Architecting A Cloud-native Data Analysis Application for ETDs

Yinlin Chen & Edward A. Fox
{ylchen, fox}@vt.edu

Virginia Tech, Blacksburg, VA 24061, USA

26 Sept. Presentation at ETD 2018, Taiwan
Agenda

• Introduction
• Monolithic vs. Cloud-native applications
• Cloud-native approach
• Data analytics architecture
• Future work
VTechWorks

- Research Documents
  - 60,000+ scholarly works
  - Inc. 30,000+ ETDs

- Using DSpace
  - Open digital repository
  - Open source project
  - Monolithic architecture
Problems

• Need for advanced analytics functionalities
  – Customizable
  – Visualization
  – Usable and flexible user interface (UI)
• Extending existing codebase is a difficult task.
  – Customization that meets community needs is rare.
  – Version update will lose or break customized functionalities.
From Research Dataset to Visualization

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>50440</td>
<td>10919/11041</td>
<td>10919/71:1</td>
<td>Larochelle, Catherine</td>
<td>Alwang, Jeffrey R.</td>
<td>Amacher, Gregory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43029</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Ketene, Alperen Nurullah</td>
<td>Agah, Masoud</td>
<td>Behkam, Baharen</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42401</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Smith, Ryan Christopher</td>
<td>Geller, E. Scott</td>
<td>Spencer, Edward F</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43484</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Zareian-Jahromi, Mohammad Amin</td>
<td>Agah, Masoud</td>
<td>Raman, Sanjay</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43064</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Schuler, Matthew Michael</td>
<td>FitzPatrick, William J.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89430</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Kohler, Rachel Elizabeth</td>
<td>Luther, Kurt</td>
<td>North, Christopher</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89094</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Dellinger, Elizabeth Aalseth</td>
<td>Gardner, Thomas M</td>
<td>Swenson, Karen</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>88738</td>
<td>10919/92911</td>
<td>10919/71:7</td>
<td>Williams, Rebecca Jean</td>
<td>Jones, Kathleen W</td>
<td>Shumsky, Neil Larri</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11228</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Lee, Dong-Il Ho</td>
<td>Lee, Fred C.</td>
<td>Chen, Dan Y.</td>
<td>Bor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11229</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Kim, Byung-Hi</td>
<td>Stutzman, Warren L</td>
<td>Sweeney, Dennis C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11230</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Kris, Kerri Lynn Murphy</td>
<td>Madison-Colmore, Octavia D.</td>
<td>Messier, Louis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11231</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Lipkovich, Ilya A</td>
<td>Smith, Eric P.</td>
<td>Ye, Keying</td>
<td>Foutz, Robert</td>
<td>Bis</td>
<td></td>
</tr>
<tr>
<td>11232</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Lorca, Tatsiana Andrea</td>
<td>Pierson, Merle D.</td>
<td>Effert, Joseph D.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11441</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Crozier, James Brooks</td>
<td>Stromberg, Erik C.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11442</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Cooper, Jamie S.</td>
<td>Cooper, Robin K. Panneton</td>
<td>Lickliter, Robert E.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11443</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Clark, Paul Alexander</td>
<td>Oyama, Shigoe Ted</td>
<td>Vandsburger, Uri</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11444</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Hafsteinsson, Leifur Geir</td>
<td>Donovan, John J.</td>
<td>Carlson, Kevin D.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11445</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Rowland, Amy Lee</td>
<td>Stratton, Richard K.</td>
<td>Poole, Jon R.</td>
<td>Kno</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11235</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Prater, Mary Renee</td>
<td>Holliday, Steven E.</td>
<td>Wong, Eric A.</td>
<td>Be</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11447</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Zhao, Xiaopeng</td>
<td>Nayfeyh, Ali H.</td>
<td>Danekowicz, Harry R.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11448</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Rosario, Astrid Christa</td>
<td>Riffle, Judy S.</td>
<td>Long, Timothy E.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11238</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Torrence, Vera D.</td>
<td>Krill, Cecelia W.</td>
<td>Parson, Stephen R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11450</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Niu, Sanjun</td>
<td>Saraf, Rafi F.</td>
<td>Oyama, Shigoe Te</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11240</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Parry, Matthew Barton</td>
<td>Stegeman, Mark</td>
<td>Eckel, Catherine C.</td>
<td>Hailer, Hans H.</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>11242</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Lahouar, Samer</td>
<td>Al-Qadi, Imadeddin L.</td>
<td>Brown, Gary S.</td>
<td>Da Wolf, David A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11244</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Seoock, Yoo-Kyoung</td>
<td>Norton, Mersorie J. T.</td>
<td>Littlefield, James I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11453</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Robinson, Tammy Renee</td>
<td>Giddings, Carol A.</td>
<td>Bailey, Giddings, Valerie L</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11454</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Cleverger, Jennifer Lynn</td>
<td>Giddings, Valerie L</td>
<td>Kincade, Doris H</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11455</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Gough, Christopher Michael</td>
<td>Seiler, John R.</td>
<td>Fox, Thomas R.</td>
<td>I</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11456</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Ahmed, Farzana</td>
<td>Larson, Timothy J.</td>
<td>Popham, David L</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11457</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Lee, Yuri</td>
<td>Kincade, Doris H</td>
<td>Giddings, Valerie</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11458</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Jiang, Hongqing</td>
<td>Hsu, Kudir W</td>
<td>Opell, Brent D.</td>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11459</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Liu, Siun</td>
<td>Stegeman, Mark</td>
<td>Eckel, Catherine C.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11460</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Li, Yuxin</td>
<td>Huong, Alex Q</td>
<td>Griffin, Carl A.</td>
<td>Maroof, M. A. Sagahai</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11461</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>van Aardt, Jan Andreas Nicholas</td>
<td>Wynne, Randolph H.</td>
<td>Oderwald, Richard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11462</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Reinecker, Scott Harold</td>
<td>Zink-Sharp, Audrey G.</td>
<td>Glasser, Wolfgang</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11463</td>
<td>10919/11041</td>
<td>10919/71:7</td>
<td>Perry, Elizabeth A.</td>
<td>Salehi-Isfahani, Djavad</td>
<td>Ducker, William A</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Categorized by Department

[Diagram showing categorized research by department]
From Log Data to Visualization
Agenda

• Introduction
• *Monolithic vs. Cloud-native applications*
• Cloud-native approach
• Data analytics architecture
• Future work
Monolithic Architecture

• Develop and deploy as a single unit
• Often requires human intervention
• Long-term commitment to a technology stack or even version
• Hard to scale development
• Difficult to scale the application
Why Towards Cloud Native

- Limited resources:
  - Developers, DevOps, Infrastructure, Time
- Reduce need to build everything from scratch
- Use services that can help deliver the project
- Facilitate the development process
- Provide better services: fault-tolerant, auto-scale, update/rollback without downtime, etc.
- Optimize resource utilization
Resource Usage Optimization and Automation

- Consume only the required resources for the applications
- Scale up and down automatically
- Service and function oriented, not server oriented
- Utilize cloud services to help understand applications (CloudWatch, Auto Scaling, Trusted Advisor, etc.)
Agenda

• Introduction
• Monolithic vs. Cloud-native applications
• Cloud-native approach
• Data analytics architecture
• Future work
What is Cloud Native?

It is not just putting applications in the cloud.

It is about building applications in the cloud that utilize the advantages provided by the cloud **AS MUCH AS POSSIBLE.**
Cloud Native

- Cloud Native Computing Foundation (CNCF)
  - An open source software foundation dedicated to making cloud native computing universal and sustainable
- Microservices oriented
- Containerized
- Dynamically orchestrated
Microservices oriented
Microservice

• Small software piece
• Messaging enabled – communicate with messages
• Decentralized
  – Autonomously developed
  – Independently deployable
  – Can change independently of each service
  – Scale individually by load
• Built and released with automated processes
• More complex architecture
Serverless

Does not mean “There are no servers at all”.

Does mean “Use fully managed services”.

Focus on application development, not server maintenance
Parallel Development and Deployment

Configure → Stacks → Application

GitHub

AWS Elastic Beanstalk

AWS Lambda

Amazon API Gateway

Application
Containerized
Containerization

- BaaS (Backend as a Service), CaaS (Container as a Service), and FaaS (Function as a Service)
- Best practice is each service in its own isolated environment, e.g., Docker container.
- But a container can run multiple services, or an entire application.
- Everything at Google runs in a container
  - 4 Billion containers per week in 2018
Dynamically orchestrated
Orchestration

- Infrastructure as Code
- Automatic deployment and operation
- Optimize resource utilization dynamically
Data Analytics Pipeline

- Collect
- Store
- Process
- Analyze & Visualize
Design Pattern and Best Practice

- The Twelve-Factor App (http://12factor.net)

<table>
<thead>
<tr>
<th>Codebase</th>
<th>Dependencies</th>
<th>Config</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backing services</td>
<td>Build, release, run</td>
<td>Processes</td>
</tr>
<tr>
<td>Port binding</td>
<td>Concurrency</td>
<td>Disposability</td>
</tr>
<tr>
<td>Dev/prod parity</td>
<td>Logs</td>
<td>Admin processes</td>
</tr>
</tbody>
</table>
Design Strategies

- Microservices
- Containers
- Orchestration
- Serverless (managed service)
- Scalability
- Automation
- Optimization (resource usage, cost, etc.)
Agenda

• Introduction
• Monolithic vs. Cloud-native applications
• Cloud-native approach
• Data analytics architecture
• Future work
Application Architecture Overview

Files upload from VTechWorks Repository with ETD Data & Logs to Amazon S3. Data is then processed by AWS Lambda and stored in Amazon S3 Buckets. The data is also sent to Amazon DynamoDB and Amazon ElasticSearch. The results can be visualized using Kibana.
AWS Services

- AWS S3: Object storage in the cloud
- AWS Lambda: Serverless compute platform for stateless code execution in response to events
- Amazon Dynamodb: Fully managed NoSQL database service
- Amazon ElasticSearch: Search engine based on Lucene
- Kibana: An open source data visualization plugin for ElasticSearch
Microservice – Using AWS Lambda

- **vtetedstodynamodb**
  - S3
  - Amazon CloudWatch Logs
  - Amazon DynamoDB
  - Amazon S3
  - Resources the function’s role has access to will be shown here

- **dyno-to-es**
  - DynamoDB
  - Amazon CloudWatch Logs
  - Amazon DynamoDB
  - Amazon Elasticsearch Service
  - Resources the function’s role has access to will be shown here

Add triggers from the list on the left.
Metadata Transformation Using AWS Lambda

According to the American Cancer Society, Cancer is the second most common cause of death in the United States, only exceeded by heart disease. Over the past decade, deciphering the complex structure of individual cells and understanding the symptoms of cancer disease has been a highly emphasized research area. The exact cause of Cancer and the genetic heterogeneity that determines the severity of the disease and its response to treatment has been a great challenge. Researchers from the engineering discipline have increasingly made use of recent technological innovations, namely the Atomic Force Microscope (AFM), to better understand cell physics and provide a means for cell biomechanical profiling.
Log Data Transformation Using AWS Lambda


{
    "website": "vtechworks.lib.vt.edu",
    "client_IP": "202.248.84.158",
    "http_version": "HTTP/1.1",
    "http_method": "GET",
    "web_client": "Mozilla/5.0",
    "request_uri": "/bitstream/handle/10919/33268/WGraf_Thesis_2005.pdf?sequence=1"
}
Results

• Facilitate application development
  – Parallel development and deployment
  – Switch services or techniques more flexibly

• Decouple the data analytics pipeline
  – More complex architecture and testing setup
  – More things to learn in order to select right tools

• Delegate maintenance tasks to cloud providers
Agenda

• Introduction
• Monolithic vs. Cloud-native applications
• Cloud-native approach
• Data analytics architecture
• Future work
Data analytics pipeline in AWS

Collect

Real-time
- Amazon Kinesis Firehose
Data Import
- Amazon Import/Export
- Snowball
Message Queuing
- Amazon SQS
Web/app Servers
- Amazon EC2

Store

Object Store
- Amazon S3
- Amazon Glacier
Real-time
- Amazon Kinesis Streams
RDBMS
- Amazon RDS
NoSQL
- DynamoDB
Search
- Amazon CloudSearch
IoT
- Amazon IoT

Process & Analyze

Hadoop Ecosystem
- Amazon EMR
Real-time
- AWS Lambda
- Amazon Kinesis Analytics
Data Warehousing
- Amazon Redshift
Machine Learning
- Amazon Machine Learning
Elastic Search Analytics
- Amazon ElasticSearch
Process & Move Data
- AWS Data Pipeline

Visualize

Business Intelligence & Data Visualization
- Amazon QuickSight
Elastic Search Analytics
- Amazon ElasticSearch

Blue: current
Green: alternate
Black: other possible
Cloud Native Data Analysis Platform in AWS

- IAM
- AWS Organizations
- AWS Lambda
- Amazon DynamoDB
- Amazon ElasticSearch
- Amazon S3
- Amazon Glacier
- Kibana
- *Amazon Athena
- *Amazon Machine Learning
- AWS CloudFormation
- AWS CloudTrail
- AWS Trusted Advisor
- Amazon CloudWatch
- AWS Config

AWS CLI
### Other Cloud Platforms

- **Amazon Web Services (AWS)** - done
- **Google Cloud Platform (GCP)** – easily done
- **Microsoft Azure, etc.** – also possible

<table>
<thead>
<tr>
<th>AWS</th>
<th>GCP</th>
<th>Azure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elastic Compute Cloud</td>
<td>Compute Engine</td>
<td>Virtual Machines</td>
</tr>
<tr>
<td>Elastic Beanstalk</td>
<td>Google App Engine</td>
<td>Cloud Services</td>
</tr>
<tr>
<td>EC2 Container Service</td>
<td>Kubernetes Engine</td>
<td>Container Service (AKS)</td>
</tr>
<tr>
<td>Kubernetes (EKS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lambda</td>
<td>Cloud Functions</td>
<td>Functions</td>
</tr>
<tr>
<td>Simple Storage Services</td>
<td>Cloud Storage</td>
<td>Storage</td>
</tr>
<tr>
<td>Virtual Private Cloud</td>
<td>Virtual Private Cloud</td>
<td>Virtual Network</td>
</tr>
</tbody>
</table>
Q & A

Supported by
Virginia Tech Libraries and
AWS Cloud Credits for Research program

Thank You!

https://github.com/yinlinchen/ETD18