

Re-Engineering Autodesk

Digital Transformation with AWS Lambda

 dynatrace
Perform

Mike Deck

Principal Solutions
Architect



Samy Senthivel

Digital Enterprise
Monitoring Manager



Josh Toklu

Principal DevOps Engineer



What is Serverless?



Mike Deck

Principal Solutions
Architect



What is serverless?







How do you want to spend your time?



VS.



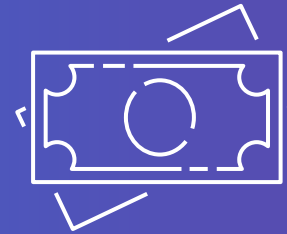
Amazon views **serverless** as the native architecture of the cloud



No provisioning,
no management



Automatic
scaling

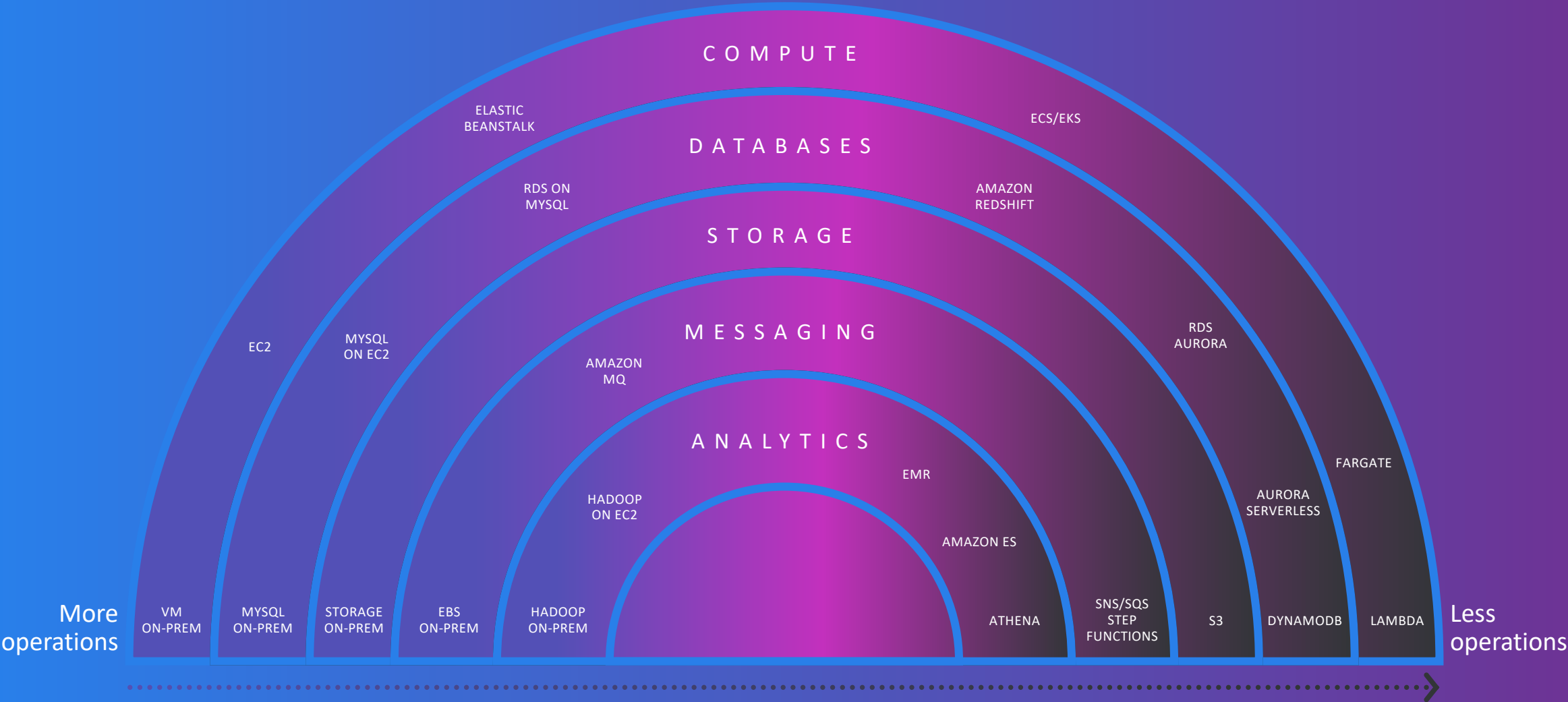


Pay for
value



Highly available
and secure

Serverless is an **operational construct**



Serverless compute



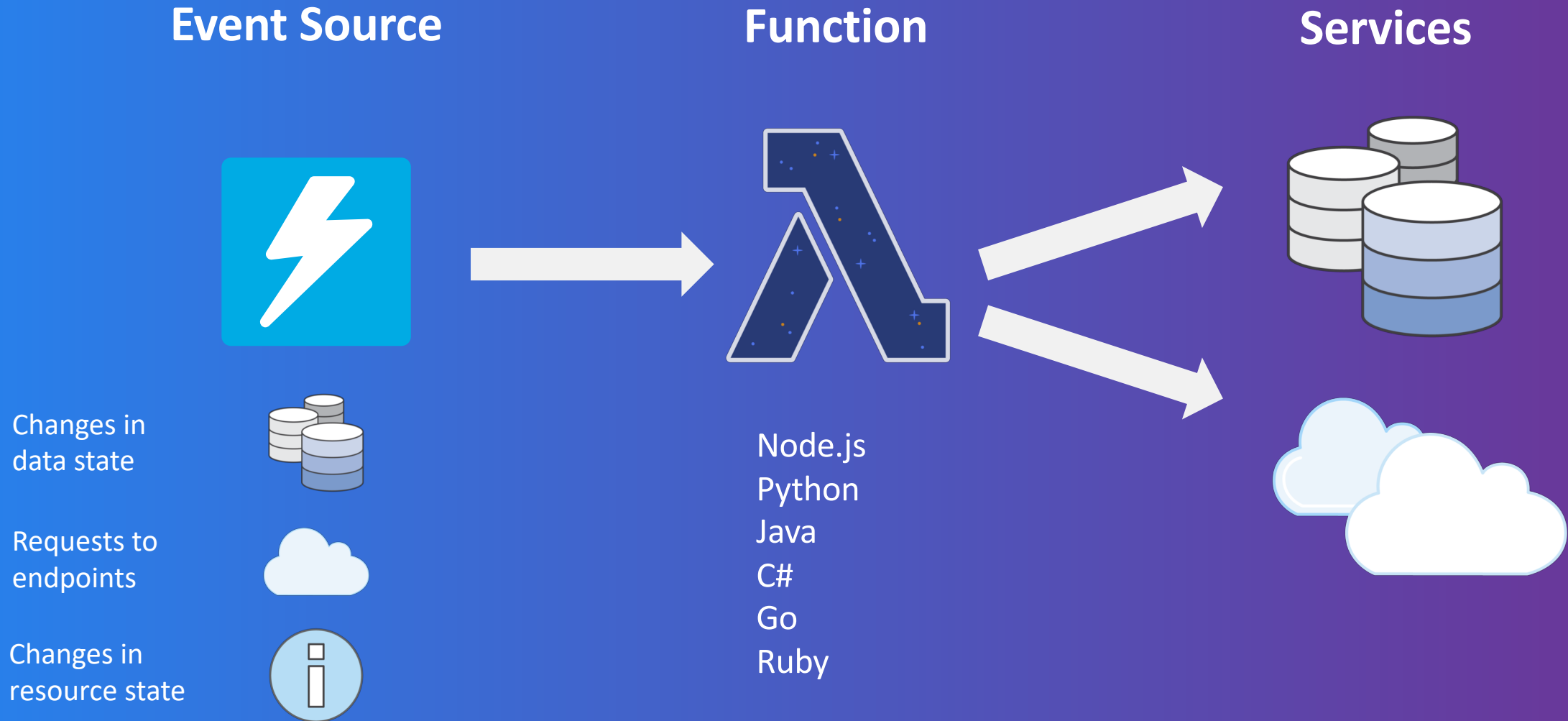
Run code without provisioning
or managing servers

Lambda handles...



- Load balancing
- Auto scaling
- Failure retries
- Security isolation
- OS management
- Utilization and capacity
(and more)

AWS Lambda: Run code in response to events



Serverless is more than just functions



About Autodesk

 dynatrace
Perform



Samy Senthivel

Digital Enterprise
Monitoring Manager



About Autodesk

What do we do?

Autodesk gives you the power to make anything!



How we are doing?



Lambda at Autodesk: Why & How



Josh Toklu

Principal DevOps
Engineer



Autodesk: The move to Lambda

- Where We Started
 - Classic datacenter Architecture
 - Java stack
 - Single monolithic code base
 - Hidden Business logic
- What We Wanted
 - Code driven infrastructure
 - A microservice domain model architecture
 - A move towards CI/CD
 - Multiple repositories for asynchronous development



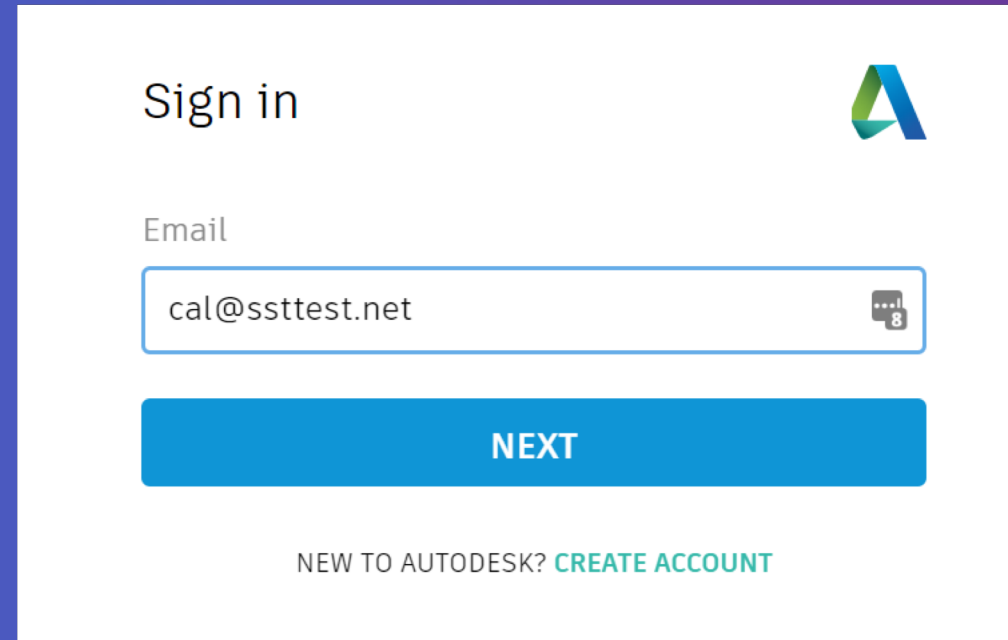
So why AWS Lambda?

- We are a software company first and foremost
 - Scales with demand automatically
 - Significantly reduces server cost
 - Eliminates server maintenance
 - Move our developer time and costs from maintenance to development



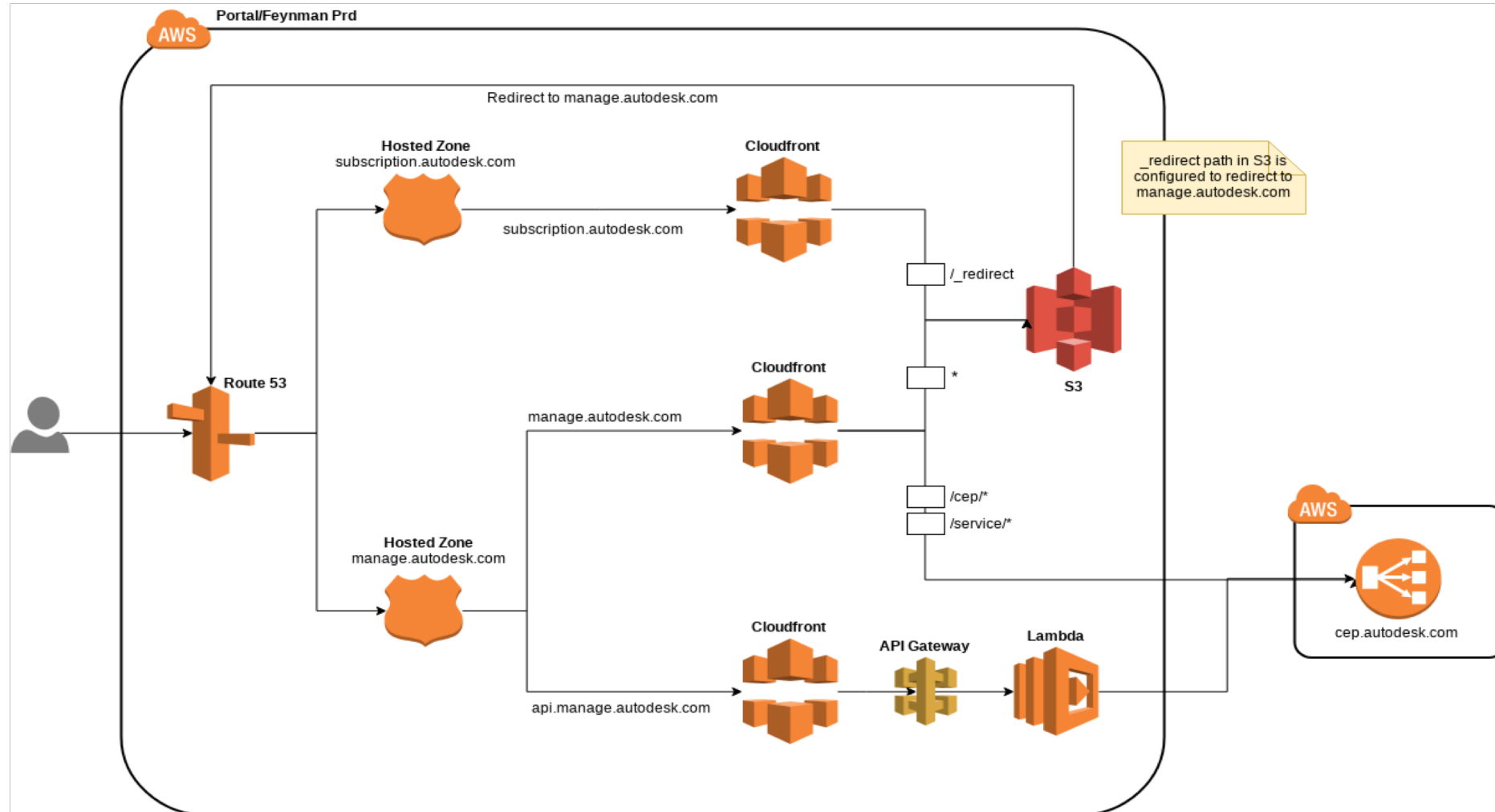
AWS Lambda and the Account Portal (manage.autodesk.com)

- Our Strategy to Migration:
 - Keep the legacy systems running
 - Only re-write code for new API services
 - Split traffic based on route of ingress



The image shows a screenshot of the Autodesk account portal sign-in page. At the top right is the Autodesk logo. Below it, the text "Sign in" is displayed. Underneath is a label "Email" followed by a text input field containing the email address "cal@ssttest.net". To the right of the input field is a small icon of a speech bubble with the number "8". Below the input field is a large blue button with the text "NEXT" in white. At the bottom of the page, there is a link that says "NEW TO AUTODESK? CREATE ACCOUNT".

Our new architecture



Further thoughts

What We Learned:

- Smaller is better. Sometimes it is better to go to a container.
- KISS: Offload your logic to your backend systems behind the API's where you can.
- You can only tweak memory, cpu allocation is directly tied to memory footprint.

Challenges:

- Onboarding and experience can be tricky.
- Test and more test: Understand the spin up behavior prior to production.
- Not having direct access to the process can be tricky to debug





Monitoring Lambda with Dynatrace







Digital Enterprise
Monitoring Manager



Essential Golden metrics for Applications

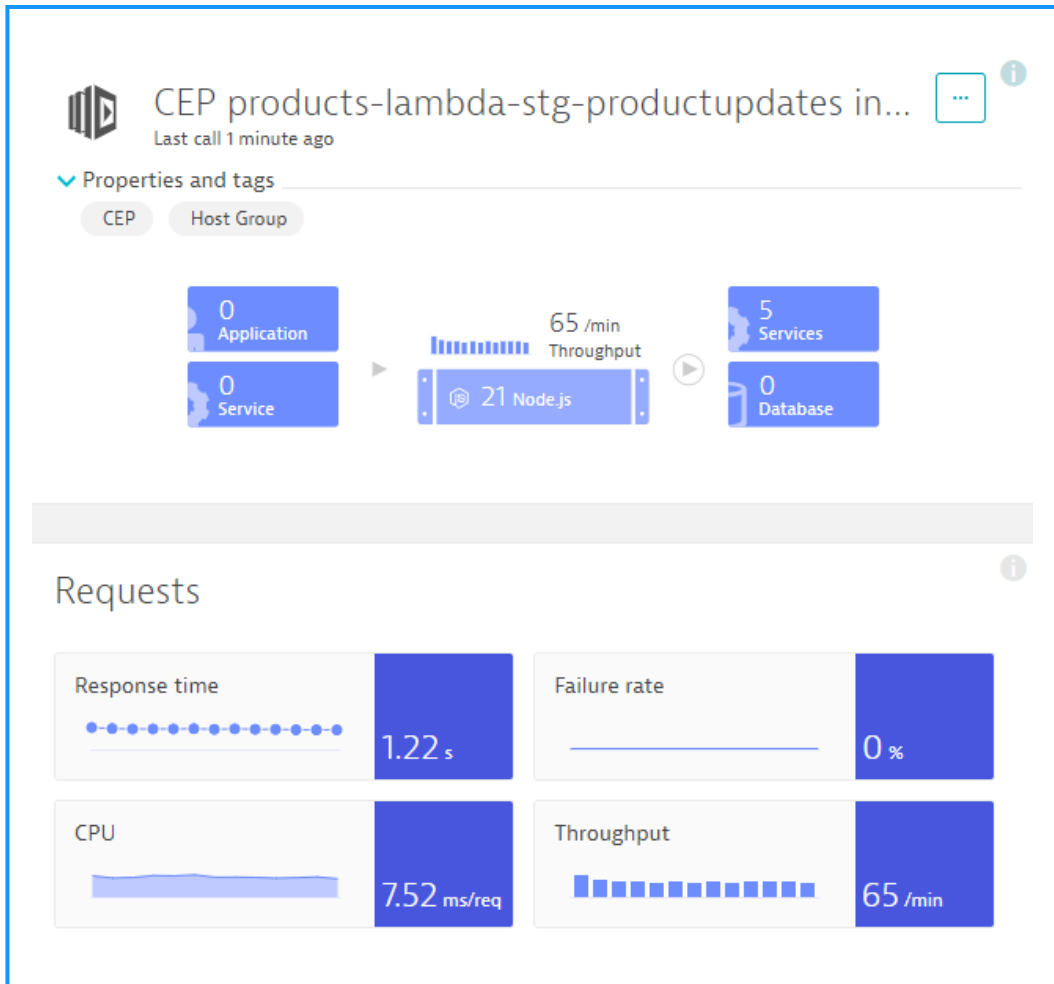
Availability 	Performance 	Error rate 	Utilization 
<p><i>Uptime Percent</i></p> <ul style="list-style-type: none">• Could we respond to the request?	<p><i>Millisecond</i></p> <ul style="list-style-type: none">• How long did it take to execute the function?	<p><i>Percent</i></p> <ul style="list-style-type: none">• # of errors in Invocations	<p><i>Request/min</i></p> <ul style="list-style-type: none">• How many invocations are done?

Dynatrace Lambda Monitoring Journey

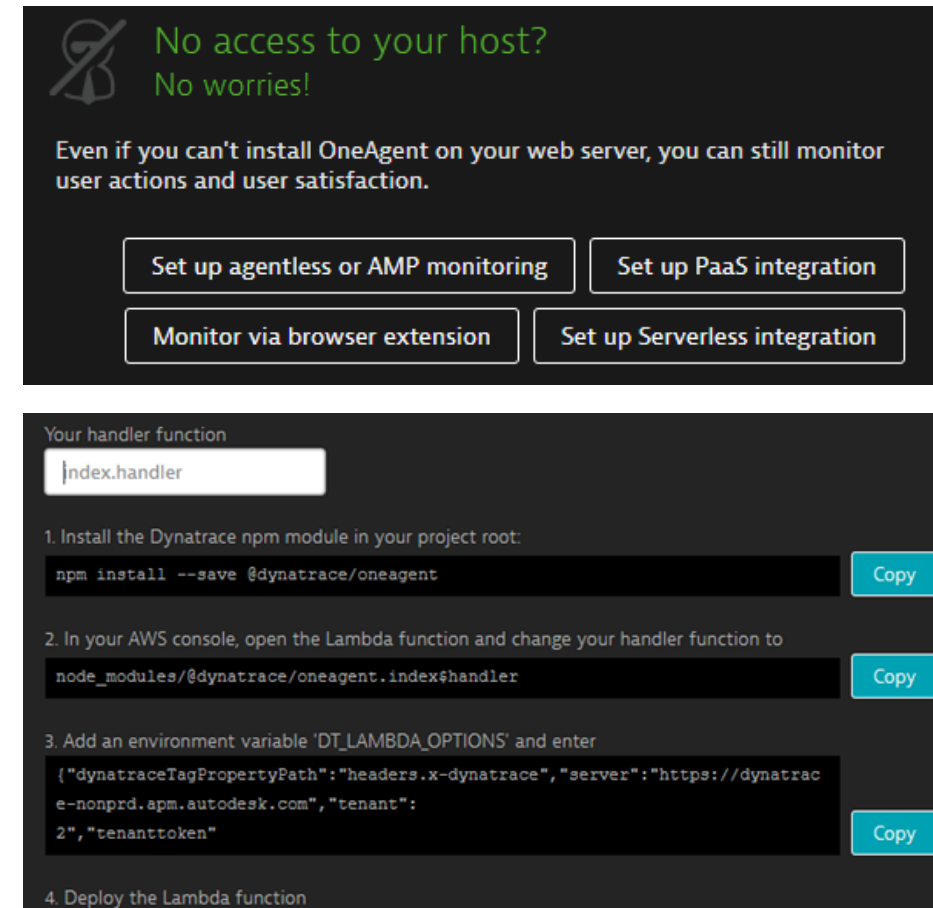
Challenges 	Goals 	Implementations 	Customer Benefits 
<ul style="list-style-type: none"> Resources exist during execution Unlike host we cannot install agents for monitoring Events need to be correlated Collecting tracing logs and metrics is challenging 	<ul style="list-style-type: none"> Ongoing measurement of performance improvement during updates and changes as we move to AWS Lambda Able to identify and resolve issues faster 	<ul style="list-style-type: none"> Instrumenting lambda code with handler provide by Dynatrace Lambda starts reporting the metrics data to Dynatrace Dynatrace collects, aggregates and baselines the metrics for alerting and dashboarding 	<ul style="list-style-type: none"> Full Stack Visibility Event correlation across all dependent services with Service flow Dynatrace AI Engine helps in anomaly detection


Dynatrace + Lambda Monitoring

- Out-of-the-box, Dynatrace provides full set of AWS Cloudwatch metrics for lambda through its AWS integration.
- Through serverless integration Dynatrace additionally provides
 - End-to-end tracing to and from lambda functions, without changing your code
 - AI powered problem detection
 - Response time
 - Failure rates
 - Process metrics
 - Event loop metrics (for node.js)



- Log in to Dynatrace environment and click on Deploy Dynatrace → Set up serverless integration.
- Follow the three steps deploy process
 - Install the Dynatrace npm module (the module comes with a helper utility that strips down the module to a specific node.js version)
 - Change the handler function in AWS console to the one given in the Dynatrace console.
 - Add an environment variable for DT_LAMBDA_OPTIONS
 - And deploy



 **No access to your host?**
No worries!

Even if you can't install OneAgent on your web server, you can still monitor user actions and user satisfaction.

[Set up agentless or AMP monitoring](#) [Set up PaaS integration](#)

[Monitor via browser extension](#) [Set up Serverless integration](#)

Your handler function

1. Install the Dynatrace npm module in your project root:

```
npm install --save @dynatrace/oneagent
```

[Copy](#)

2. In your AWS console, open the Lambda function and change your handler function to

```
node_modules/@dynatrace/oneagent.index$handler
```

[Copy](#)

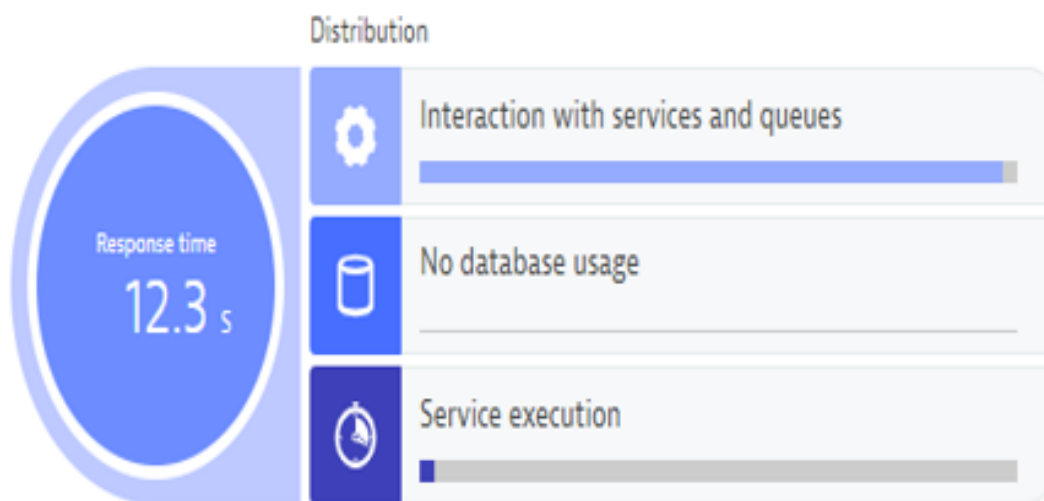
3. Add an environment variable 'DT_LAMBDA_OPTIONS' and enter

```
{ "dynatraceTagPropertyPath": "headers.x-dynatrace", "server": "https://dynatrac  
e-nonprd.apm.autodesk.com", "tenant":  
2", "tenanttoken"
```

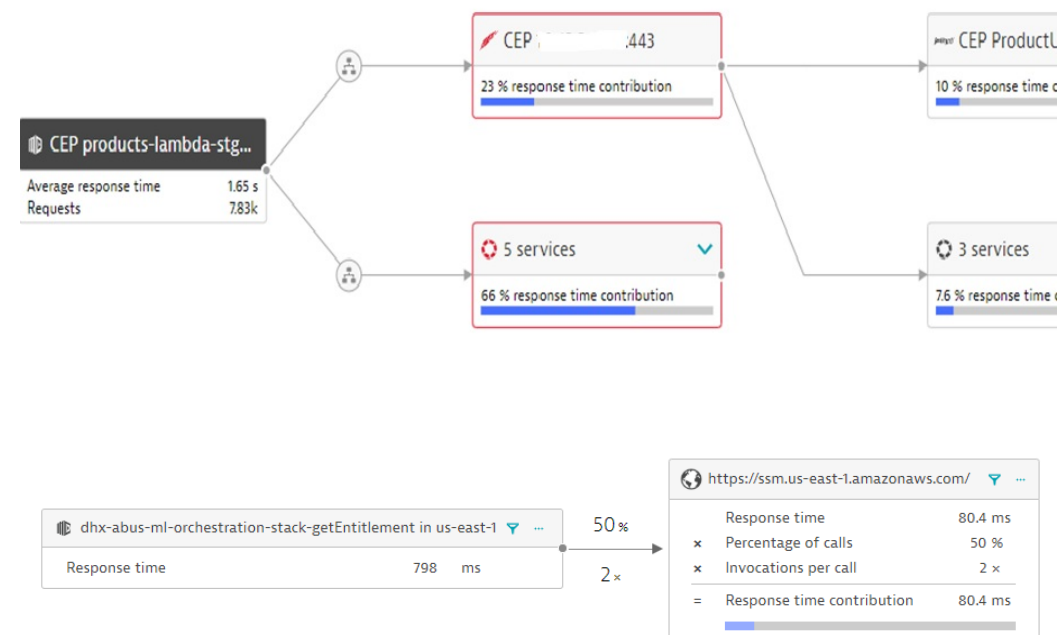
[Copy](#)

4. Deploy the Lambda function

Response time



Service Flow



50% of the requests call <https://ssm.us-east-1.amazonaws.com/> (averaging 2 calls per request).

Purepath Deepdive

Purepath

'CEP products-lambda-stg-productupdates in us-west-2' PurePath

Start time
2019 January 23 08:58:38



Response time
12.3 s

Total processing time
1.36 s








Breakdown of PurePath processing time

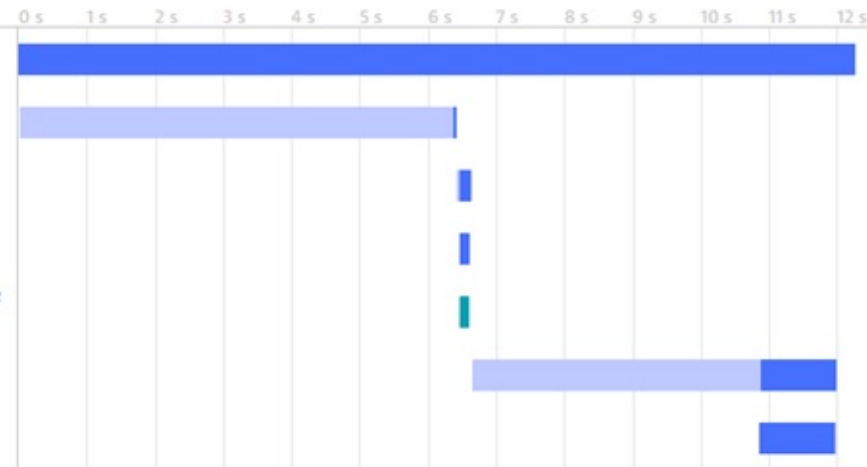


Top findings

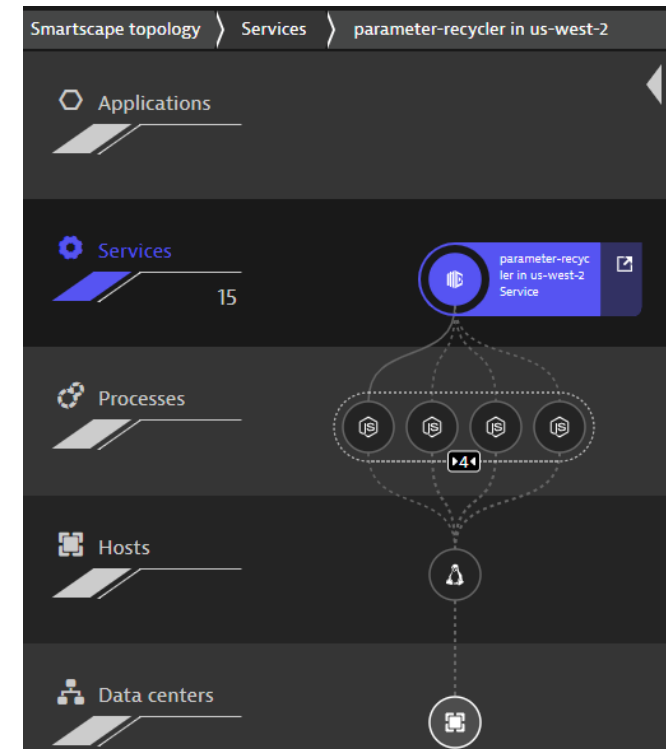
-  2 databases consuming 4.96 ms
-  1 third party resource consuming 152 ms

Search name, url, sql, attribute,...

- ✓  invoke
CEP products-lambda-stg-productupdates in us-west-2
- >  /service/entitlement/v2/primary-entitlements
CEP 443
- ✓  /service/user/loggedin-profile
CEP 43
- ✓  getLoggedInProfile
CEP UserRestApi
-  https://accounts-staging.autodesk.com/api/accounts/v1/user/P6DAJ36F922F
Requests to public networks
- ✓  /service/product-updates/getProductUpdates.json
CEP 443
- ✓  getProductUpdates
CEP ProductUpdateServiceImpl



Smartscape View



Dynatrace + Lambda Quick Wins



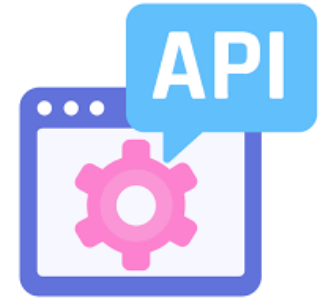
Able to instrument our applications with very little effort.



Able to quickly find and address performance concerns



Able to see metrics



Visualize the Service call graph.

Closing remarks

Daniel Khan
Technical Product Manager - Dynatrace

Further resources

- **Apply to our EAP**
bit.ly/lambda-eap
- **Dynatrace Documentation for Lambda**
bit.ly/dynatrace-doc-lambda
- **Blog-post on deploying Dynatrace with Lambda in 8 Minutes**
bit.ly/dynatrace-deploy-lambda
- **Dynatrace OneAgent for Serverless Module**
bit.ly/dynatrace-serverless
- **Questions? Reach out!**
daniel.khan@dynatrace.com
or meet me at the Innovation Center

Let us know how we did!

- 2 minute survey
- Find it from the Perform app menu
- Complete survey for each breakout you attend

Track = Cloud Ops

