Serverless for Managers: Everything You Need to Know

—

Erin McKean
Developer Advocate, SF City Team

think 2018

Think 2018 / March 20, 2018

IBM

Please note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Contents

0. What is ‘Serverless Computing’?

1. The Serverless Ecosystem

2. Serverless Use Cases

3. Serverless Best Practices

4. Serverless Resources

What is ‘Serverless Computing’?

‘serverless’ ≠ ‘no servers’

photo CC-BY-2.0 Flickr Paul Hammond

Features

Polyglot

- Major language support (Java, Node, Go, Python)
- Support for other languages via shims or community workarounds

Features

Cloud Deployment

- Usually writing for one provider, although frameworks can lessen pain of switching
- OpenWhisk allows you to create your own FAAS platform.

Features

Pay-per-invocation

- Pricing is usually based per-invocation and the amount of memory used
- Most providers have generous free tiers

Features

Scaling

- SEP: someone else’s problem

The Serverless Ecosystem

Major Players

AWS Lambda

Node.js (JavaScript), Python, Java (Java 8 compatible), and C# (.NET Core) and Go.

IBM Cloud Functions

Node.js , Python, PHP, and Swift; other languages can be used with Docker containers.

IBM CF is a managed instance of Apache OpenWhisk, the open-source serverless project.

Google Cloud Functions

Node.js only.

Azure Functions

C#, F#, and Node.js are GA; Java, Python, PHP, TypeScript, Bash, PowerShell, and .bat/.cmd are supported in Experimental mode in v1.X.

Other Players

Twilio Functions, Webtask from Auth0, Iron.io IronFunctions, StdLib, SpotInst, and more all the time ...

Choose your FAAS provider based on your needs, not hype

photo CC-BY-2.0 Flickr bdebaca

Serverless Use Cases

Use Cases

Low-Volume

- Contact forms
- Static (brochureware) websites
- Automated backups
- Bots
- Ops tasks (policy enforcement, uptime checks)

Use Cases

Bursty Compute

- Data processing (especially ETL)
- Data pipelines of all flavors
- Event-driven computing

Use Cases

Heavy Backend Tasks

- Generating PDFs
- Image resizing
- Batch processing

Use Cases

Tiny Apps

- APIs
- Microservices (sometimes called nanoservices)

Don’t run a server all the time if you don’t need a server all the time!

photo CC-BY-ND-2.0 Flickr santinet

Serverless Best Practices

Best Practices

Think Stateless!

Functions may have a ’clean slate’ each time … or they may not.

Best Practices

Think Event-Based!

Make sure you have a plan to handle failed or lost events.

Best Practices

Think Security!

Keep roles tight (think principle of least privilege) to limit access to functions, and to limit what resources functions can access.

Best Practices

Think Resources!

Tune your functions to use the right amount of memory and watch your execution time.

Best practices for serverless are much like best practices for cleanly coded functions.

photo CC-BY-2.0 Flickr daveynin

Things to Consider

Development

- Can you develop locally?
- How will you handle unit and integration testing?

Things to Consider

Deployment

- How will you handle versioning? Rollback?
- How will your CI/CD process change?

Things to Consider

Monitoring

- How will you profile your code?
- What will debugging look like?
- How will you manage and inspect your function logs?

Things to Consider

Service Discovery

- How will you keep track of service credentials?

Think about the characteristics of serverless applications both separately and together.

photo CC-BY-2.0 Flickr sidelong

Gotchas

Size

- If you’re using big libraries, you may run into the upper limit of deployable function size.
- Know the limits on the number of concurrent functions and number of deployed functions.

Gotchas

Recursion

- If your function calls itself, or calls other functions, watch out for infinite loops!

Gotchas

Datastores

- Your function may scale, but can your DB handle the write load?

Gotchas

Latency

- Cold starts are a problem!
- Consider keeping your function “warm” with scheduled invocations

Keep in mind the warning signs of a malfunctioning serverless function.

photo CC-BY-2.0 Flickr jakecaptive

Serverless Resources

Resources

Lists

github.com/anaibol/awesome-serverless

github.com/pmuens/awesome-serverless

twitter.com/tmclaughbos/lists/serverless

Newsletters

Serverless Weekly

Serverless Status

Cold Start

Code & Architecture

developer.ibm.com/code/patterns/category/serverless

github.com/serverless/examples

martinfowler.com/articles/serverless.html

Resources

Create a Learning Plan

- Pick one use case (look for a code pattern!)
- Decide on success criteria that matters to you
  - speed of development
  - reduced costs
  - more flexibility
- Look for blog posts by people who have implemented your use case—learn from others' mistakes!
- Prototype your business logic
- Prototype end-to-end:
  - local development
  - Logging/monitoring
  - CI/CD

Resources

IBM Cloud

- Sign up for an IBM Cloud Lite Account!
  https://ibm.biz/BdZqwA

Notices and disclaimers

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed "as is" without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply."

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Notices and disclaimers continued

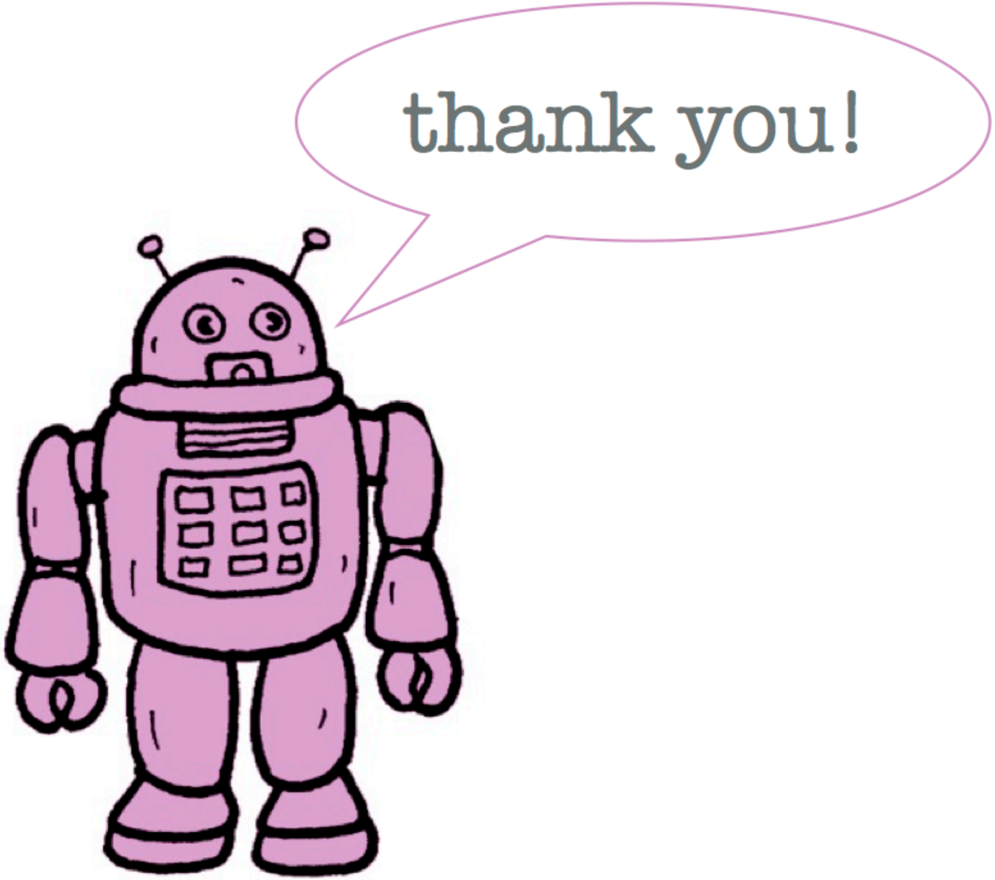
Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**
The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com and [names of other referenced IBM products and services used in the presentation] are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.

.

Thank you

Erin McKean
Developer Advocate, SF City Team
—
erin.mckean@ibm.com
@emckean
ibm.com



IBM