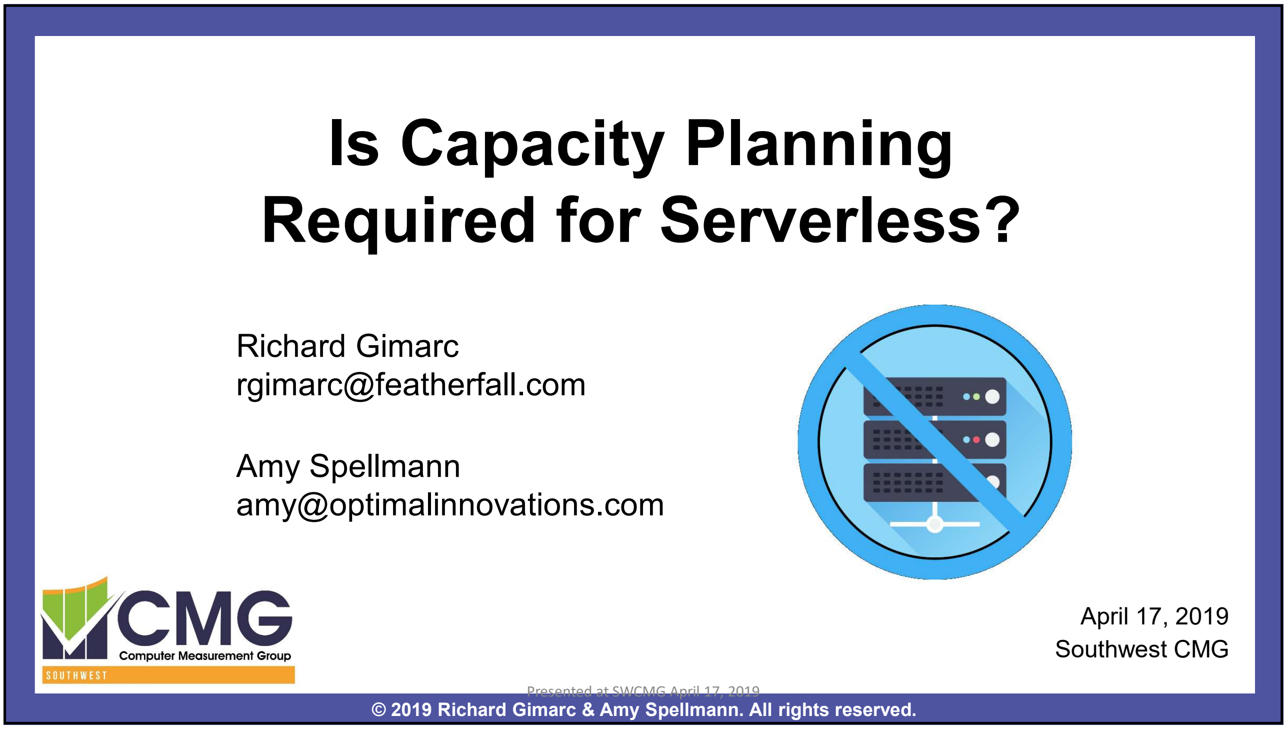


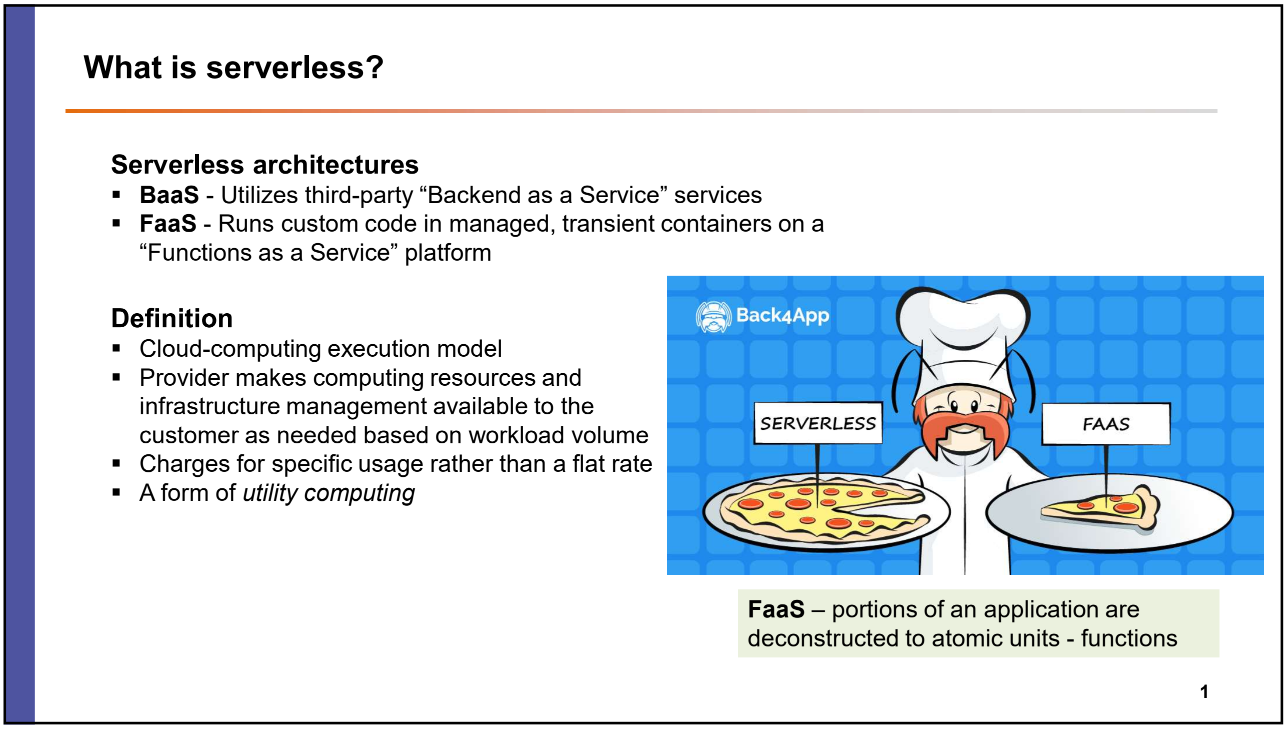
Is Capacity Planning Required for Serverless?

Richard Gimarc
rgimarc@featherfall.com

Amy Spellmann
amy@optimalinnovations.com

April 17, 2019
Southwest CMG

Presented at SWCMG April 17, 2019



What is serverless?

Serverless architectures

- **BaaS** - Utilizes third-party "Backend as a Service" services
- **FaaS** - Runs custom code in managed, transient containers on a "Functions as a Service" platform

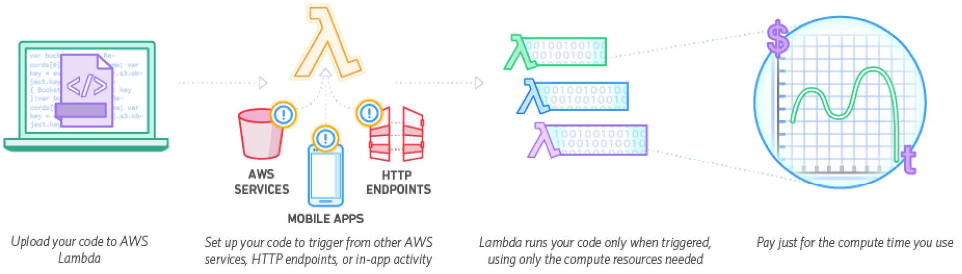
Definition

- Cloud-computing execution model
- Provider makes computing resources and infrastructure management available to the customer as needed based on workload volume
- Charges for specific usage rather than a flat rate
- A form of *utility computing*

FaaS – portions of an application are deconstructed to atomic units - functions

1

Advantages of serverless


AWS SERVICES
MOBILE APPS
HTTP ENDPOINTS
$
t
Upload your code to AWS Lambda
Set up your code to trigger from other AWS services, HTTP endpoints, or in-app activity
Lambda runs your code only when triggered, using only the compute resources needed
Pay just for the compute time you use

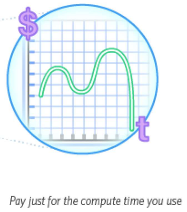
- Goal is to maximize the efficient use of resources and/or minimize associated costs
- Change in pricing model: you now pay by function call (like a utility)
- Advantage of a low or no initial cost to acquire computer resources - resources are essentially rented
- No cost for idle capacity
- FaaS automatically scales up/down based on workload volume

Image Ref:
[DEV2018]

Why go serverless?

Time vs Money & Business vs Developers

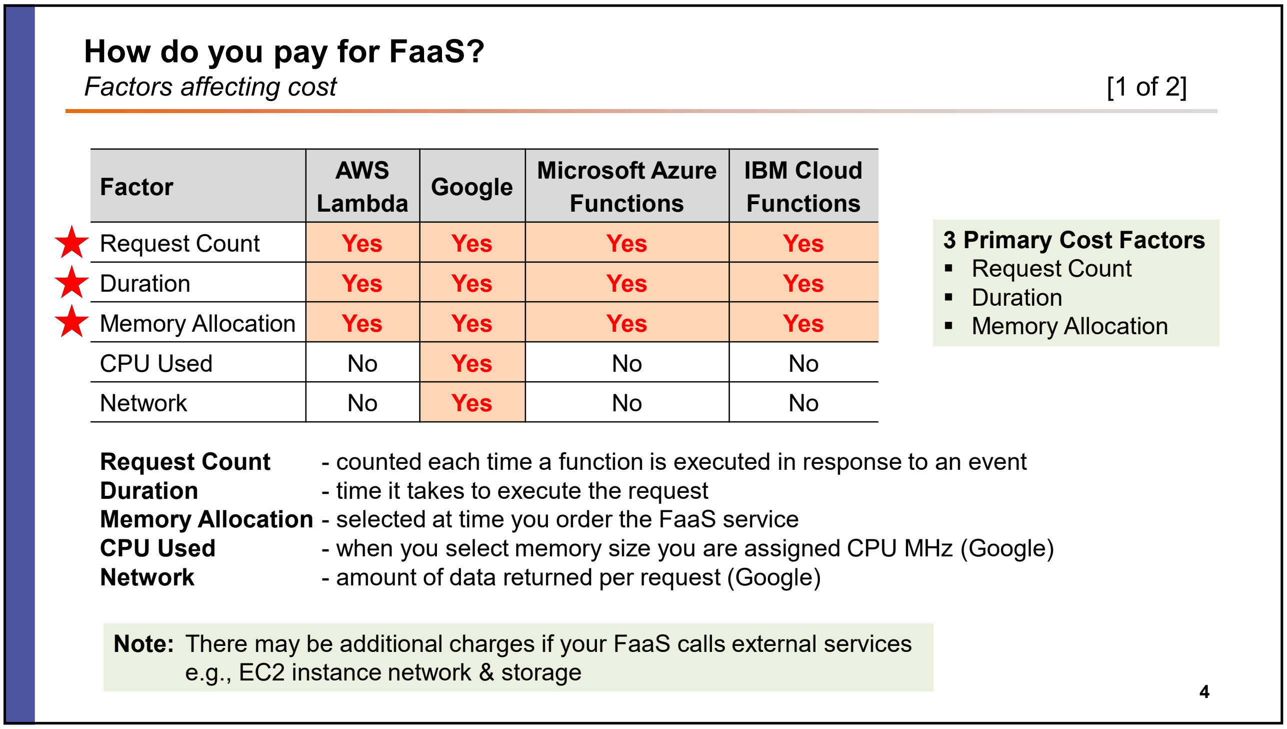
Save time	▪ Less concern/planning for application scalability ▪ No need to deal with implementation upon deployment ▪ Spend time on further innovation rather than dealing with the infrastructure ▪ Third party responsible for monitoring & scaling infrastructure
Save money	▪ When you pay-per-trigger, you don't need to plan reservations or plan for spikes ▪ You just pay for what you use



REF: [SHC2018]

Business	▪ No longer paying for resources that are not used
Developers	▪ Microservices architecture move away from monolithic applications to focus on simple logic that does one thing (and one thing only) ▪ Simplicity – each FaaS is a isolated piece of logic ▪ Less concern about scaling, monitoring and other activities associated with traditional servers

REF: [ALA2017]



How do you pay for FaaS?

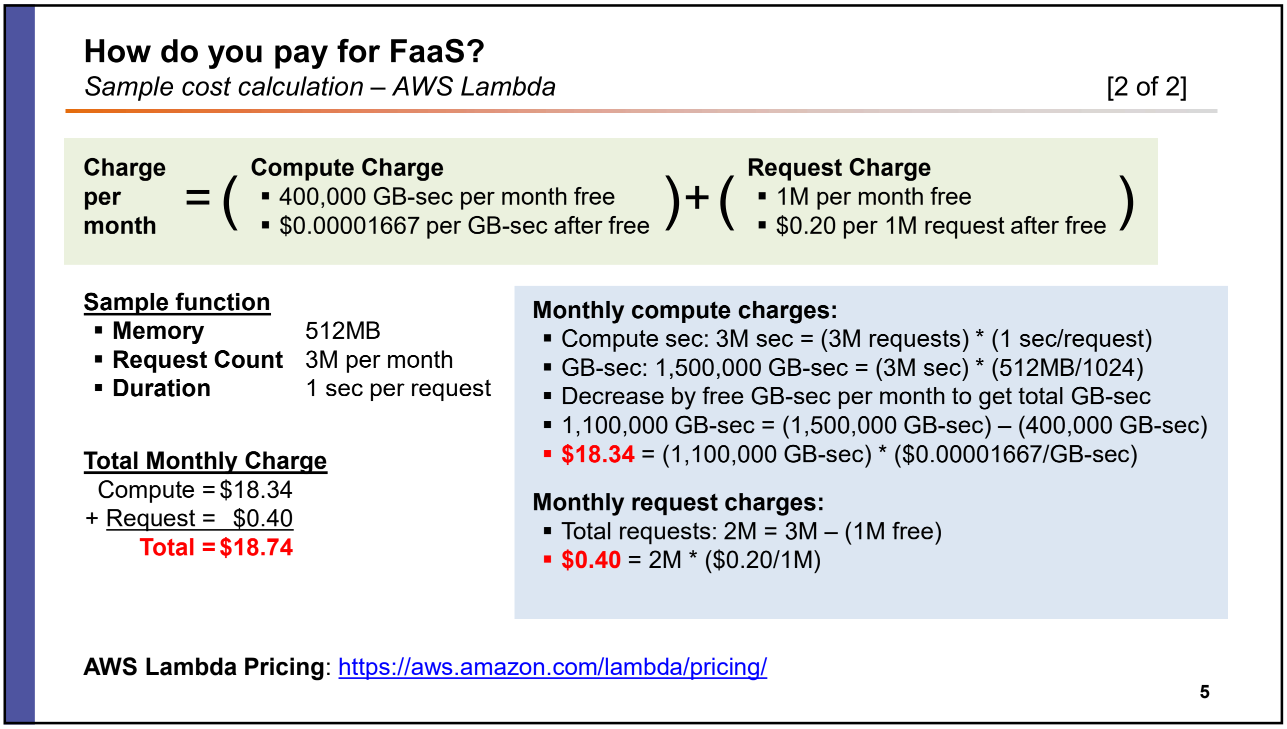
Factors affecting cost [1 of 2]

	Factor	AWS Lambda	Google	Microsoft Azure Functions	IBM Cloud Functions
★	Request Count	Yes	Yes	Yes	Yes
★	Duration	Yes	Yes	Yes	Yes
★	Memory Allocation	Yes	Yes	Yes	Yes
	CPU Used	No	Yes	No	No
	Network	No	Yes	No	No

3 Primary Cost Factors
- Request Count
- Duration
- Memory Allocation

Request Count - counted each time a function is executed in response to an event
Duration - time it takes to execute the request
Memory Allocation - selected at time you order the FaaS service
CPU Used - when you select memory size you are assigned CPU MHz (Google)
Network - amount of data returned per request (Google)

Note: There may be additional charges if your FaaS calls external services e.g., EC2 instance network & storage



How do you pay for FaaS?

Sample cost calculation – AWS Lambda [2 of 2]

Charge per month = (**Compute Charge**
- 400,000 GB-sec per month free
- $0.00001667 per GB-sec after free

) + (**Request Charge**
- 1M per month free
- $0.20 per 1M request after free

)

Sample function

Memory	512MB
Request Count	3M per month
Duration	1 sec per request

Total Monthly Charge

Compute =	$18.34
+ Request =	$0.40
Total =	**$18.74**

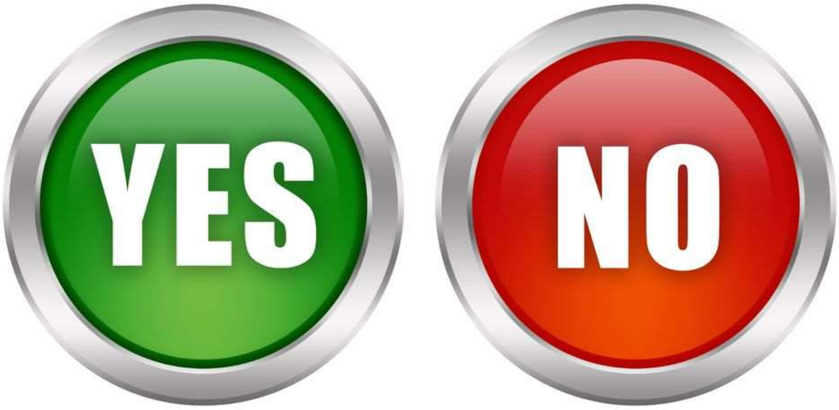
Monthly compute charges:
- Compute sec: 3M sec = (3M requests) * (1 sec/request)
- GB-sec: 1,500,000 GB-sec = (3M sec) * (512MB/1024)
- Decrease by free GB-sec per month to get total GB-sec
- 1,100,000 GB-sec = (1,500,000 GB-sec) – (400,000 GB-sec)
- **$18.34** = (1,100,000 GB-sec) * ($0.00001667/GB-sec)

Monthly request charges:
- Total requests: 2M = 3M – (1M free)
- **$0.40** = 2M * ($0.20/1M)

AWS Lambda Pricing: https://aws.amazon.com/lambda/pricing/

Is Capacity Planning Required for Serverless?

Time to vote!

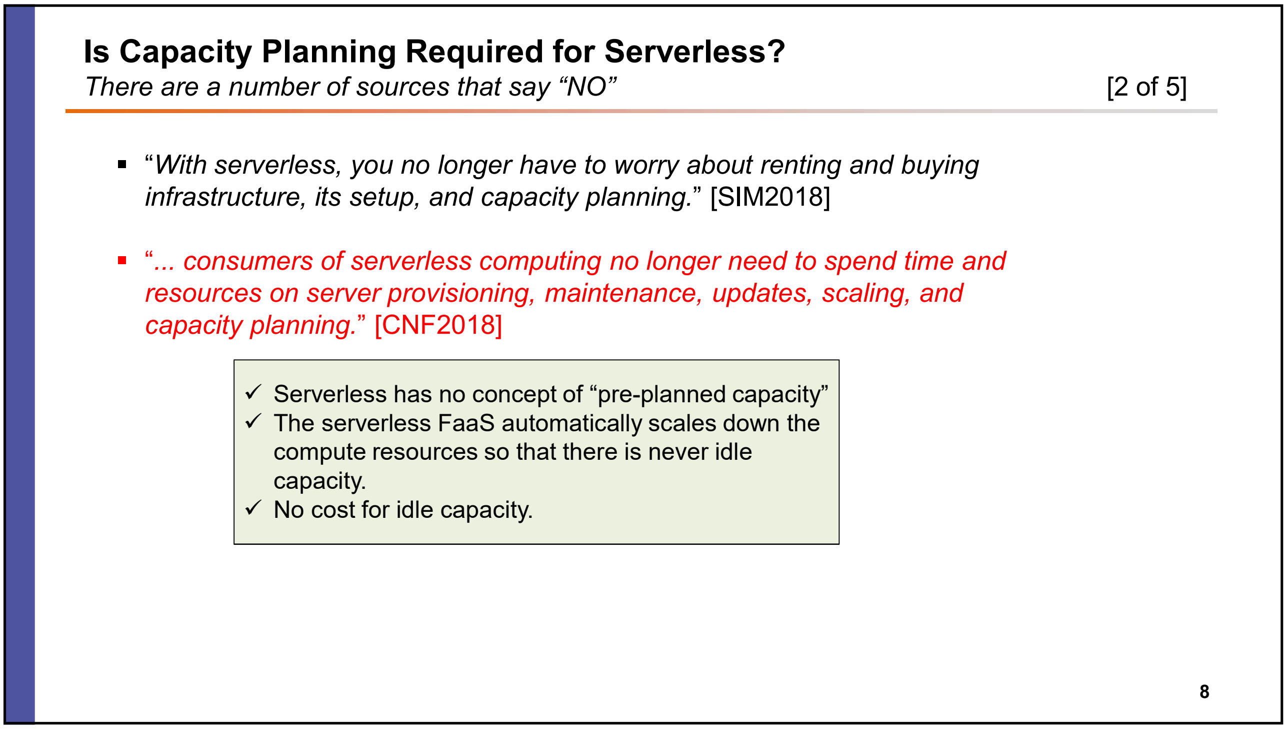


Is Capacity Planning Required for Serverless?

There are a number of sources that say "NO" [1 of 5]

- *"With serverless, you no longer have to worry about renting and buying infrastructure, its setup, and capacity planning."* [SIM2018]

✓ Responsibility for these tasks is with the serverless platform provider.
✓ You "only" have to think about how to properly plan, design and develop your applications.



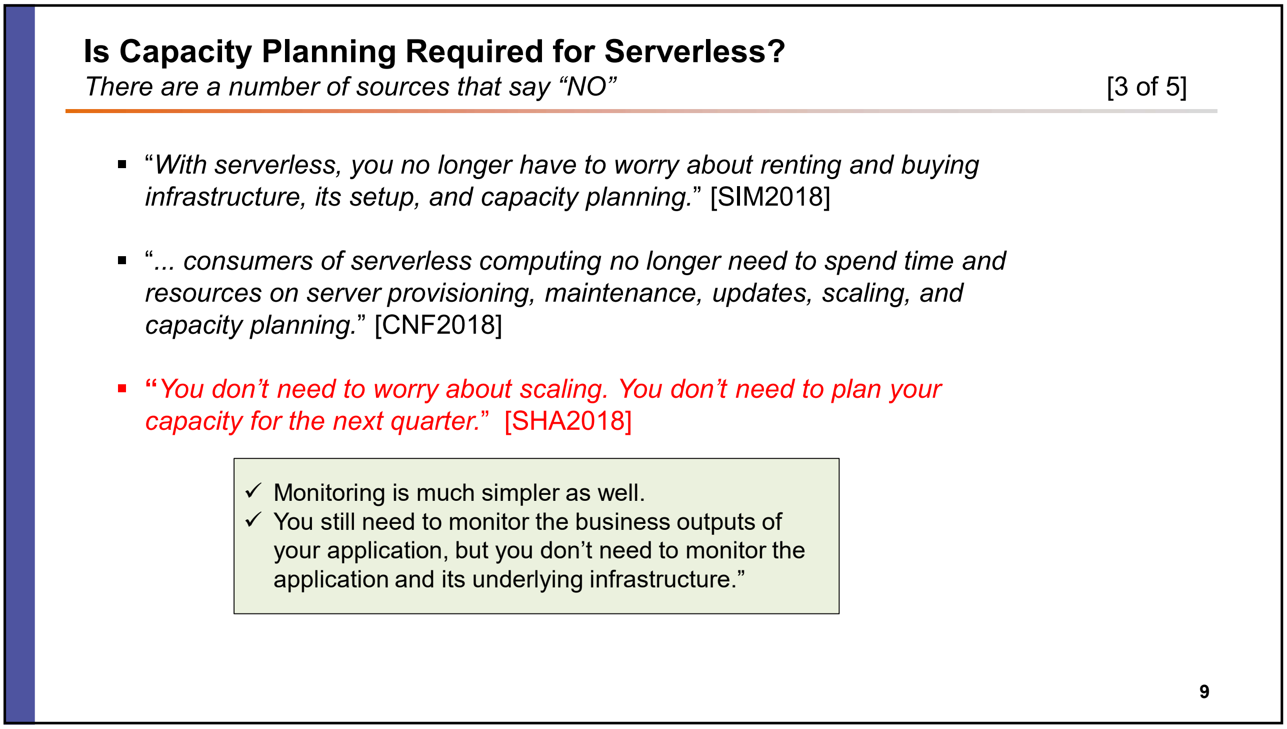
Is Capacity Planning Required for Serverless?

There are a number of sources that say "NO" [2 of 5]

- "*With serverless, you no longer have to worry about renting and buying infrastructure, its setup, and capacity planning.*" [SIM2018]
- "*... consumers of serverless computing no longer need to spend time and resources on server provisioning, maintenance, updates, scaling, and capacity planning.*" [CNF2018]

✓ Serverless has no concept of "pre-planned capacity"
✓ The serverless FaaS automatically scales down the compute resources so that there is never idle capacity.
✓ No cost for idle capacity.

8



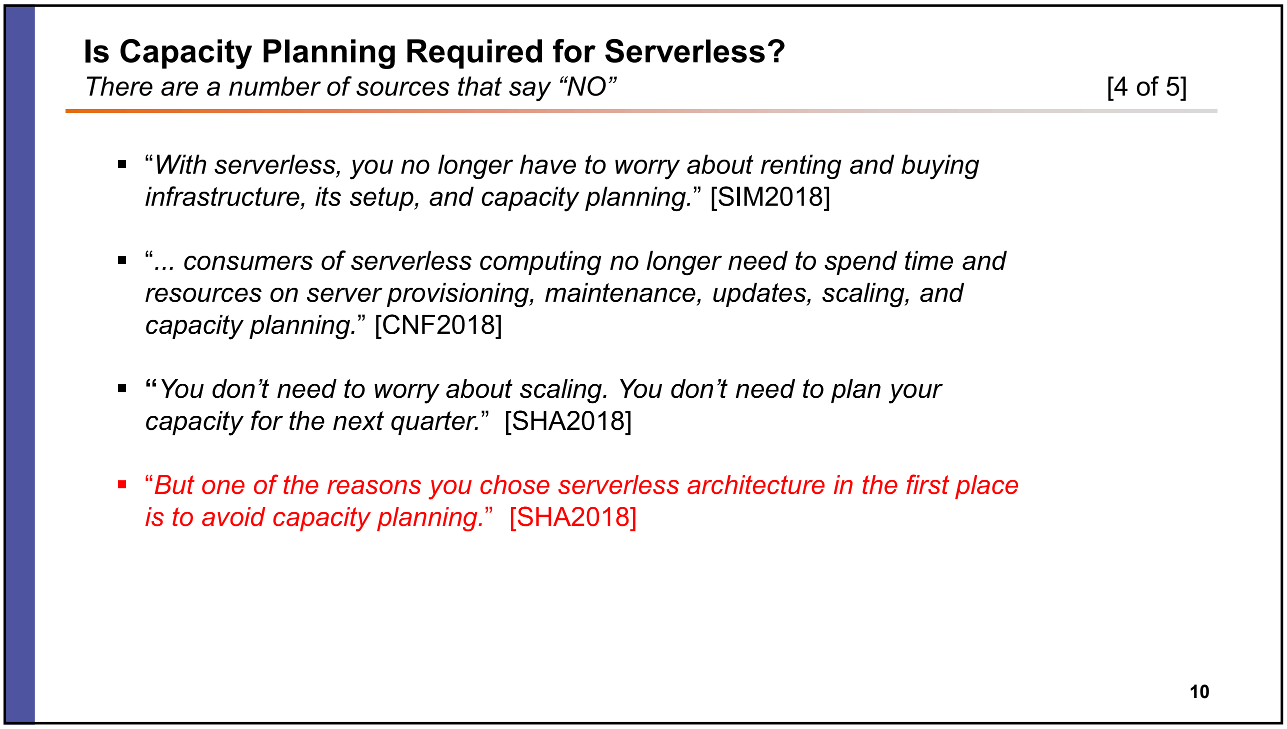
Is Capacity Planning Required for Serverless?

There are a number of sources that say "NO" [3 of 5]

- "*With serverless, you no longer have to worry about renting and buying infrastructure, its setup, and capacity planning.*" [SIM2018]
- "*... consumers of serverless computing no longer need to spend time and resources on server provisioning, maintenance, updates, scaling, and capacity planning.*" [CNF2018]
- "*You don't need to worry about scaling. You don't need to plan your capacity for the next quarter.*" [SHA2018]

✓ Monitoring is much simpler as well.
✓ You still need to monitor the business outputs of your application, but you don't need to monitor the application and its underlying infrastructure."

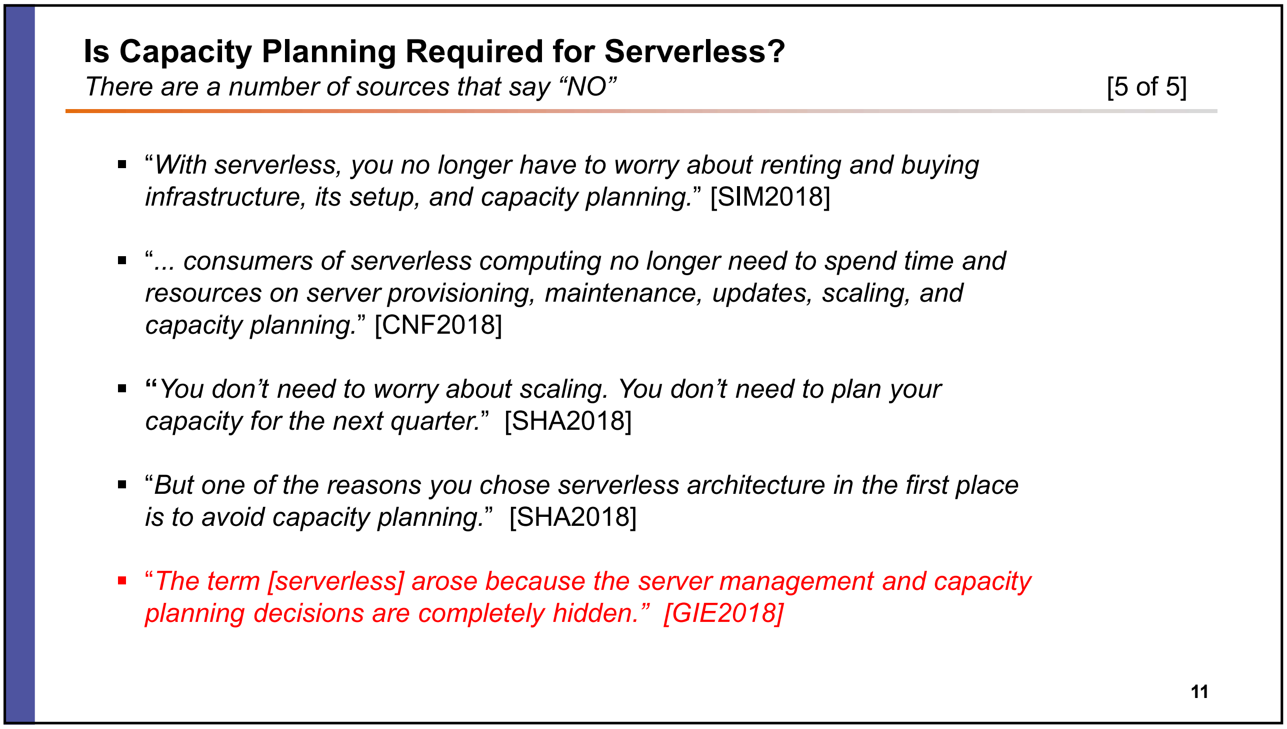
9



Is Capacity Planning Required for Serverless?

There are a number of sources that say “NO” [4 of 5]

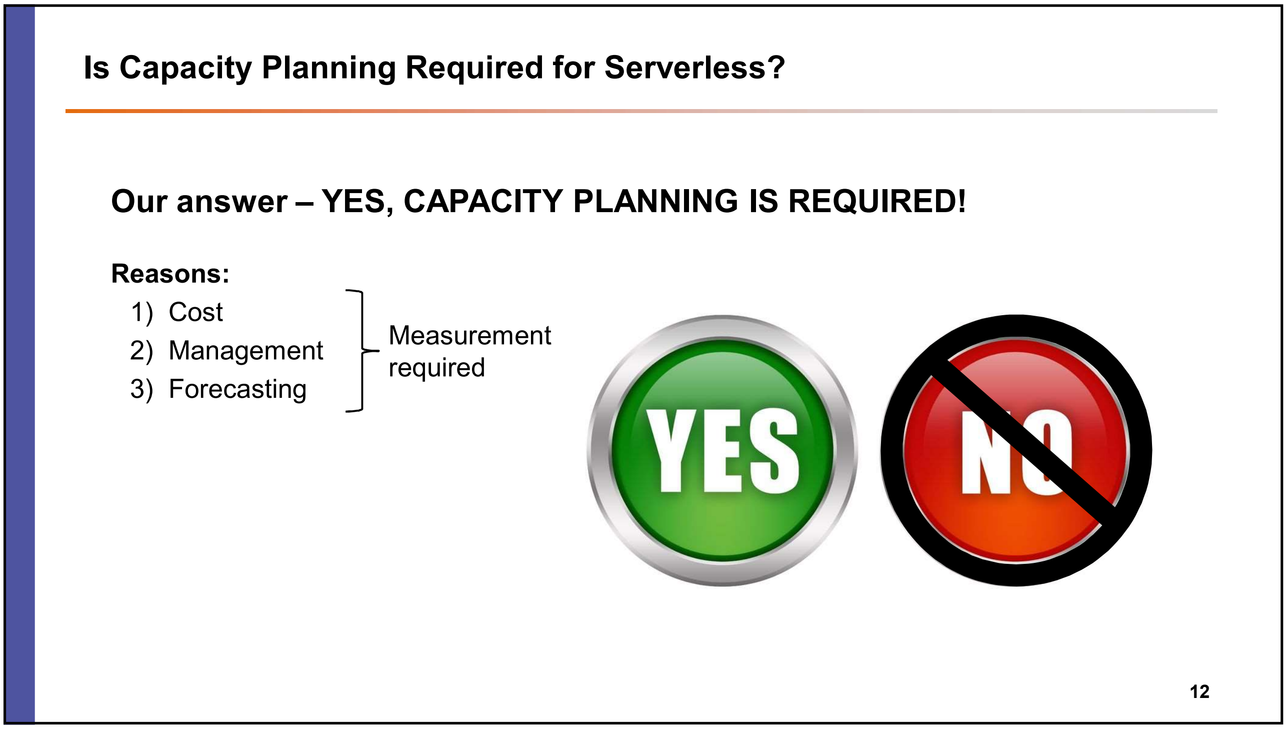
- “*With serverless, you no longer have to worry about renting and buying infrastructure, its setup, and capacity planning.*” [SIM2018]
- “*... consumers of serverless computing no longer need to spend time and resources on server provisioning, maintenance, updates, scaling, and capacity planning.*” [CNF2018]
- “*You don’t need to worry about scaling. You don’t need to plan your capacity for the next quarter.*” [SHA2018]
- “*But one of the reasons you chose serverless architecture in the first place is to avoid capacity planning.*” [SHA2018]



Is Capacity Planning Required for Serverless?

There are a number of sources that say “NO” [5 of 5]

- “*With serverless, you no longer have to worry about renting and buying infrastructure, its setup, and capacity planning.*” [SIM2018]
- “*... consumers of serverless computing no longer need to spend time and resources on server provisioning, maintenance, updates, scaling, and capacity planning.*” [CNF2018]
- “*You don’t need to worry about scaling. You don’t need to plan your capacity for the next quarter.*” [SHA2018]
- “*But one of the reasons you chose serverless architecture in the first place is to avoid capacity planning.*” [SHA2018]
- “*The term [serverless] arose because the server management and capacity planning decisions are completely hidden.” [GIE2018]*



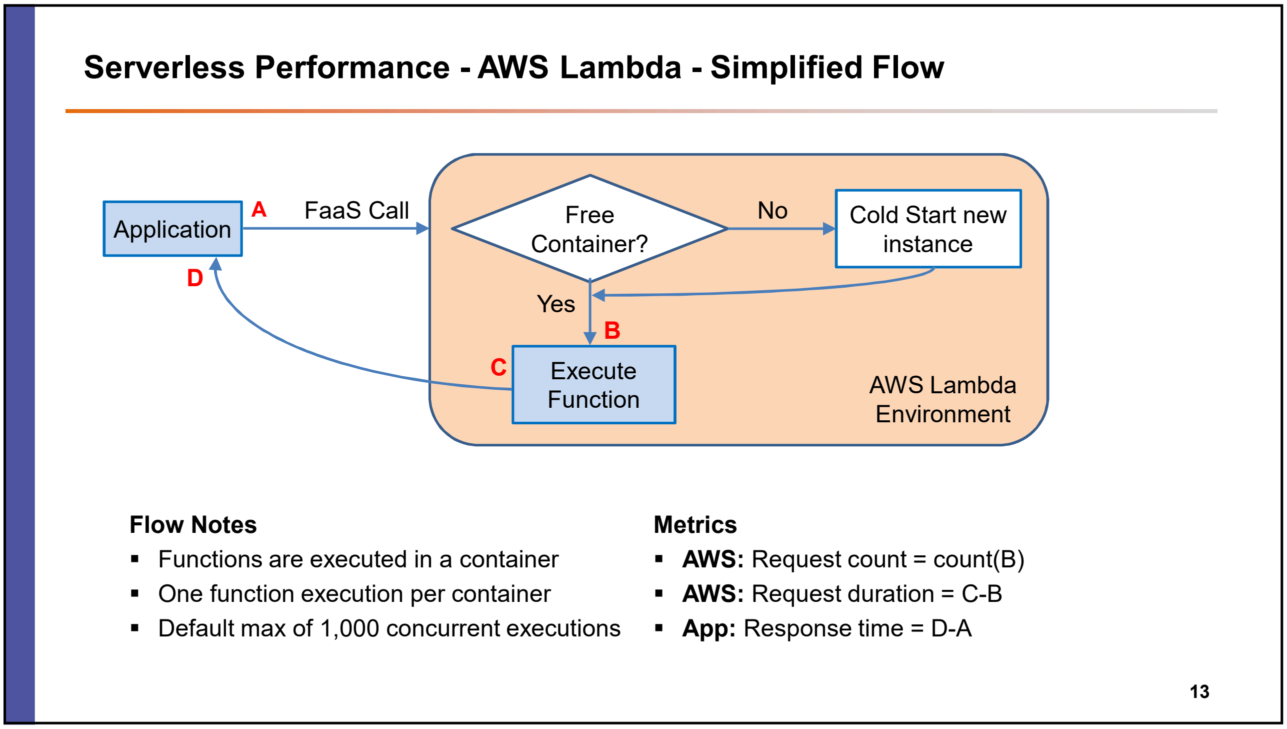
Is Capacity Planning Required for Serverless?

Our answer – YES, CAPACITY PLANNING IS REQUIRED!

Reasons:

1) Cost
2) Management
3) Forecasting

Measurement required



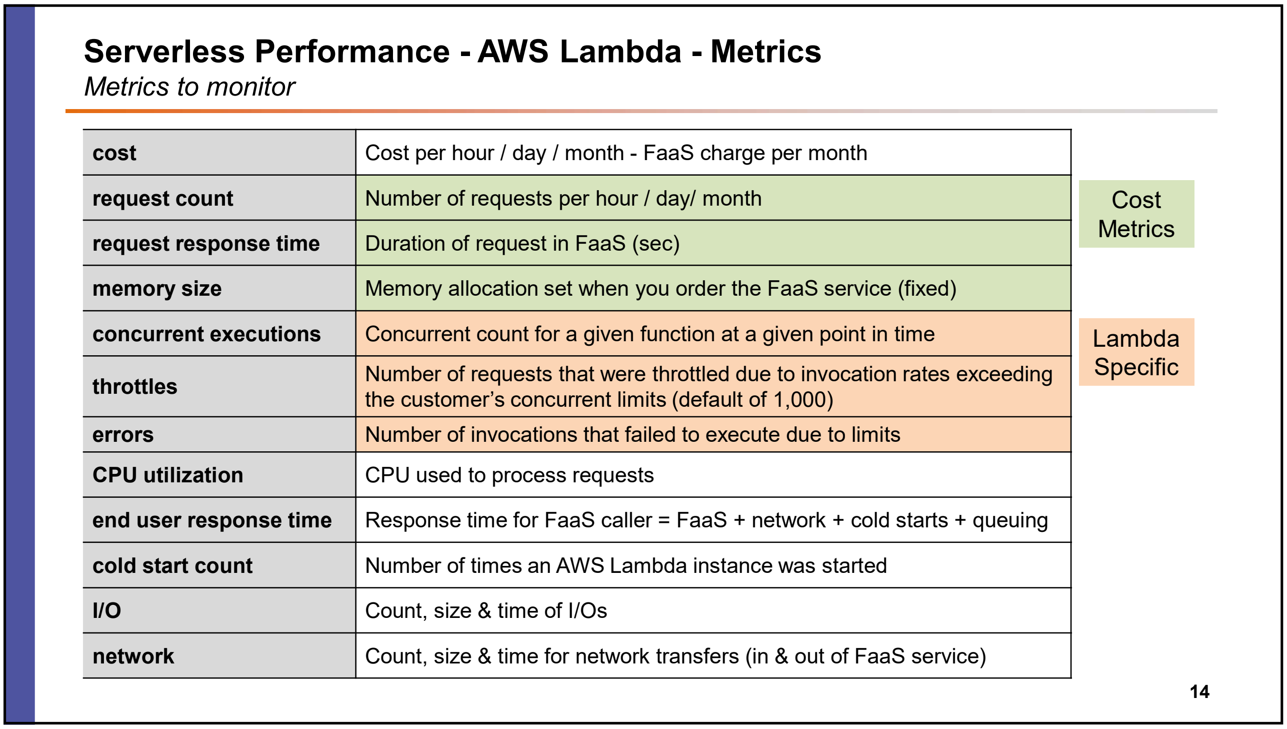
Serverless Performance - AWS Lambda - Simplified Flow

Flow Notes

- Functions are executed in a container
- One function execution per container
- Default max of 1,000 concurrent executions

Metrics

- **AWS:** Request count = count(B)
- **AWS:** Request duration = C-B
- **App:** Response time = D-A



Serverless Performance - AWS Lambda - Metrics

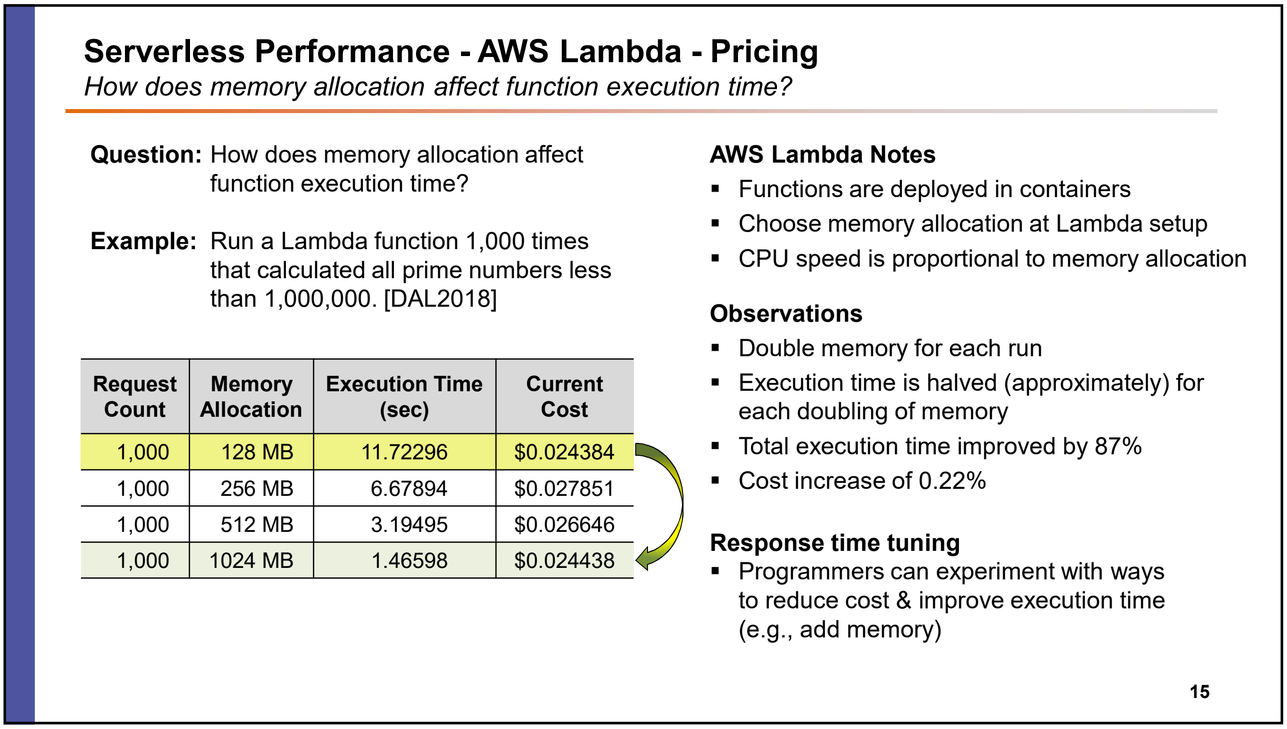
Metrics to monitor

cost	Cost per hour / day / month - FaaS charge per month
request count	Number of requests per hour / day/ month
request response time	Duration of request in FaaS (sec)
memory size	Memory allocation set when you order the FaaS service (fixed)
concurrent executions	Concurrent count for a given function at a given point in time
throttles	Number of requests that were throttled due to invocation rates exceeding the customer's concurrent limits (default of 1,000)
errors	Number of invocations that failed to execute due to limits
CPU utilization	CPU used to process requests
end user response time	Response time for FaaS caller = FaaS + network + cold starts + queuing
cold start count	Number of times an AWS Lambda instance was started
I/O	Count, size & time of I/Os
network	Count, size & time for network transfers (in & out of FaaS service)

Cost Metrics

Lambda Specific

14



Serverless Performance - AWS Lambda - Pricing

How does memory allocation affect function execution time?

Question: How does memory allocation affect function execution time?

Example: Run a Lambda function 1,000 times that calculated all prime numbers less than 1,000,000. [DAL2018]

Request Count	Memory Allocation	Execution Time (sec)	Current Cost
1,000	128 MB	11.72296	$0.024384
1,000	256 MB	6.67894	$0.027851
1,000	512 MB	3.19495	$0.026646
1,000	1024 MB	1.46598	$0.024438

AWS Lambda Notes

- Functions are deployed in containers
- Choose memory allocation at Lambda setup
- CPU speed is proportional to memory allocation

Observations

- Double memory for each run
- Execution time is halved (approximately) for each doubling of memory
- Total execution time improved by 87%
- Cost increase of 0.22%

Response time tuning

- Programmers can experiment with ways to reduce cost & improve execution time (e.g., add memory)

15

Serverless Sprawl

[1 of 2]

“*Serverless functions are like tribbles. They start out small and cute, but then they proliferate, and you end up neck-deep in them. Suddenly, what was meant to be simple is simple no longer.*”

[CHE2018]

Star Trek
“*The Trouble With Tribbles*”
S2 E15
Dec 29, 1967

16

Serverless Sprawl

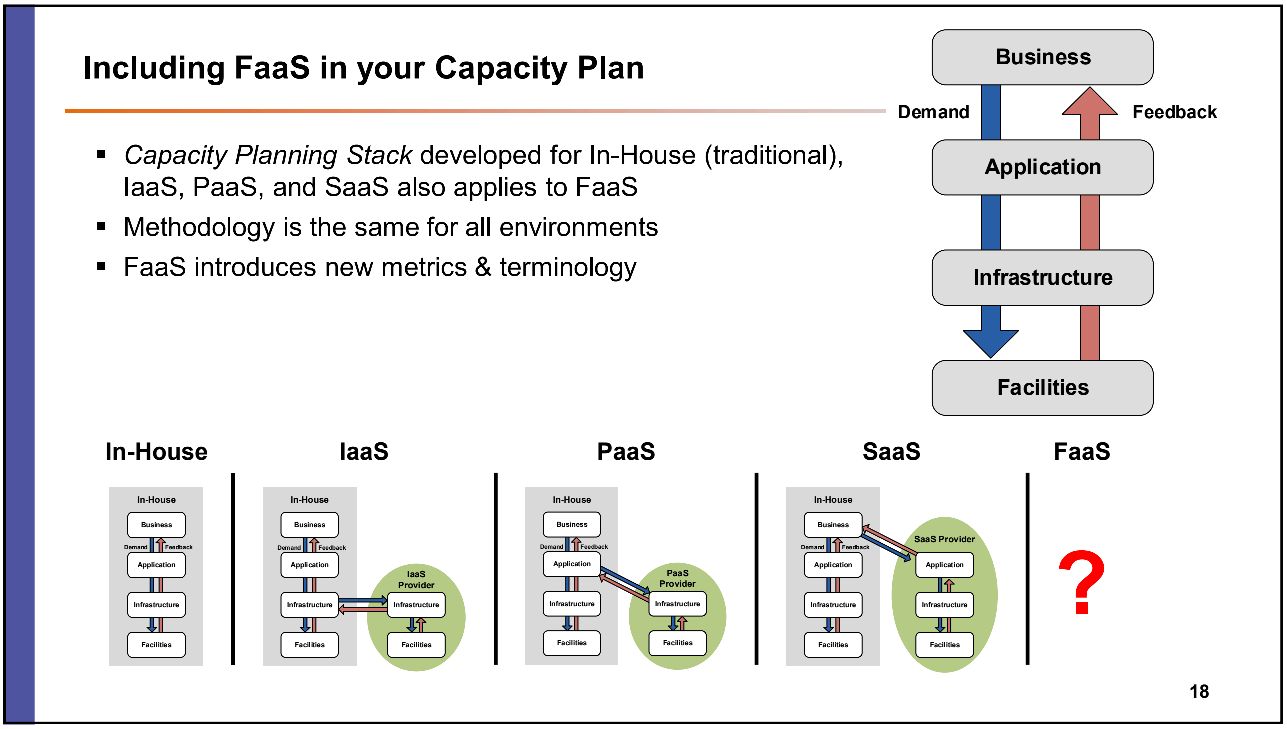
[2 of 2]

- **Where do you see sprawl?**
  - Development – creating new instances every hour/day without regard to cost
  - Production – FaaS reacting to offered workload
- **What causes production sprawl?**
  - Workload arrival rate & pattern (sequential vs parallel vs bulk)
  - Request duration
- **What do you monitor?**
  - cost
  - memory size
  - request count
  - duration

Prerequisites to address sprawl [HUB2017]

- **Inventory** – up to date inventory of FaaS components
- **Relationships** – describe end-to-end relationships of FaaS components to the higher level business service
- **Report** – smart alerts are required, but you also need proactive reporting that keeps you ahead of problems

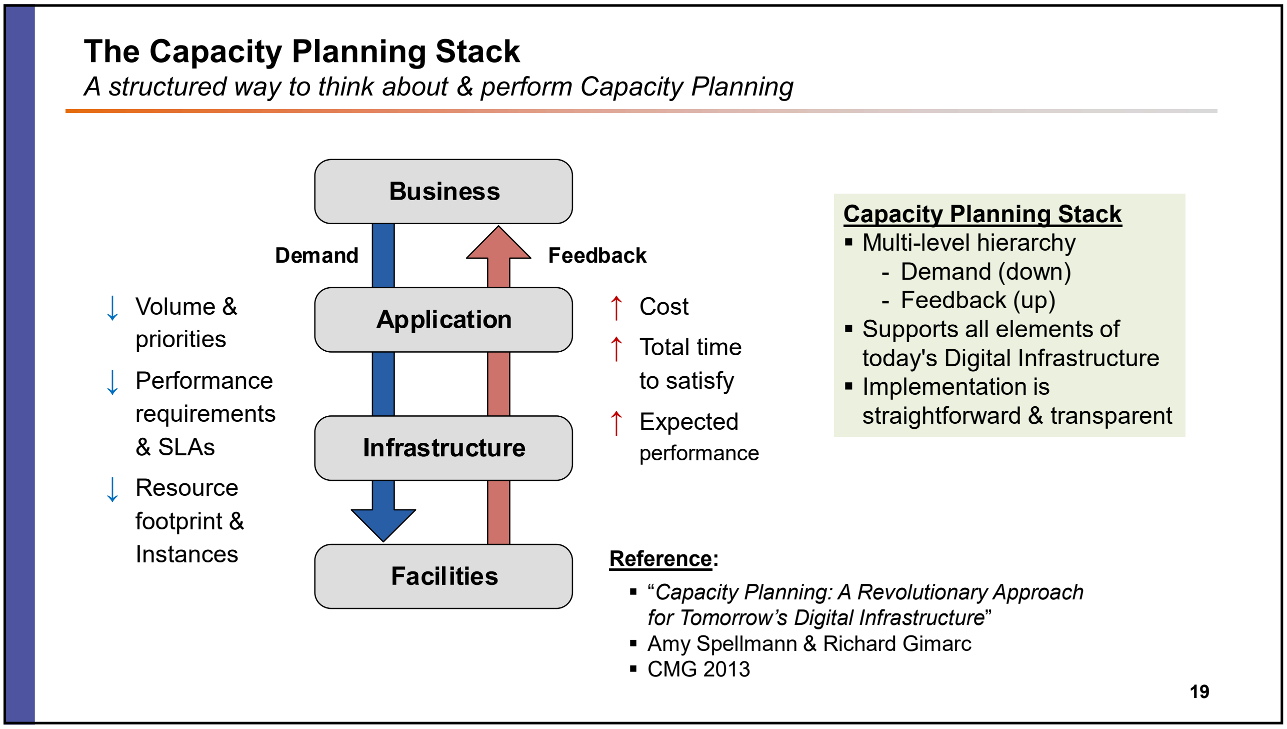
17



Including FaaS in your Capacity Plan

- *Capacity Planning Stack* developed for In-House (traditional), IaaS, PaaS, and SaaS also applies to FaaS
- Methodology is the same for all environments
- FaaS introduces new metrics & terminology

18



The Capacity Planning Stack

A structured way to think about & perform Capacity Planning

- ↓ Volume & priorities
- ↓ Performance requirements & SLAs
- ↓ Resource footprint & Instances

- ↑ Cost
- ↑ Total time to satisfy
- ↑ Expected performance

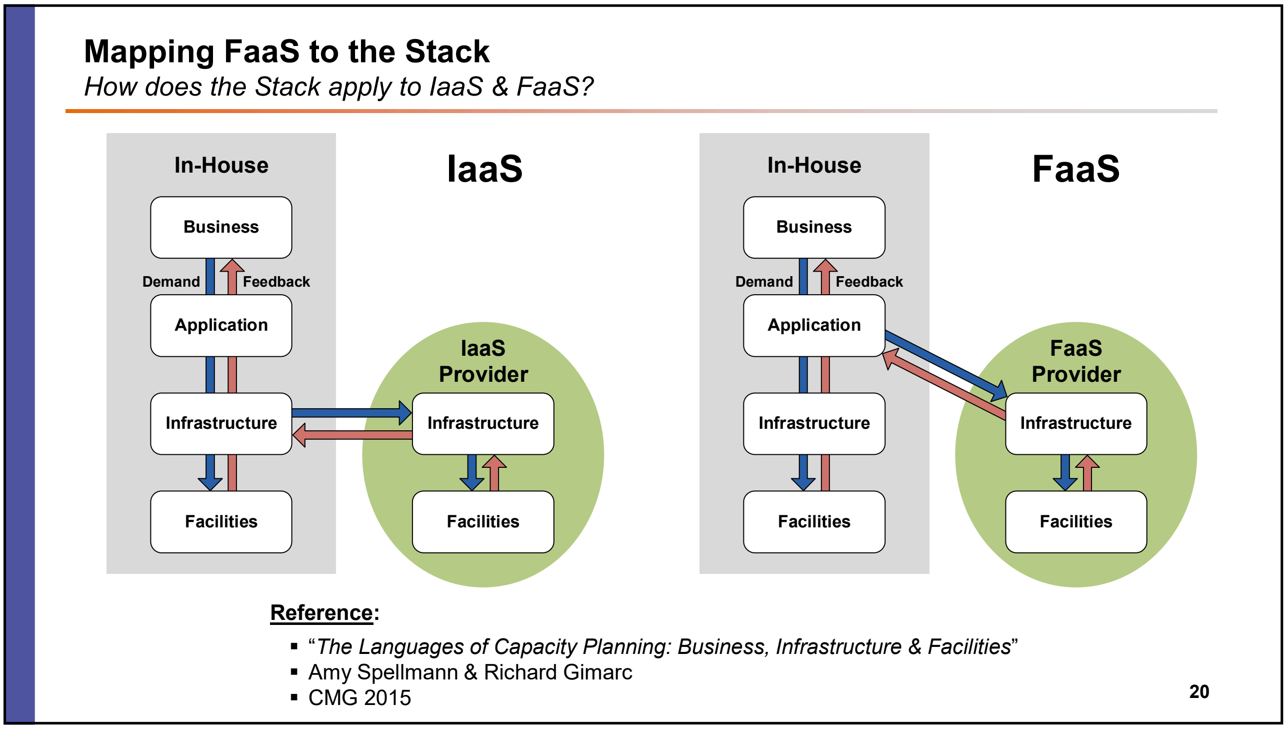
Capacity Planning Stack

- Multi-level hierarchy
  - Demand (down)
  - Feedback (up)
- Supports all elements of today's Digital Infrastructure
- Implementation is straightforward & transparent

Reference:

- "*Capacity Planning: A Revolutionary Approach for Tomorrow's Digital Infrastructure*"
- Amy Spellmann & Richard Gimarc
- CMG 2013

19



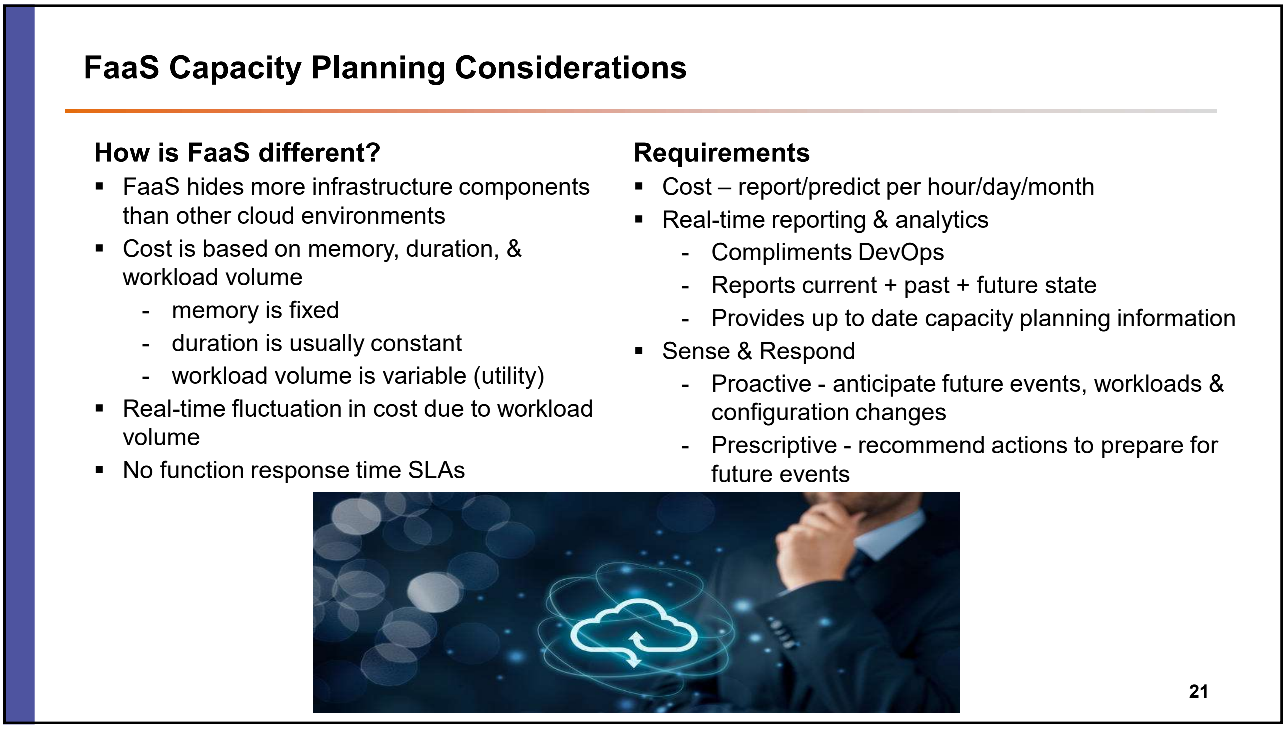
Mapping FaaS to the Stack

How does the Stack apply to IaaS & FaaS?

Reference:

- “*The Languages of Capacity Planning: Business, Infrastructure & Facilities*”
- Amy Spellmann & Richard Gimarc
- CMG 2015

20



FaaS Capacity Planning Considerations

How is FaaS different?

- FaaS hides more infrastructure components than other cloud environments
- Cost is based on memory, duration, & workload volume
  - memory is fixed
  - duration is usually constant
  - workload volume is variable (utility)
- Real-time fluctuation in cost due to workload volume
- No function response time SLAs

Requirements

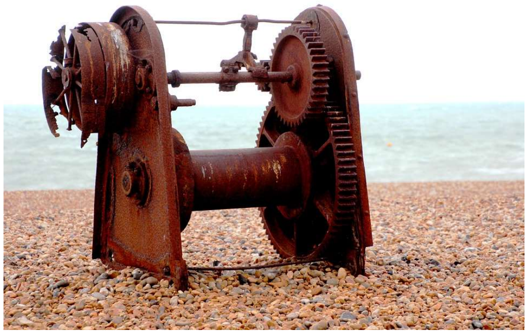
- Cost – report/predict per hour/day/month
- Real-time reporting & analytics
  - Compliments DevOps
  - Reports current + past + future state
  - Provides up to date capacity planning information
- Sense & Respond
  - Proactive - anticipate future events, workloads & configuration changes
  - Prescriptive - recommend actions to prepare for future events

21

Time for a new paradigm

Why do we need a new paradigm?

- Change perception about capacity planning
- Cost is the dominant factor
- FaaS makes cost even more variable & visible
- Small number of tuning knobs: memory, duration, workload



Why did people claim Capacity Planning is not required for serverless?

- FaaS does not require provisioning & scaling the application environment (servers)
- There's a lot of old baggage associated with the term "*capacity planning*"

Do we need a new term to replace "Capacity Planning"?

Audience Poll & Discussion

New Name?

- Digital Experience Planning
- BEV planning
- Digital Infrastructure Strategy
- ?
- ?

"*When the size of a problem changes by an order of magnitude, the problem itself changes.*" [Edsger Dijkstra]

References

[1 of 2]

[ALA2017] Sami Alabed, "Function as Service", April 2017, The University of Manchester

[CHE2018] Boris Chen, "*AppSec in the World of 'Serverless*", June 21, 2018, https://www.darkreading.com/cloud/appsec-in-the-world-of-serverless/a/d-id/1332078

[CNF2018] Cloud Native Computing Foundation (CNCF), "CNCF WG-Serverless Whitepaper v1.0", 2018, https://github.com/cncf/wg-serverless

[DAL2018] Jeremy Daly, "*15 Key Takeaways from the Serverless Talk at AWS Startup Day*", July 11, 2018, https://www.jeremydaly.com/15-key-takeaways-from-the-serverless-talk-at-aws-startup-day/

[DEV2018] Sushant Dewan, "*Monitoring Apps in the Serverless World (Part 1)*", July 24, 2018, https://www.wavefront.com/monitoring-applications-in-the-serverless-world-part-1-of-2/

[GIE2018] Michelle Gienow , "Serverless 101: How to Get Serverless Started in the Enterprise", June 4, 2018, https://thenewstack.io/serverless-101-how-to-get-serverless-started-in-the-enterprise/

[HUB2017] Patrick Hubbard, "*The Dark Side of Software-Defined Sprawl*", January 24, 2017, https://www.sdxcentral.com/articles/contributed/software-defined-sprawl/2017/01/

24

References

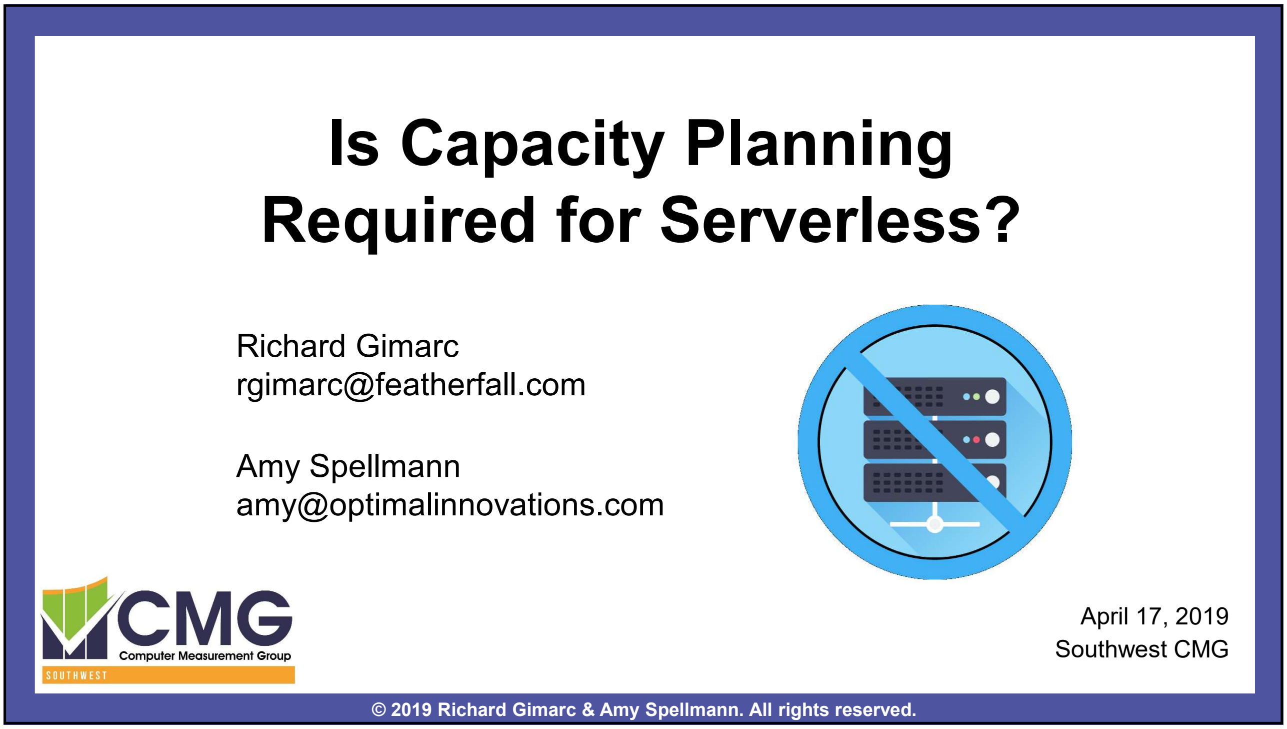
[2 of 2]

[SHA2018] Gwen Shapira, "Are you really building a serverless system?", InfoWorld, September 7, 2018, https://www.infoworld.com/article/3304385/serverless-computing/are-you-really-building-a-serverless-system.html

[SHC2018] Amiram Shachar, "The hidden costs of serverless", January 17, 2018, https://medium.com/@amiram_26122/the-hidden-costs-of-serverless-6ced7844780b

[SIM2018] Aleksandar Simovic, "7 ways Your Business will benefit through Serverless", January 26, 2018, https://hackernoon.com/7-ways-your-business-will-benefit-through-serverless-522b3f628a33

25



Is Capacity Planning Required for Serverless?

Richard Gimarc
rgimarc@featherfall.com

Amy Spellmann
amy@optimalinnovations.com

CMG Computer Measurement Group SOUTHWEST

April 17, 2019
Southwest CMG