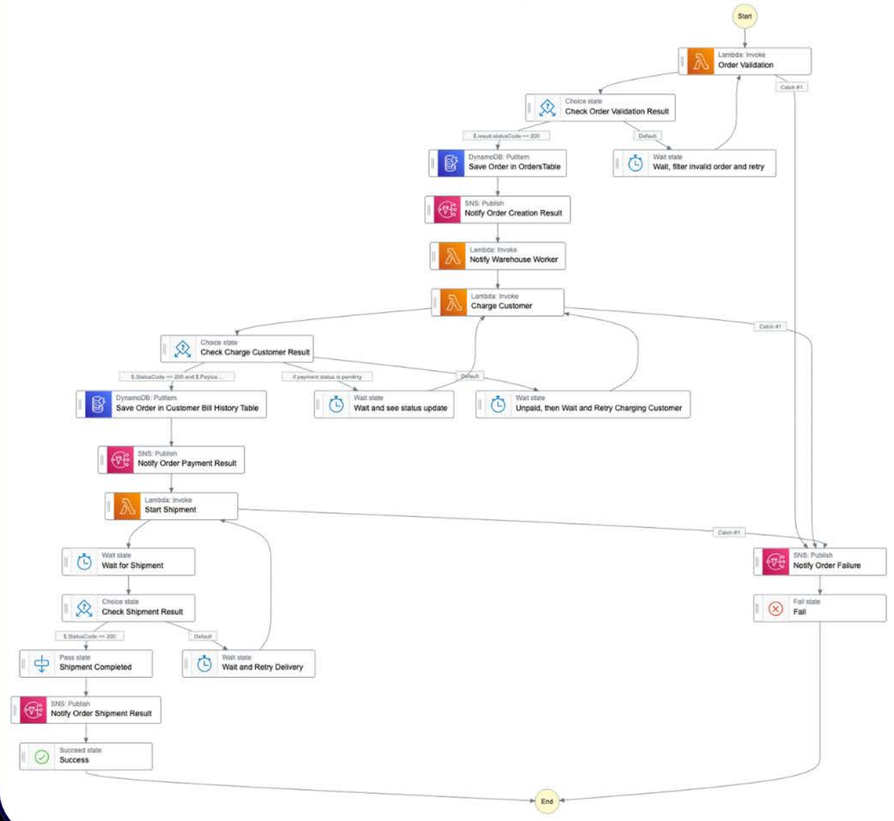
Problem Overview

- Not enough Research have been done for AWS Step Function in terms of Performance and Cost.
- What is the advantange of Step Function when comparing with Event-Driven AWS application and Code-based application?

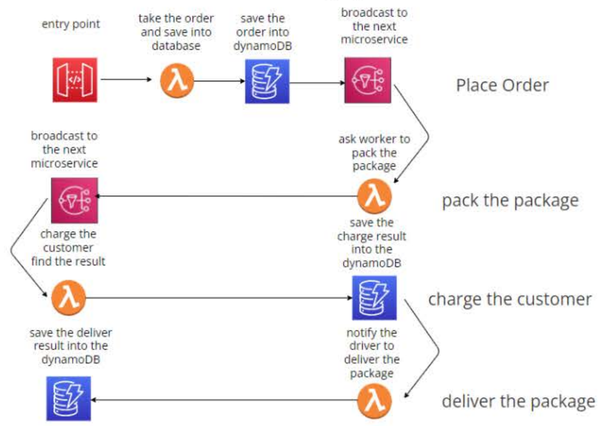
Event-Driven Application VS **Code-Based Application**

Experimental E-Commerce Application

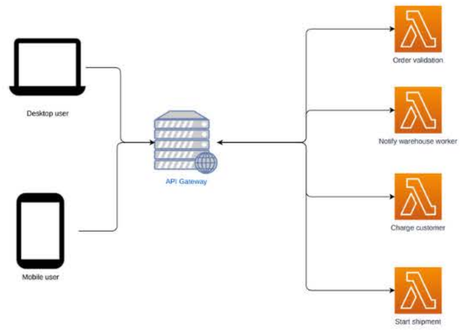
Step Function Approach



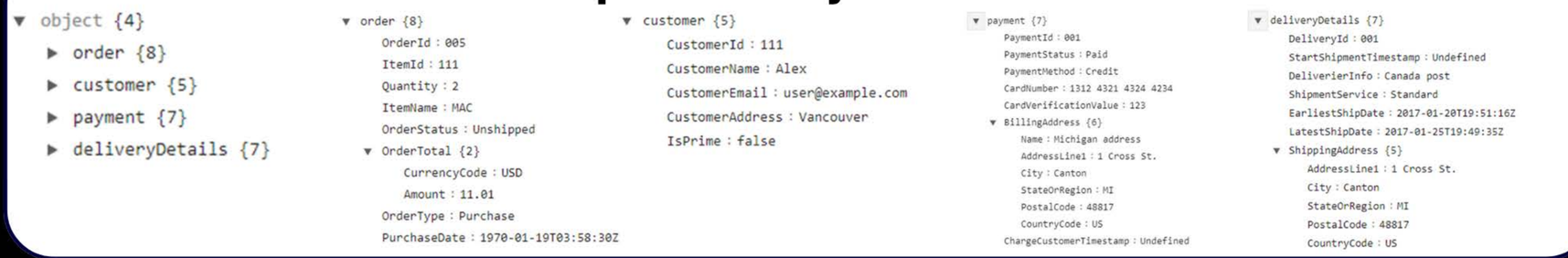
Event-Driven Approach



Code-Based Approach



Input Json Payload



Northeastern University
Khoury College of Computer Sciences

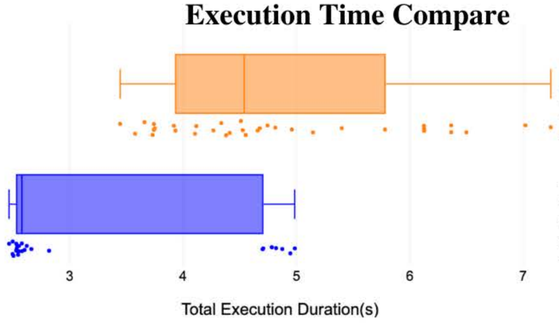
SIYANG ZHANG, HUAXING WANG, QIU CHEN,
IMDADUL AL IMRAAN,MICHAL AIBIN

EXPERIMENTAL STUDY ON MICROSERVICES ORCHESTRATION WITH AMAZON STEP FUNCTION

Experiments & Results

Experiments Between Step Function Express and Standard

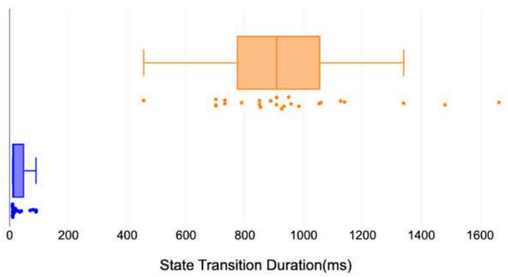
Execution Time Compare



Results: Average execution time of Express workflow is 3.199s, 37% faster than Standard workflow's 5.061s.

Analysis: JSON file as event call to simulate order, short-live execution, idempotent action e.g. DynamoDB put operation. All above features of application align with Express workflow more closely

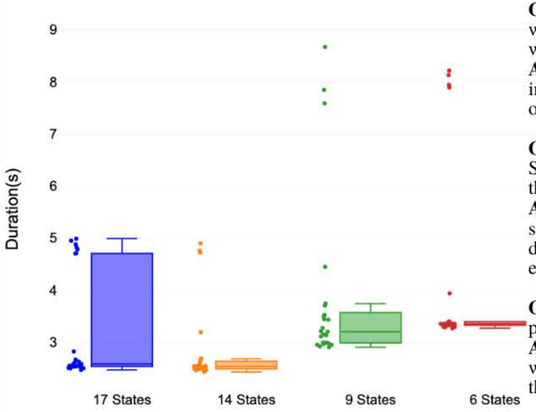
State Transition Duration Compare



Results: Most Express Workflow's state transition time lies in the range of 10ms to 14ms, significantly less than Standard workflow's 947ms average state transition time.

Analysis: The difference lies in the feature of Express workflow that it doesn't store state transition information, which greatly saves the transition time between states

Number of States Graph for Express Workflow

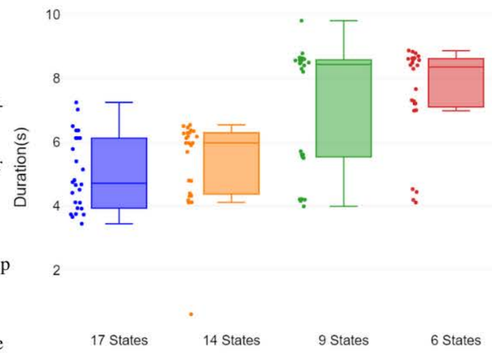


Observation 1: The average running time of the 17-States workflow is 3.1996s, slightly slower than the 14-States workflow's 2.82488s.
Analysis 1: Express workflow doesn't store state transition information therefore adjusting the number of Choice states only contributes to limited performance improvement.

Observation 2: The average running time of the 9-States & 6-States workflow is 3.8224s and 4.11444s respectively, longer than 17-States & 14-States workflow.
Analysis 2: Workflow with fewer states should perform better since the State Transition Time is saved. This abnormality demonstrates that Step Function orchestrates services more efficiently than Lambda orchestration

Observation 3: All four workflow's data layout displays group pattern: lie in two groups
Analysis 3: The pattern where data points from all four workflows lie in two major ranges is caused by the effect of the Lambda function's warm/cold start on ASF execution time

Number of States Graph for Standard Workflow

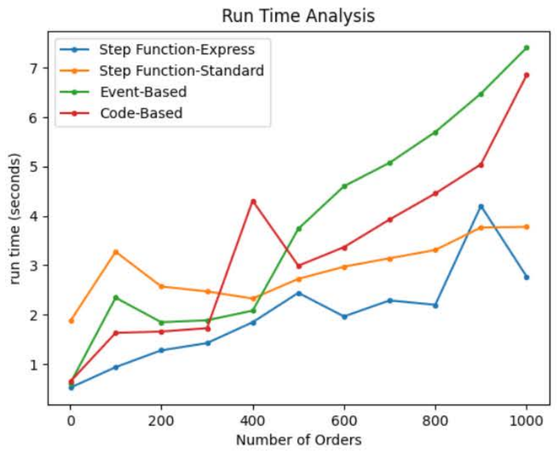


Observation 1:The average running time of the 17-States workflow is 5.061s, similar to the 14-states' 5.3887s.
Analysis 1: State transition of the choice states has a very limited effect on performance.

Observation 2: From 14 states to 9 states, the average running time increased to 7.32609s.
Analysis 2: Step Function orchestrates services more efficiently compared with invoking other AWS services within the Lambda function.

Observation 3: Data points grouping layout doesn't show a specific pattern.
Analysis 3: Lambda cold/warm start doesn't have a significant influence on the Standard Workflow's performance.

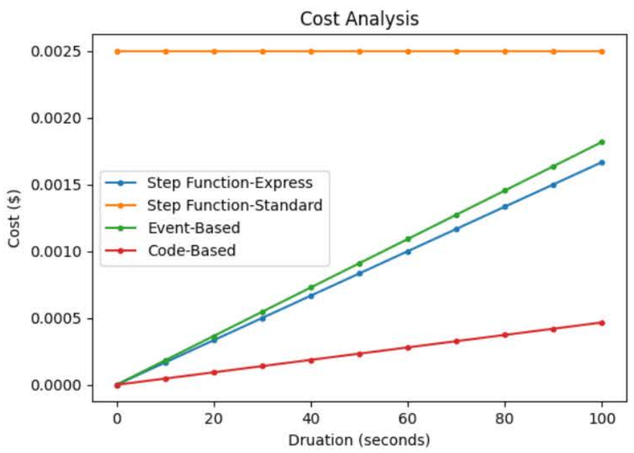
Experiments Between Step Function, Event-Based, Code-Based



1 to 100
All four applications showed the big jump.

200 to 400
All four application starts to decrease.

400 and 500
Event-based application starts to increase dramatically, Two step function grow steadily.



AWS step function standard
$0.0025 / 100 states

AWS step function express
$0.0000166667 per second

lambda($0.0000166667 / s)

SNS($0.5 / million notification)

DynamoDB($1.5 per million read and write pair)

Conclusion

In the experiments between Express and Standard workflow

- Adjusting the number of Choice states only contributes to subtle improvement in performance for Express workflow
- Standard workflow's feature of storing state transition information prolongs state transition time between states
- Service orchestration with Step Function performs better than invoking AWS services within the Lambda function
- Express ASF's performance is significantly influenced by the Lambda function' code/warm start. While this factor doesn't impact Standard Workflow too much.
- Express workflow's feature of not storing state transition information saves state transition time between states

- Short-term, event-based application with idempotent operation is more suitable to implement with Express workflow.

- In Standard workflow, reducing the non-service state only has little influence on the performance efficiency

Future Steps

- No available backend API to directly export execution logs, further research could add tools to enable automatic export execution logs.
- The current application is more aligned with Express workflow's feature, further research could be executed on applications that fit Standard workflow more closely, to test its performance and cost
- One limitation is that the current application is relatively simple, contains only limited AWS resources, future steps could add more components to simulate real-world applications