Orchestrating the Deployment of Computations in the Cloud with Conductor

Alexander Wieder
Pramod Bhatotia
Ansley Post
Rodrigo Rodrigues

NSDI 2012

27.04.2012
What's the best strategy to use cloud services?
Why is choosing the best strategy challenging?

**Variety of services and providers** with different:
- Pricing models
- Performance characteristics
- Locations
- Interfaces

**Hybrid deployments**
- Use own infrastructure and/or multiple different services at the same time

**Dynamics during runtime**
- Performance variations
- Spot markets
Conductor Goals

Simplify the management of cloud resources:

- **Automatization**: Automatically optimize resource allocation
- **Transparency**: Use multiple different services seamlessly
- **Adaptivity**: Automatically adapt to dynamics
  - Performance variations
  - Variable resource cost on spot markets
Outline

- Conductor System Overview
- Modeling Computations
- Using Cloud Resources Transparently
- Evaluation
High Level System Design

- **Frameworks**: Dryad
  - submit job to framework
  - monitor execution
  - allocate resources
  - run job

- **Controller**
  - allocate resources
  - execution plan

- **LP Solver**
  - execution plan

How can we transparently use cloud resources?

How can we model computations?
Outline

• Conductor System Overview ✔
• Modeling Computations
• Using Cloud Resources Transparently
• Evaluation
Modeling Computations

- Hard to model computations in general case

- **Unknown:**
  - Data access patterns
  - Processing time
  - Scalability

- Feasible for specific programming models, e.g., **MapReduce**
Modeling MapReduce Computations

How can we model MapReduce Computations?

• Data-parallel processing
• Mostly linear dependencies:
  • Performance
  • Resources
  • Cost

→ Problem calls for a formulation as a linear program!
Computation steps:
- Storing data
- Transferring data
- Processing data
- Migrating data

Graph based model:
- **Vertices**: data storage and processing
- **Edges**: data transfer
Outline

• Conductor System Overview

• Modeling Computations

• Using Cloud Resources Transparently

• Evaluation
Deploying Jobs on the Cloud

Frameworks:
- Dryad
- Hadoop

Resource Abstraction Layer:
- Storage
- Computation

Backend specific interface

Local HD on VM

S3

migrate and upload
Outline

• Conductor System Overview ✔
• Modeling Computations ✔
• Using Cloud Resources Transparently ✔
• Evaluation
Evaluation

Questions we answer in the evaluation:

• Can Conductor find optimal execution plans?
• Can Conductor efficiently adapt to dynamics?
• Can Conductor enable hybrid deployments?
• What overheads does Conductor impose?

see paper
Evaluation
Finding Optimal Execution Plans

Scenario:

- Job: k-means clustering, 32GB input data
- Resources: EC2, S3
- Deadline: 6h
- Minimize monetary cost

Goal:

- Automatically select resources
- Manage data transfer
- Launch job
Evaluation
Finding Optimal Execution Plans

storing 1/3 on S3 and 2/3 on EC2 is optimal
Evaluation
Adapting to Dynamics

Observed resource performance in the cloud can vary for several reasons:
• Interference with co-located VM instances
• Network congestion
• Failures

Scenario:
• EC2 performance ~3x overestimated

Conductor doesn't allocate enough resources to finish before deadline
Evaluation
Adapting to Dynamics

Job progress:

Conductor updated deployment after 1h

Allocated nodes:
Can Conductor help cutting cost by leveraging spot resources?
Evaluation
Adapting to Spot Market Prices

Methodology:

- Simulate job deployment using EC2 spot instances
- Spot pricing history over ~4 weeks
- Conductor uses an oracle or simple pricing predictor

![Bar chart showing cost comparison]

- Regular: 26.6
- Oracle: 12.12
- Predictor: 12.33
Outline

- Conductor System Overview ✓
- Modeling Computations ✓
- Using Cloud Resources Transparently ✓
- Evaluation ✓
Summary and Conclusion

Observation:
Making best use of the cloud is hard!

Conductor's approach:
- LP-based system model
- Optimize for user goals
- Resource abstraction layers
- Adapt during runtime

Evaluation results:
Conductor can efficiently manage cloud deployments

Future work:
Apply Conductor's approach to other frameworks
Thanks for your Attention!