

Serverless C++

Benny Gaechter

How to transform a CLI tool into a serverless application

Cloureach[™]

Agenda

An abstract graphic on the left side of the slide, featuring a series of overlapping, wavy, orange-colored bands that create a sense of depth and movement, resembling a stylized staircase or a series of waves.

1. Intro / Motivation
2. What is Serverless
3. Theory and Architecture
4. Demo
5. Recap and Learnings
6. Q&A

Motivation

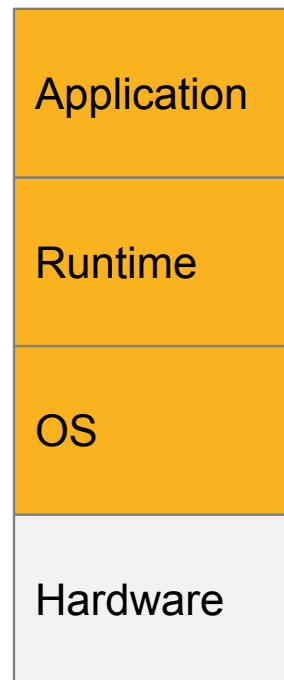
- Interest in all things cloud
- AWS offers a C++ SDK
- Wanted to do some hands on :)

What is serverless?

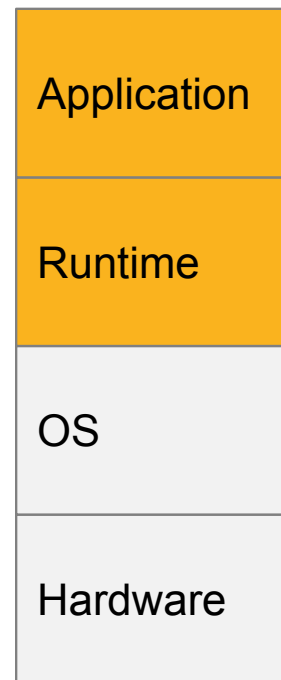
Traditional
on-prem



IaaS



PaaS



Serverless
(FaaS)



AWS Lambda in numbers

Resource	Limit
Function memory allocation	128 MB to 3,008 MB, in 64 MB increments.
Function timeout	900 seconds (15 minutes)
Function environment variables	4 KB
Function resource-based policy	20 KB
Function layers	5 layers
Function burst concurrency	500 - 3000 (varies per region)
Invocation frequency (requests per second)	10 x concurrent executions limit (synchronous – all sources) 10 x concurrent executions limit (asynchronous – non-AWS sources) Unlimited (asynchronous – AWS service sources)

AWS Lambda runtime

- C++ needs a custom runtime
- Provide an entrypoint
- Has to perform some initialization tasks
- Processes invocation events in a loop until terminated
- There is a premade custom runtime from AWS

Inspirational quote generator

- Like fortune
- But with “AI” to generate quotes

Theory

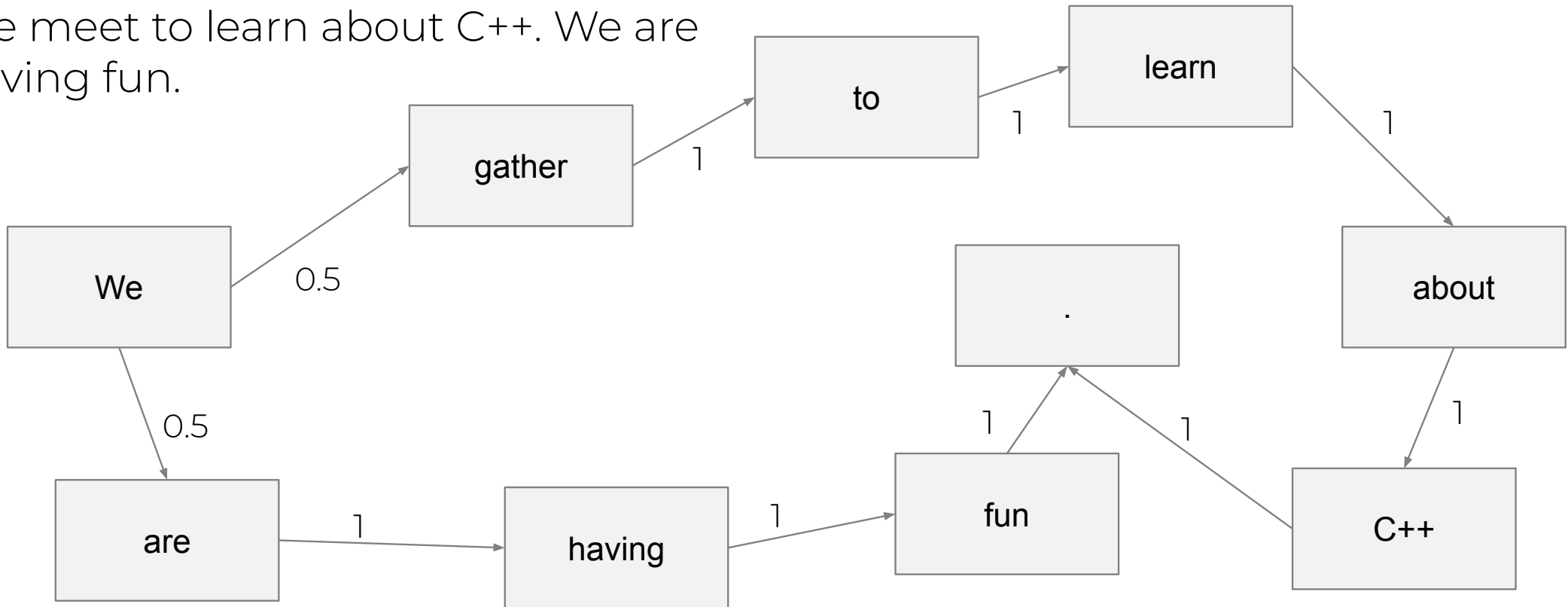
- “Learn” from text
- Use probability to find next word(s)

Markov Chain

We gather to learn about C++. We are having fun.

Markov Chain Example

We meet to learn about C++. We are having fun.



In C++ Code

```
std::map<std::string, std::vector<std::string>> dictionary;
```

```
[...]
```

```
void createDictionary(std::string *input) {
```

```
[...]
```

```
while (iss >> word) {
```

```
    dictionary[prev_prev_word + " " + prev_word].push_back(word);
```

```
    prev_prev_word = prev_word;
```

```
    prev_word = word;
```

```
}
```

```
}
```

The chain as map of vectors

Key	Value
We	meet, are, think, think
meet	to
to	learn, listen, learn

Creating the quote

```
while (quote.back() != '.' && quote.back() != '?' && quote.back() != '!') {  
    const auto key = quote.substr(key_start, quote.size());  
    auto words = dictionary[key];  
    std::uniform_int_distribution<> word_dis(0, words.size() - 1);  
    int index = word_dis(gen);  
    quote += " " + words[index];  
    key_start = advanceWord(key_start, quote);  
}
```

Making it serverless

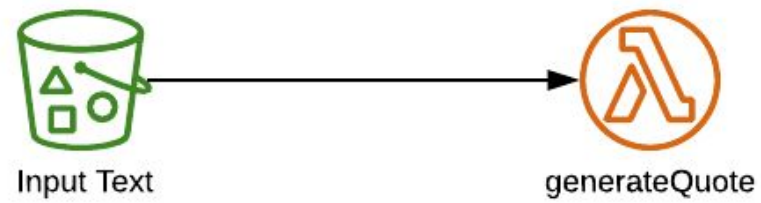
```
static invocation_response my_handler(invocation_request const &req) {  
    try {  
        auto const quote = generate_quote();  
        return invocation_response::success(quote, "application/json");  
    } catch (const std::exception &e) {  
        return invocation_response::failure(e.what(), "application/json");  
    }  
}  
  
int main() {  
    run_handler(my_handler);  
    return 0;  
}
```

Architecture

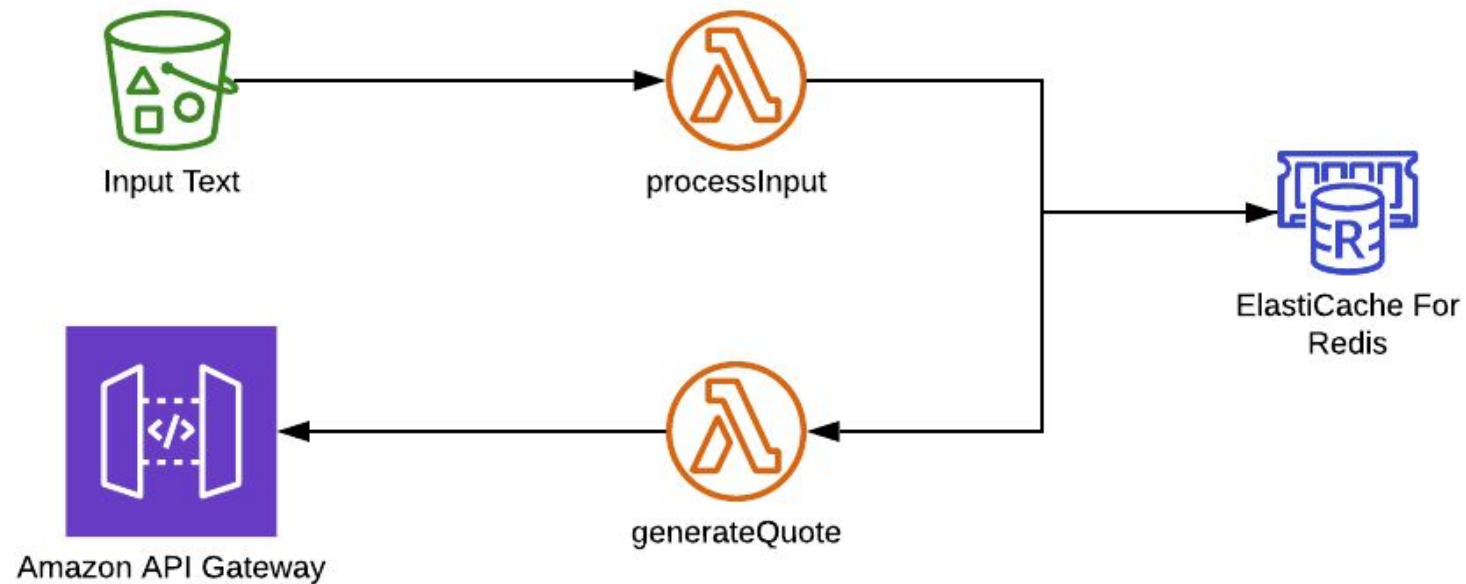


generateQuote

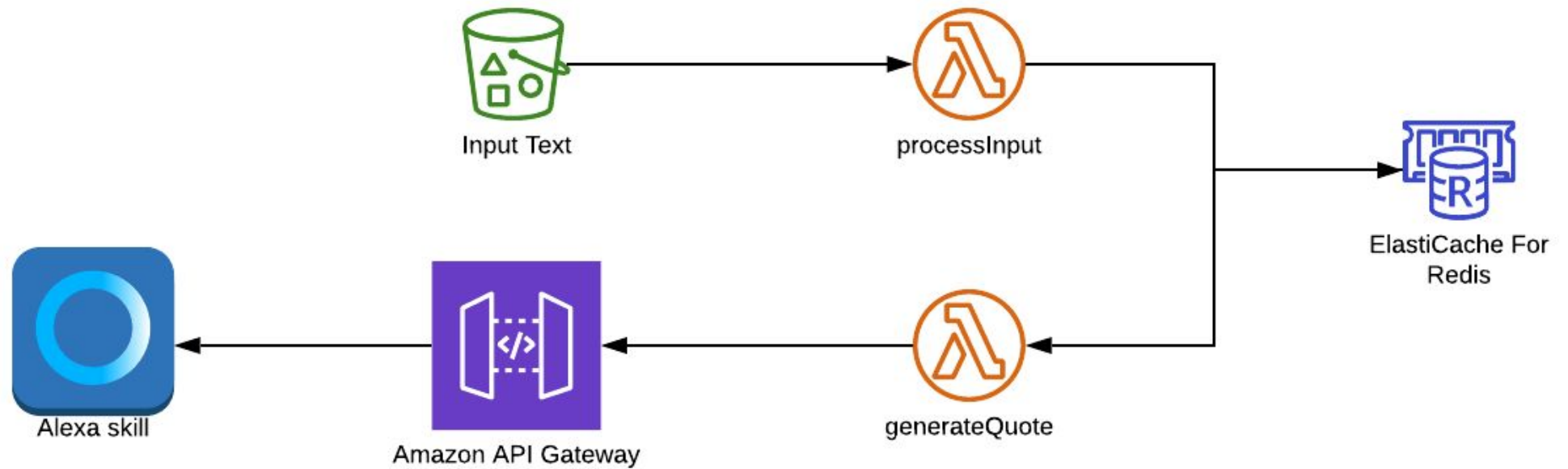
Architecture



Architecture



Overview



Demo



Recap

Code can be easily moved to the cloud but usually needs transformation

Automation (especially CI/CD) becomes more important

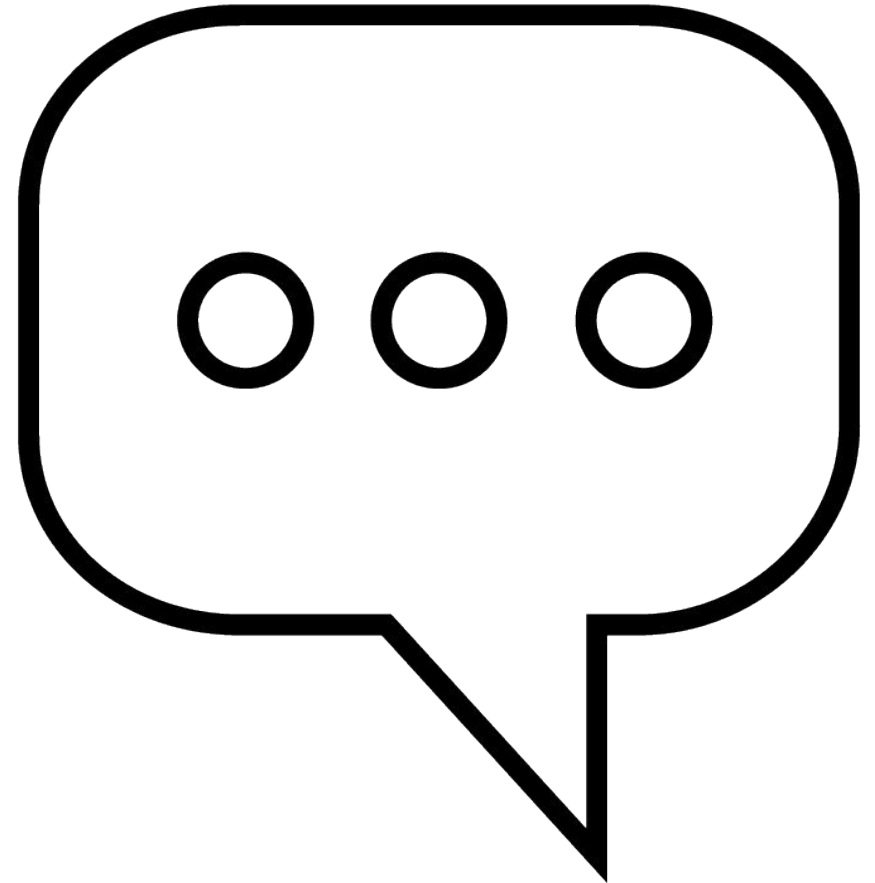
Let the cloud take care of tedious stuff (Auth, persistence, availability)

Trade off between scalability vs. single execution time

AWS cpp sdk is amazing - once it is setup :)

Q & A

Code: github.com/bgaechter/serverless-cpp



Thank you