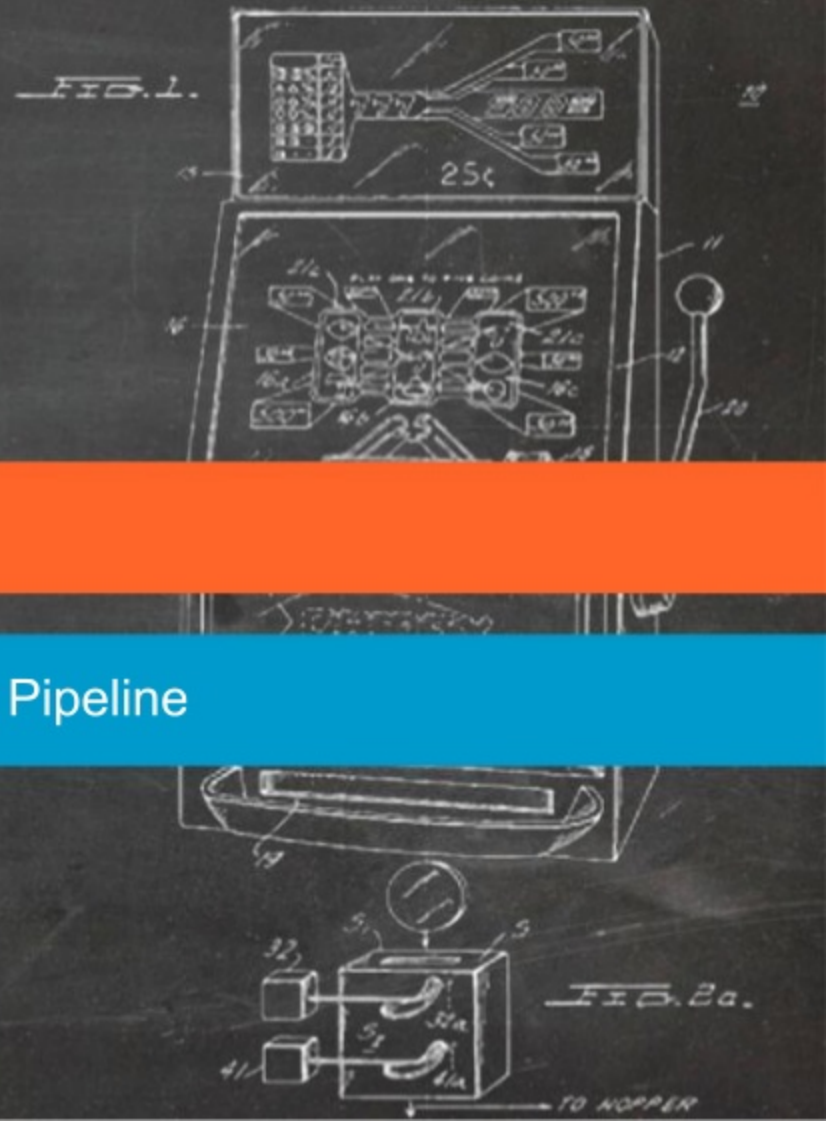
AWS re:Invent

November 27- 29, 2012 | The Venetian | Las Vegas

AWS Data Pipeline

Kathryn Shih, Sr. Product Manager, AWS Data Pipeline



What is AWS Data Pipeline?

- Scheduling, management, and orchestration for data-driven workflows
- Data Pipeline manages the annoying details so that you can focus on the business logic

Pricing

- On demand pricing
- Pay for what you use
- No minimum commitment or upfront fees

The Goals

- What's the problem?
- How does AWS Data Pipeline solve it?
- Technical examples

Example problem: Content targeting

- Managing a website with premium content
- Goal: improve business by better targeting content
- Second goal: don’t break/rewrite the website
- Third goal: lower costs!

Where does the data live?

Amazon S3

Amazon
RDS

Amazon
DynamoDB

Amazon
Redshift

HDFS
(Amazon EMR)

On
Premise

Amazon DynamoDB

- Very fast I/O
- Provision as much throughput as you need

Amazon S3

- Bulk object storage
- Huge items

Targeting: What data do we have?

- Historical webserver logs for all users
- Near real-time purchase data for registered users
- User demographic data (from signups)
- 3rd party IP geolocation data

Where does the data live?

Amazon S3

Amazon
RDS

Amazon
DynamoDB

Amazon
Redshift

HDFS
(Amazon EMR)

On
Premise

Where does the data live?

Amazon S3

Amazon
DS

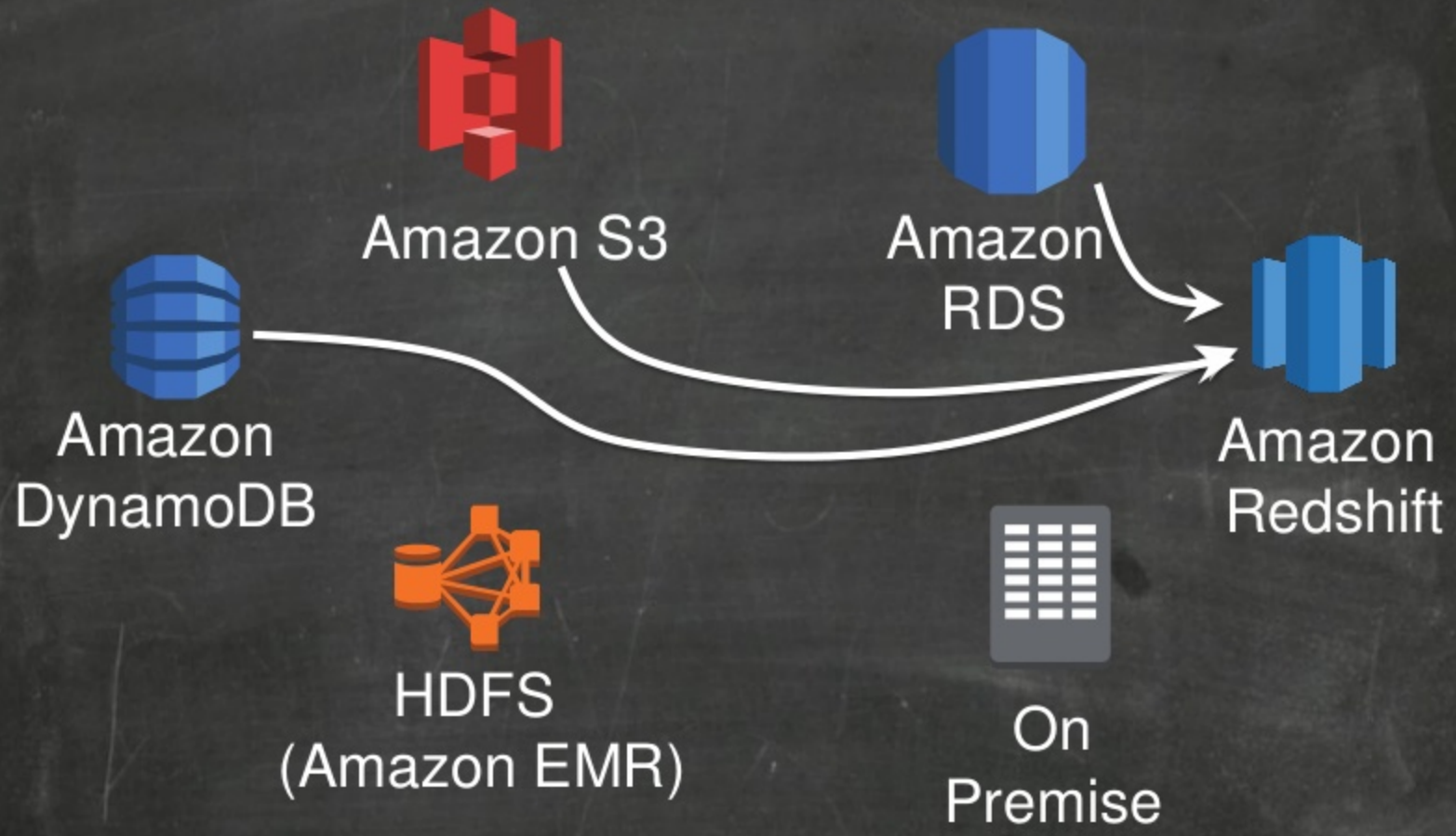
Amazon
DynamoDB

Amazon
Redshift

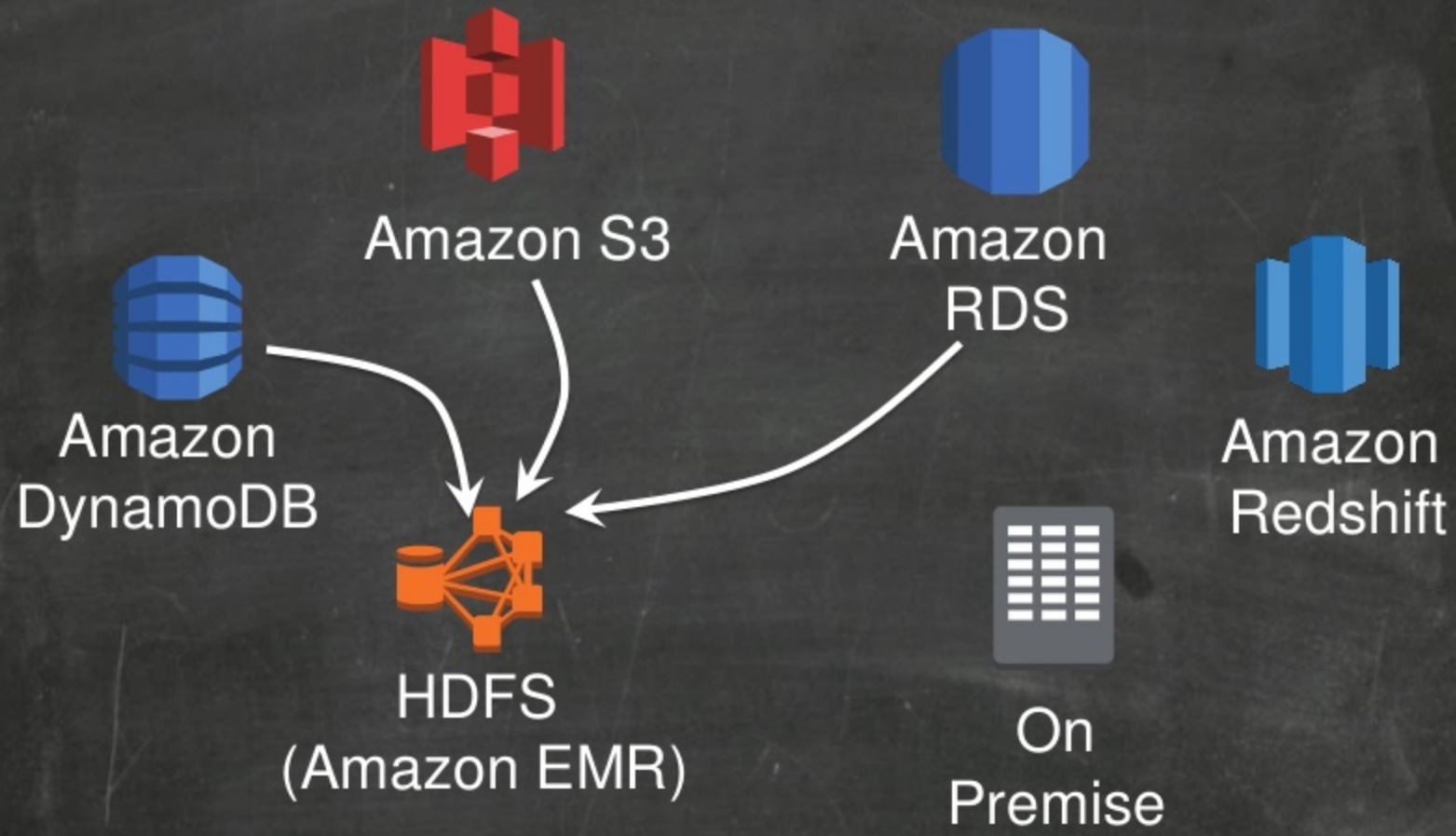
HDFS
(Amazon EMR)

On
Premise

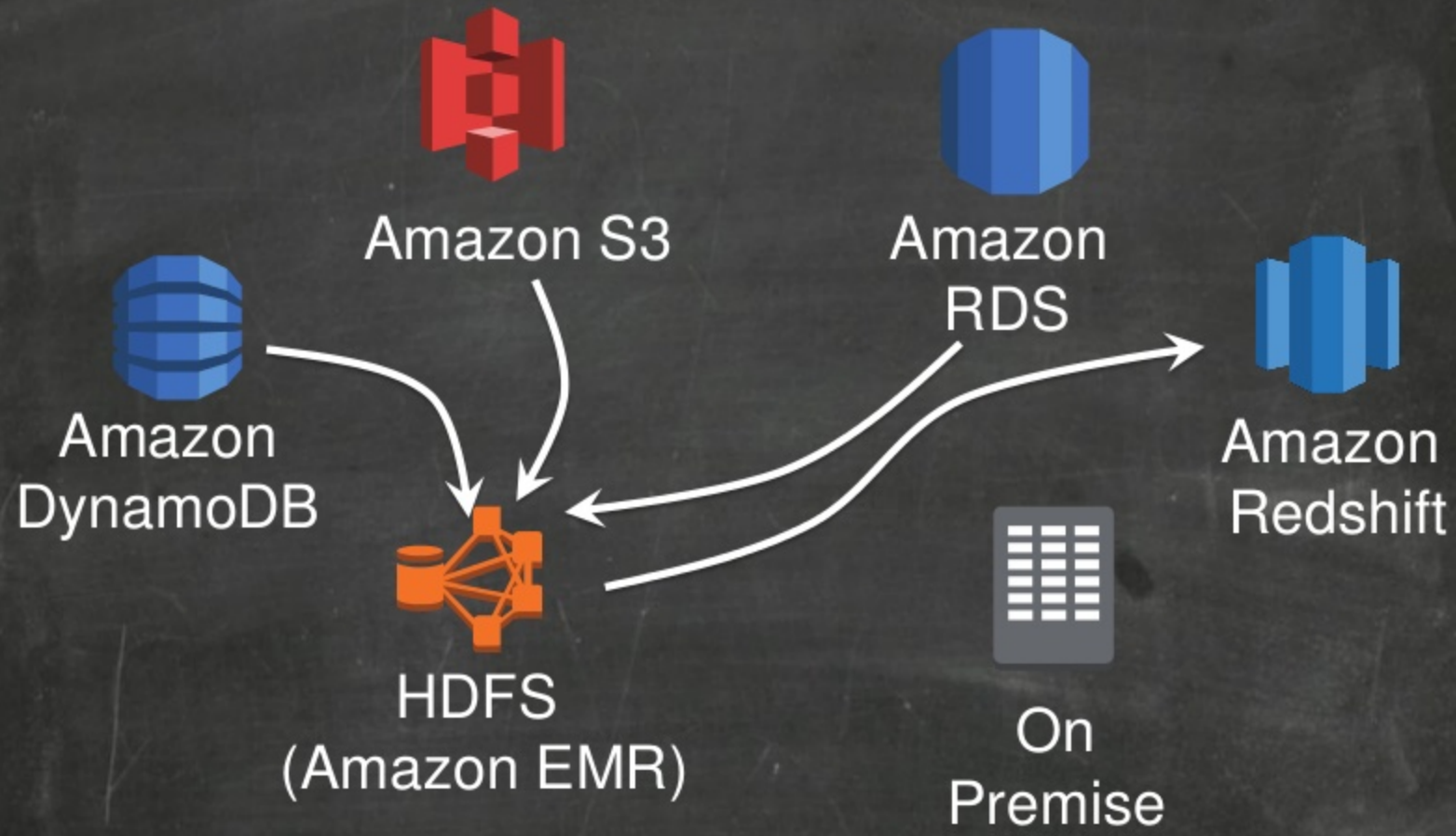
Where does the data live?



Where does the data live?



Where does the data live?



Tip of the iceberg

- Social network data
- Purchase/usage history
- Simulation outputs
- Videos & images
- Sensor data (e.g., GPS)
- Additional 3rd party datasets

Data Pipeline tour

- What's it do?
- How's it help here?

Minimum Viable Pipeline

Input Datanode

Activity

[Output Datanode]

Less Minimum

Input Datanode with precondition check

Activity with failure & delay notifications

Ouput Datanode

Preconditions

Hosted by AWS Data Pipeline:

- Amazon S3 Files/Directories exist
- Amazon DynamoDB tables exist
- Amazon RDS queries
- Amazon Redshift queries

Hosted by you:

- JDBC queries against on-premise databases return results
- Custom scripts

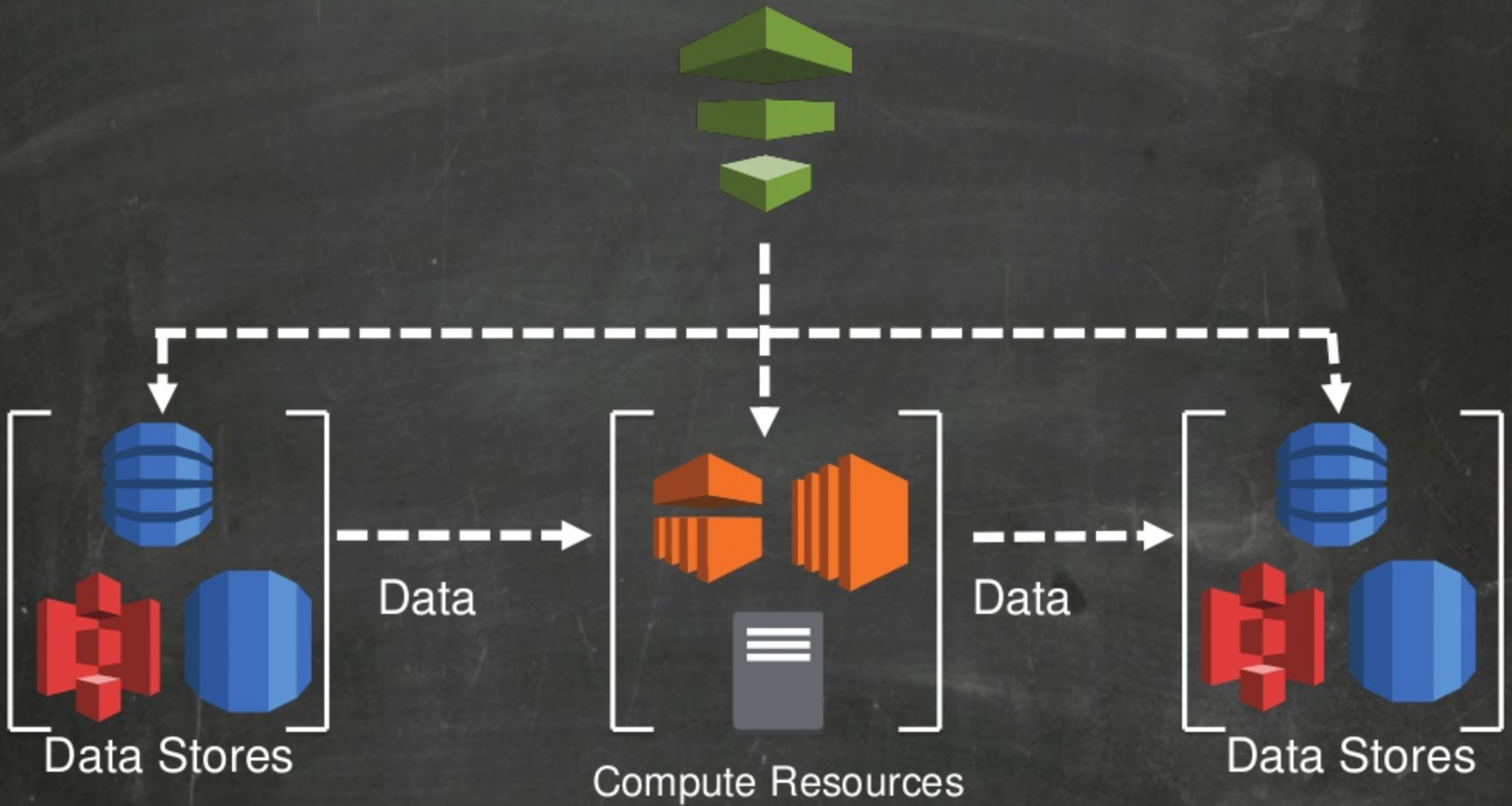
Retries & Notifications

Retries:

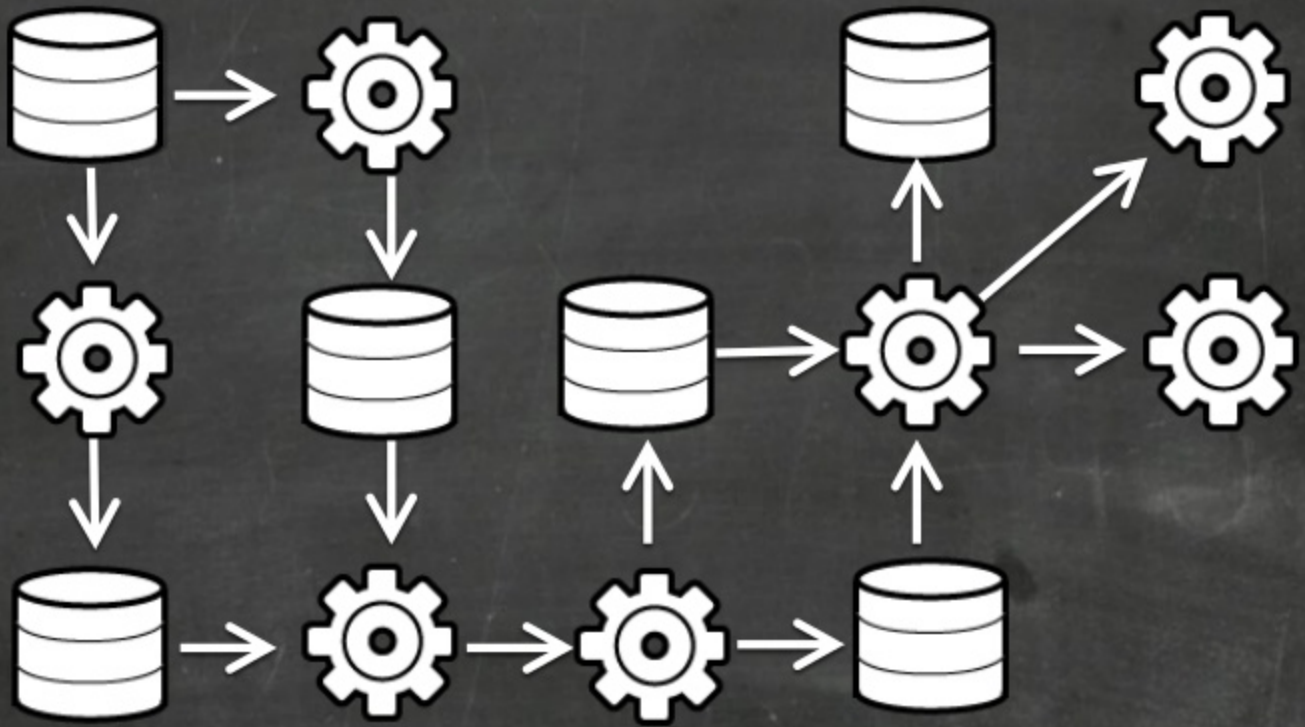
- Default 3 attempts/activity
- Configurable max-time to try (lateness)

Configurable Amazon SNS notifications for:

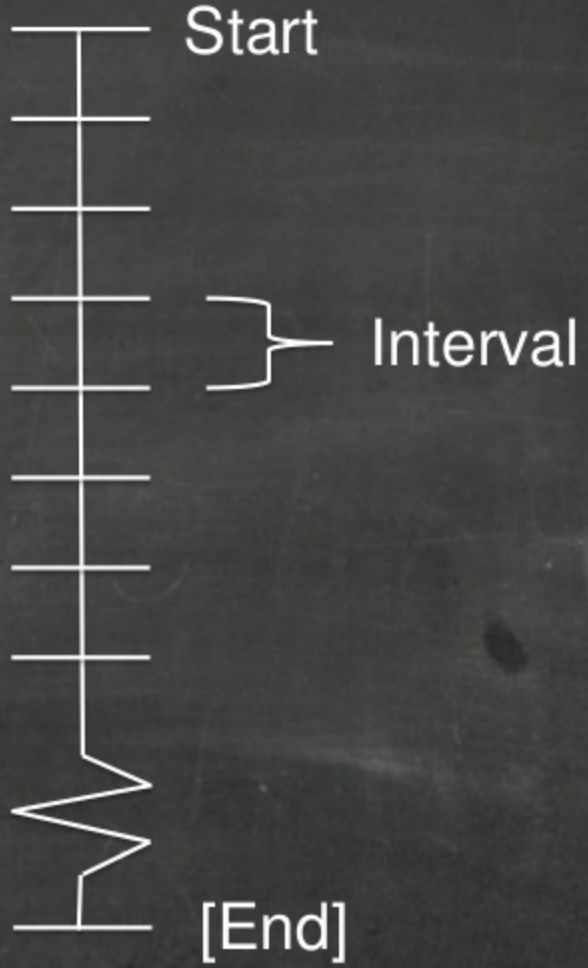
- Success
- Late
- Failure (attempts exhausted)


Data
Data
Data Stores
Compute Resources
Data Stores

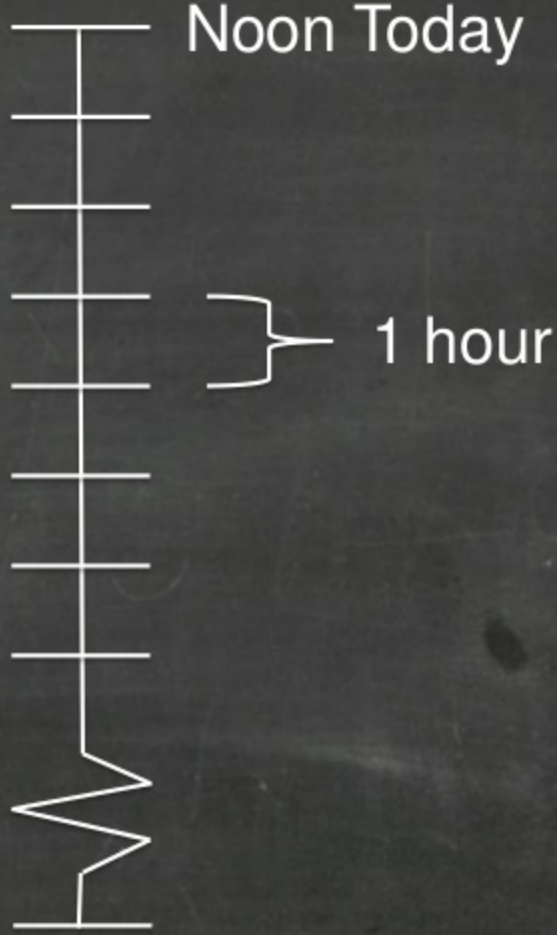
Activities & Datanodes chain



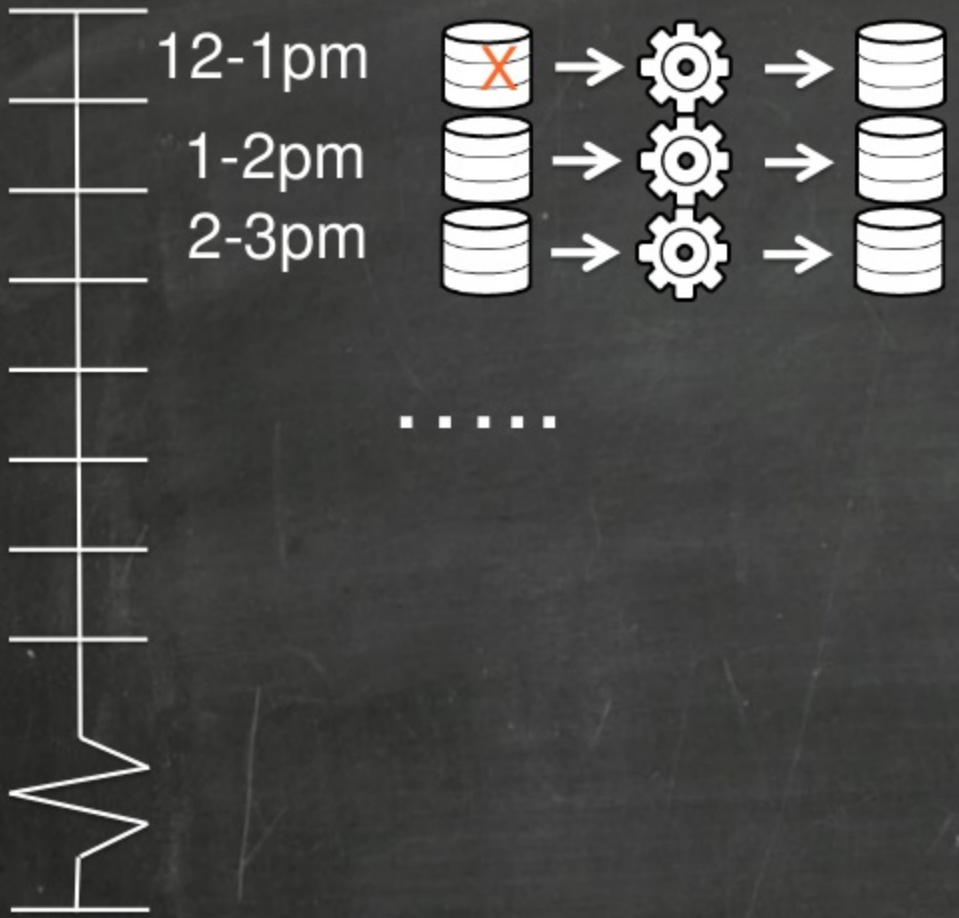
Scheduling



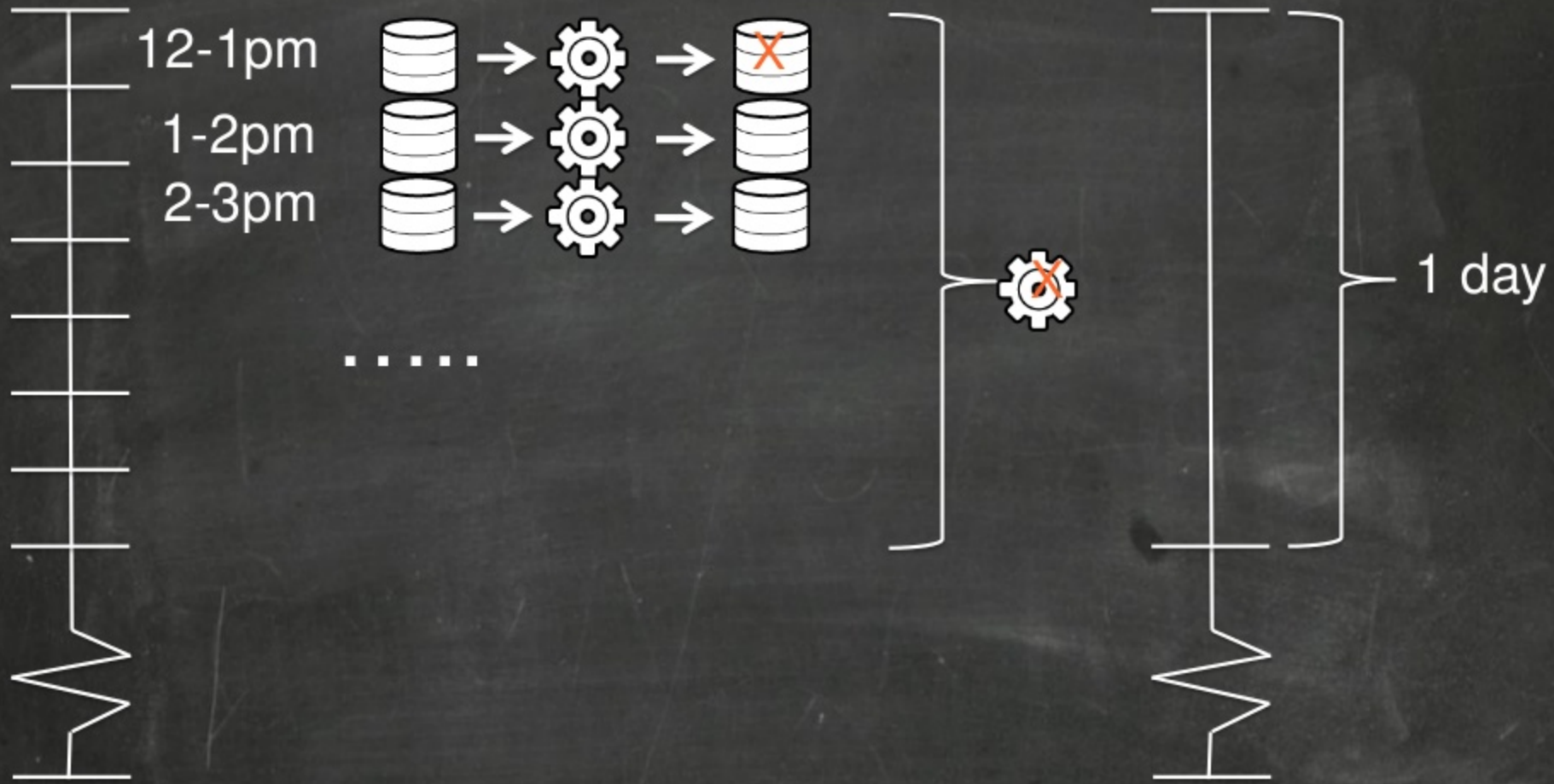
Scheduling



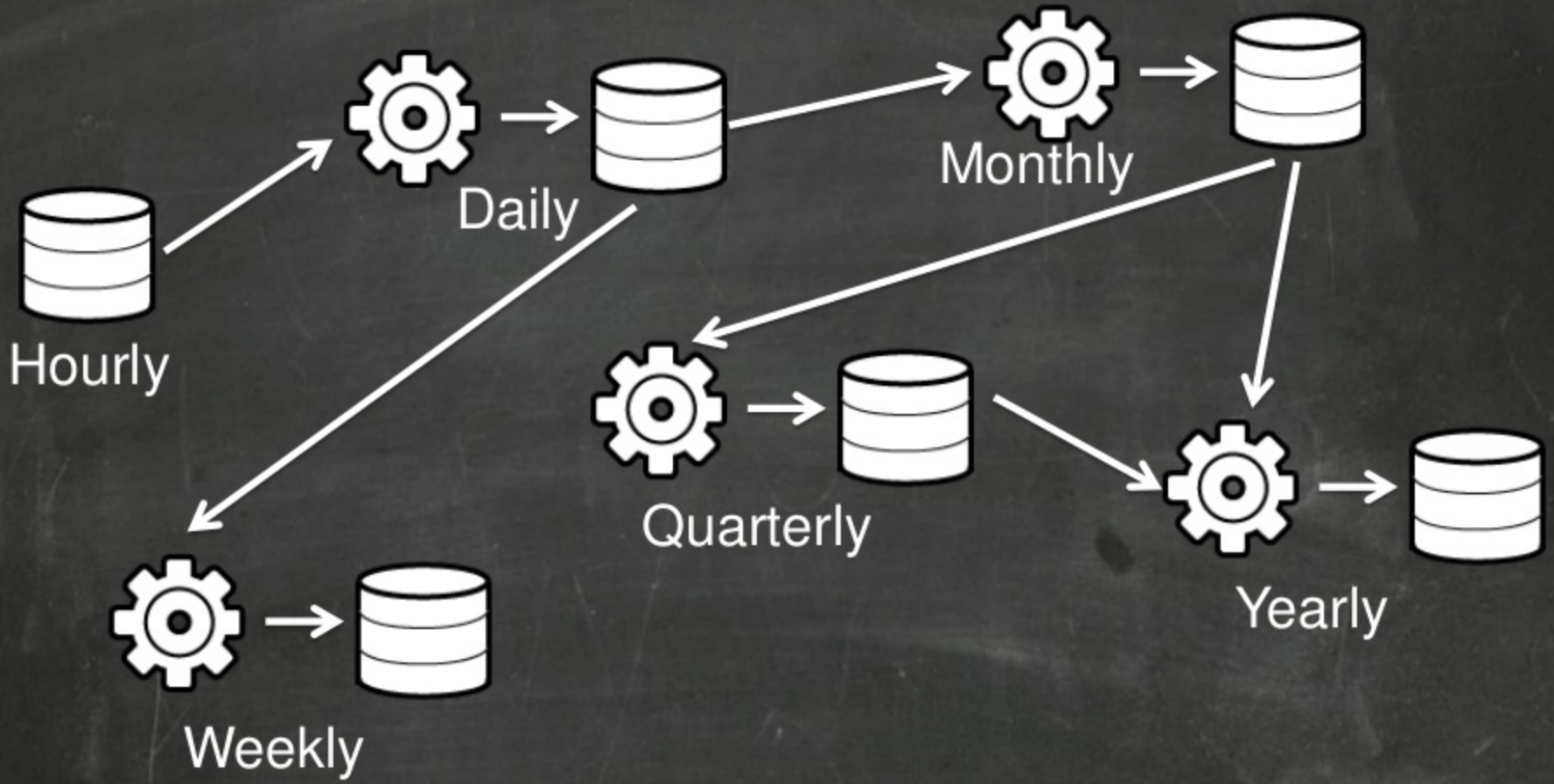
Scheduling



Scheduling: Rollups



Scheduling: Rollups



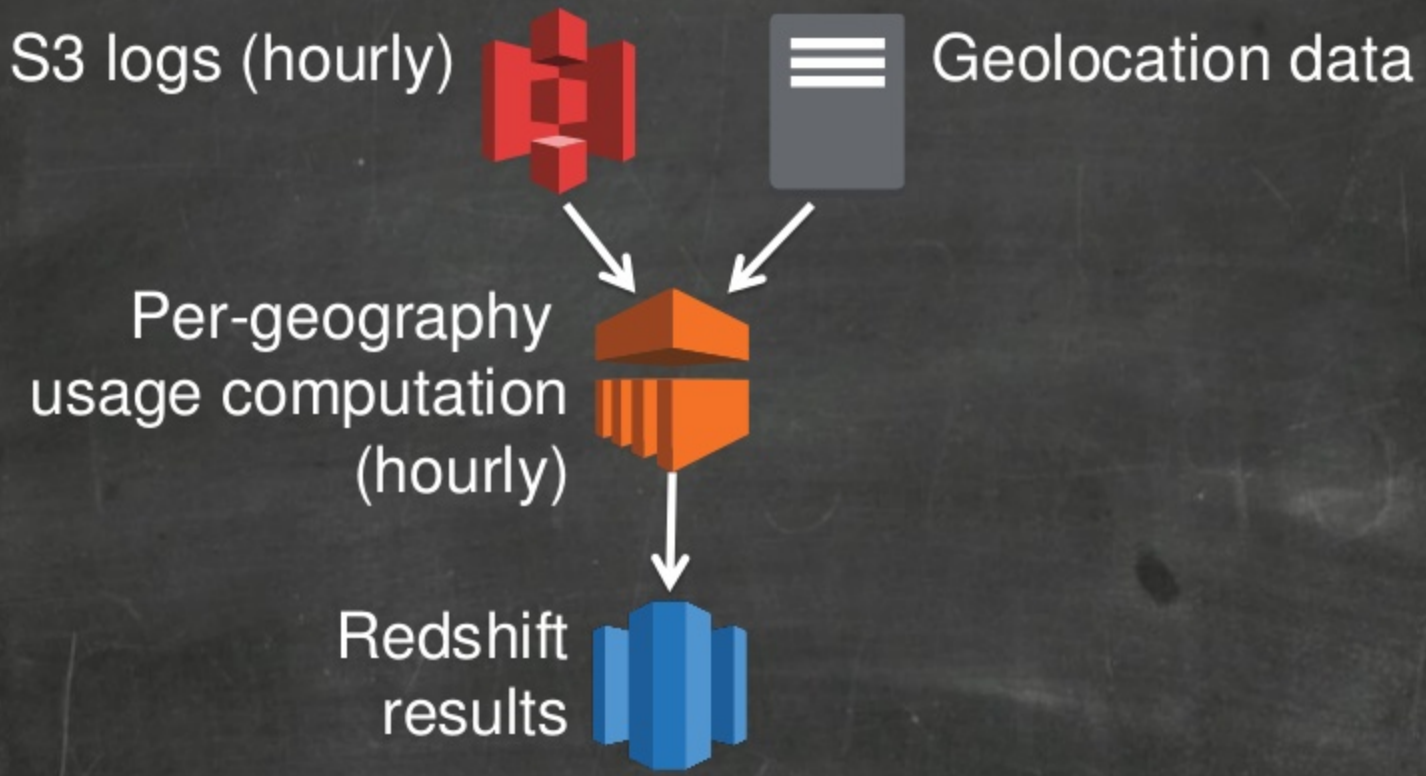
Example problem: Content targeting

- Managing a website with premium content
- Goal: improve business by better targeting content
- Second goal: don’t break/rewrite the website
- Third goal: lower costs!

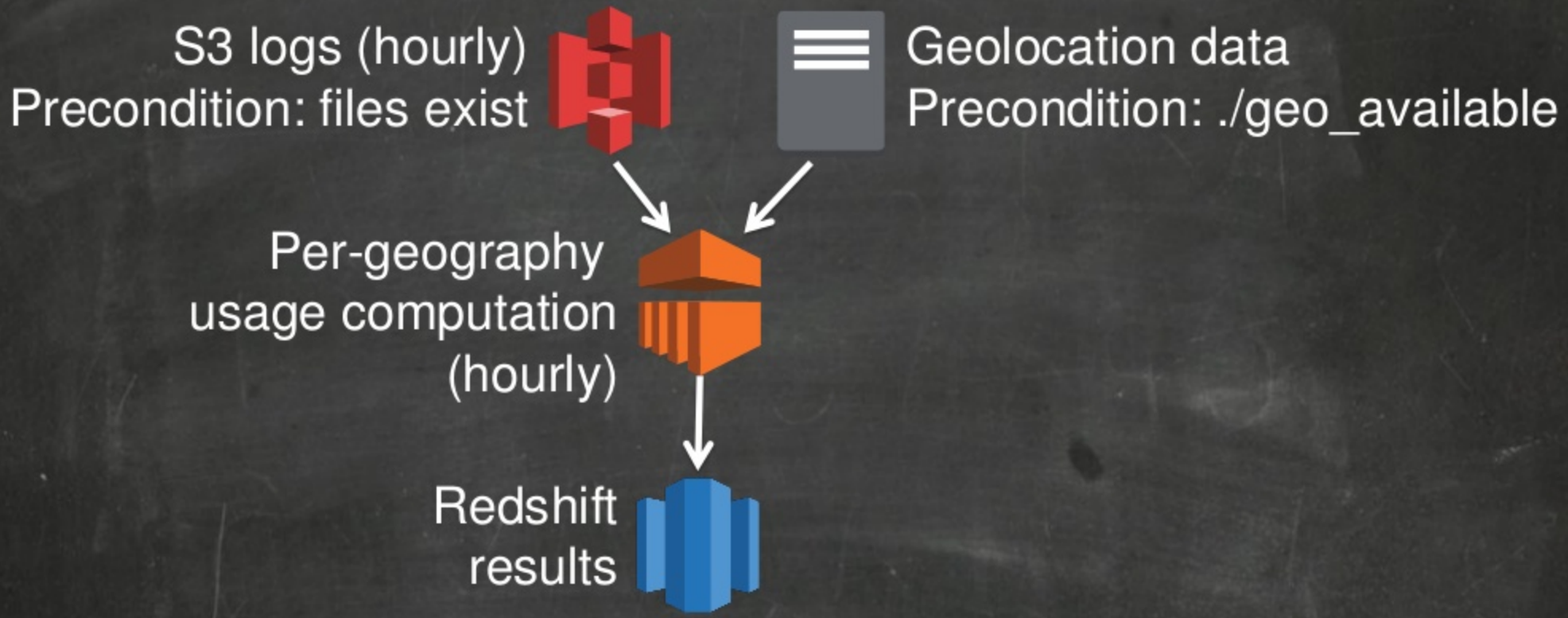
Targeting: the plan

1. Merge geolookup data with server logs to figure out what people look at by region
2. Merge event data with user demographics to figure out what types of users do what things
3. Put results into database for programmatic consumption by web servers
4. Display results as a reporting dashboard and generate automated reports for business owner

Targeting: Building the pipeline



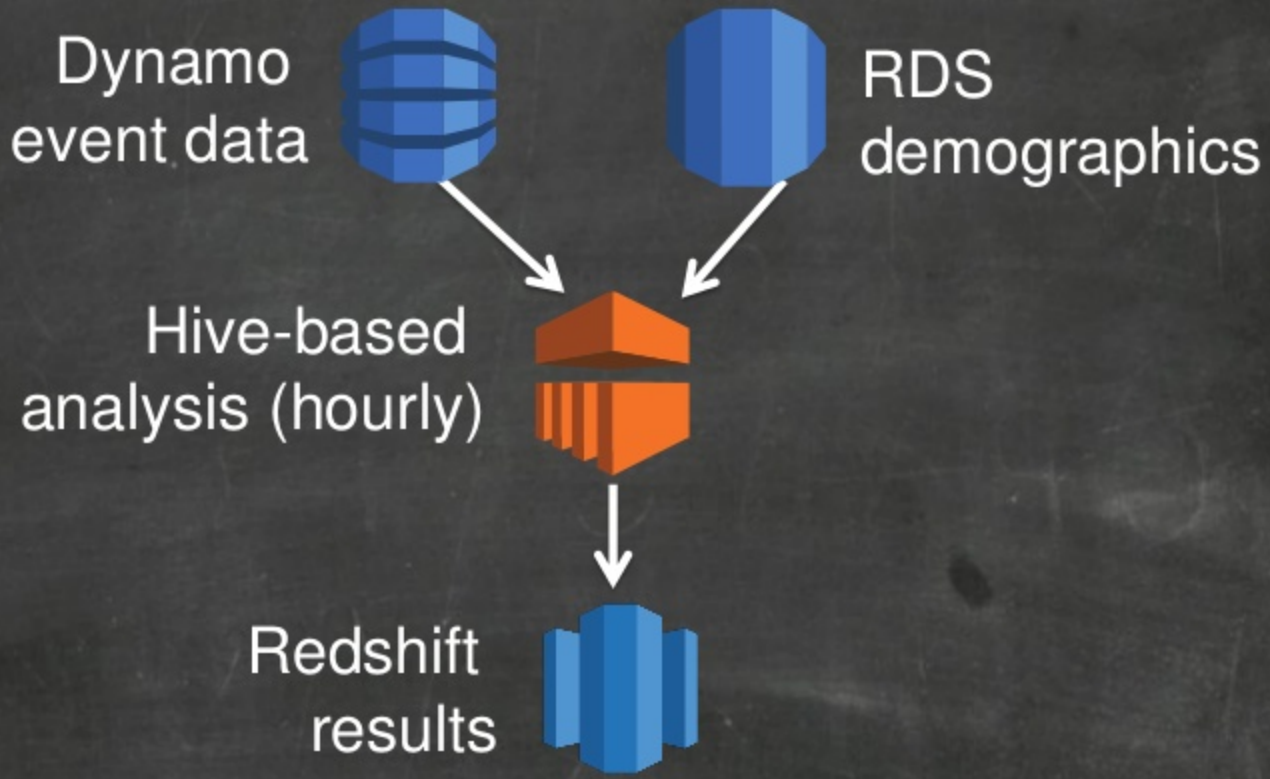
Targeting: Building the pipeline



Targeting: the plan

- ✓ Merge geolookup data with server logs to figure out what people look at by region
2. Merge event data with user demographics to figure out what types of users do what things
3. Put results into database for programmatic consumption by web servers
4. Generate automated reports for business owner

Targeting: Building the pipeline



Targeting: the plan

- ✓ Merge geolookup data with server logs to figure out what people look at by region
- ✓ Merge event data with user demographics to figure out what types of users do what things
- ✓ Put results into database for programmatic consumption by web servers

4. Display results as a reporting dashboard and generate automated reports for business owner

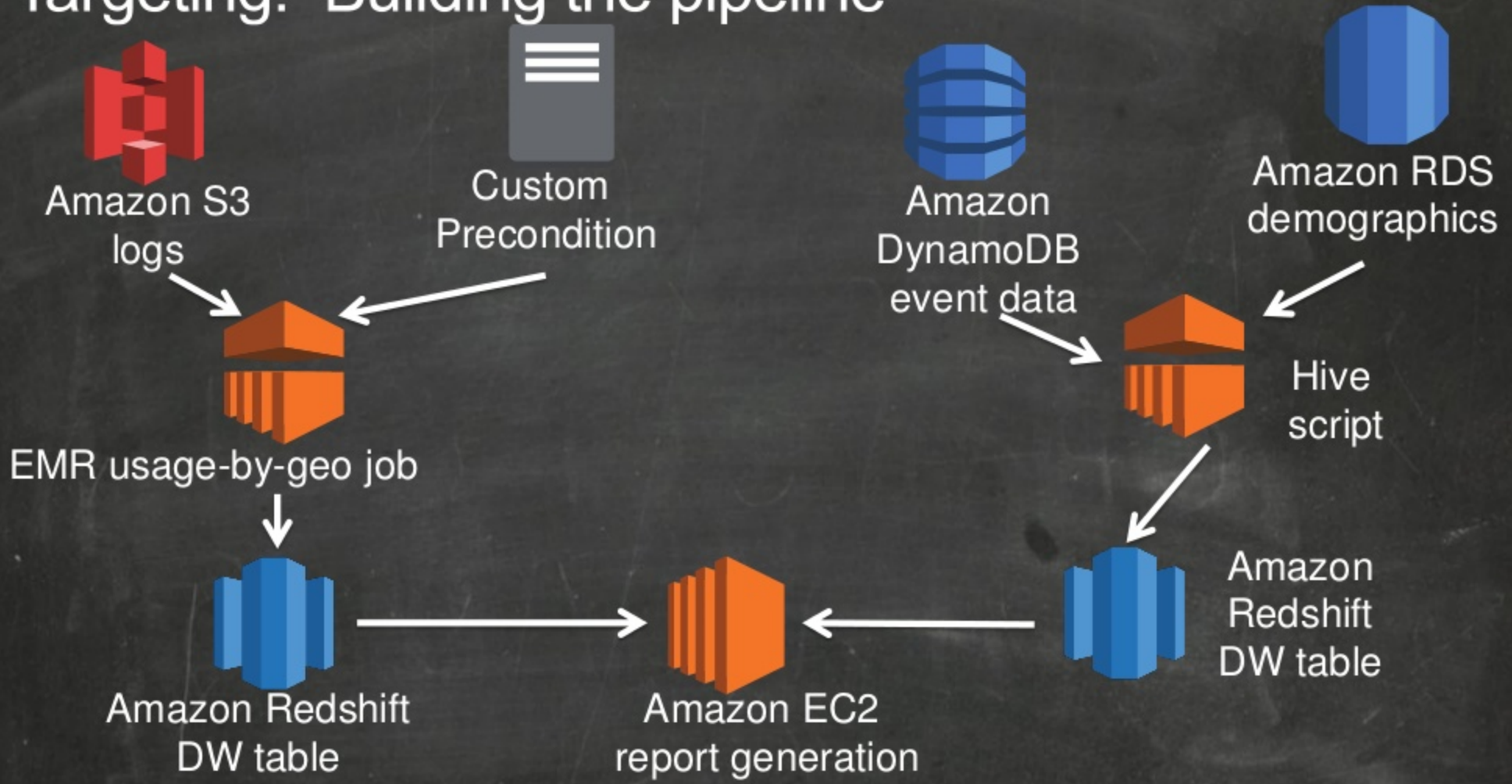
Targeting: Building the pipeline

#4: Merge results into reporting dashboard & emails

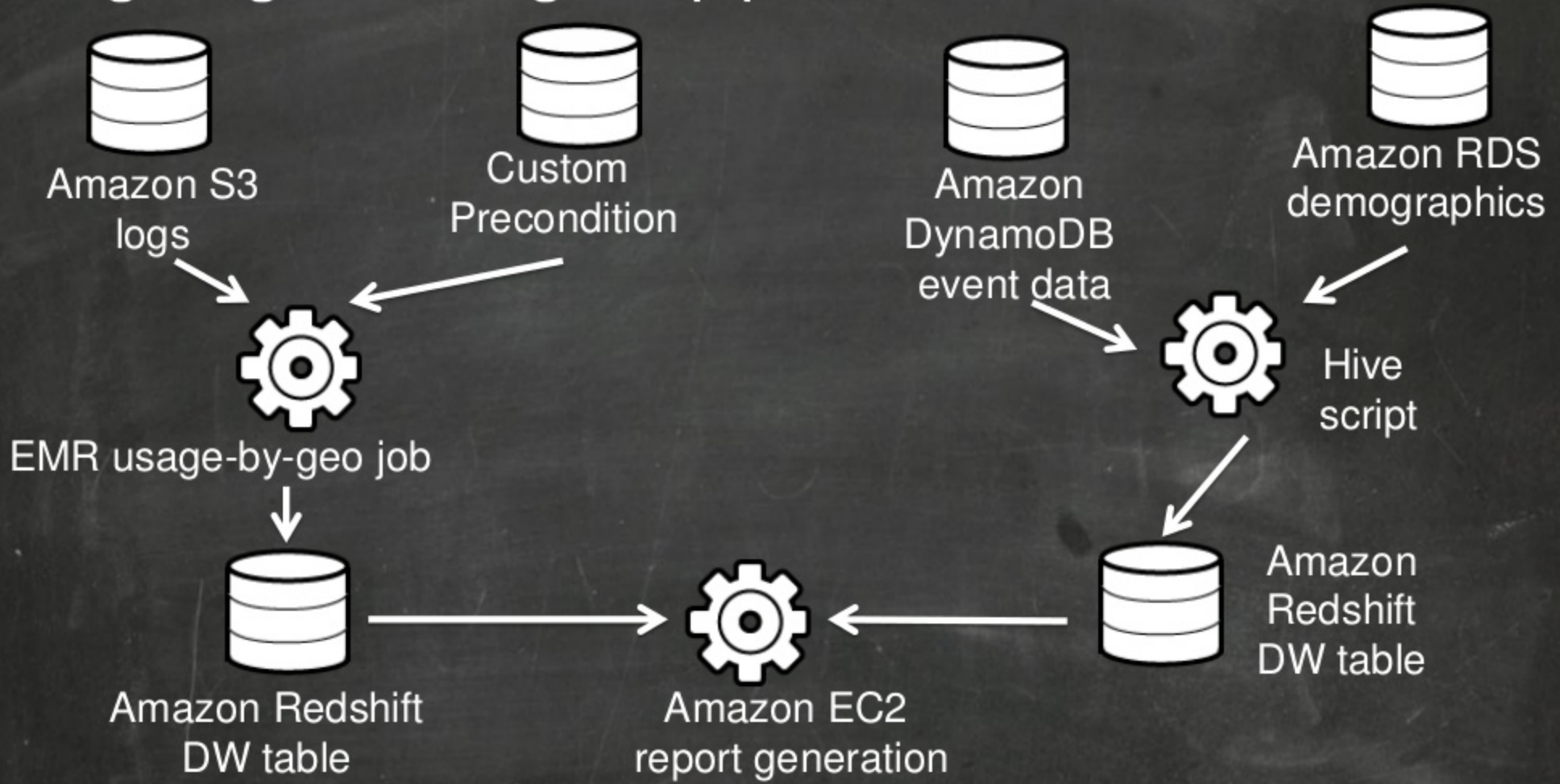
Targeting: the plan

- ✓ Merge geolookup data with server logs to figure out what people look at by region
- ✓ Merge event data with user demographics to figure out what types of users do what things
- ✓ Display results as a reporting dashboard and generate automated reports for business owner

Targeting: Building the pipeline



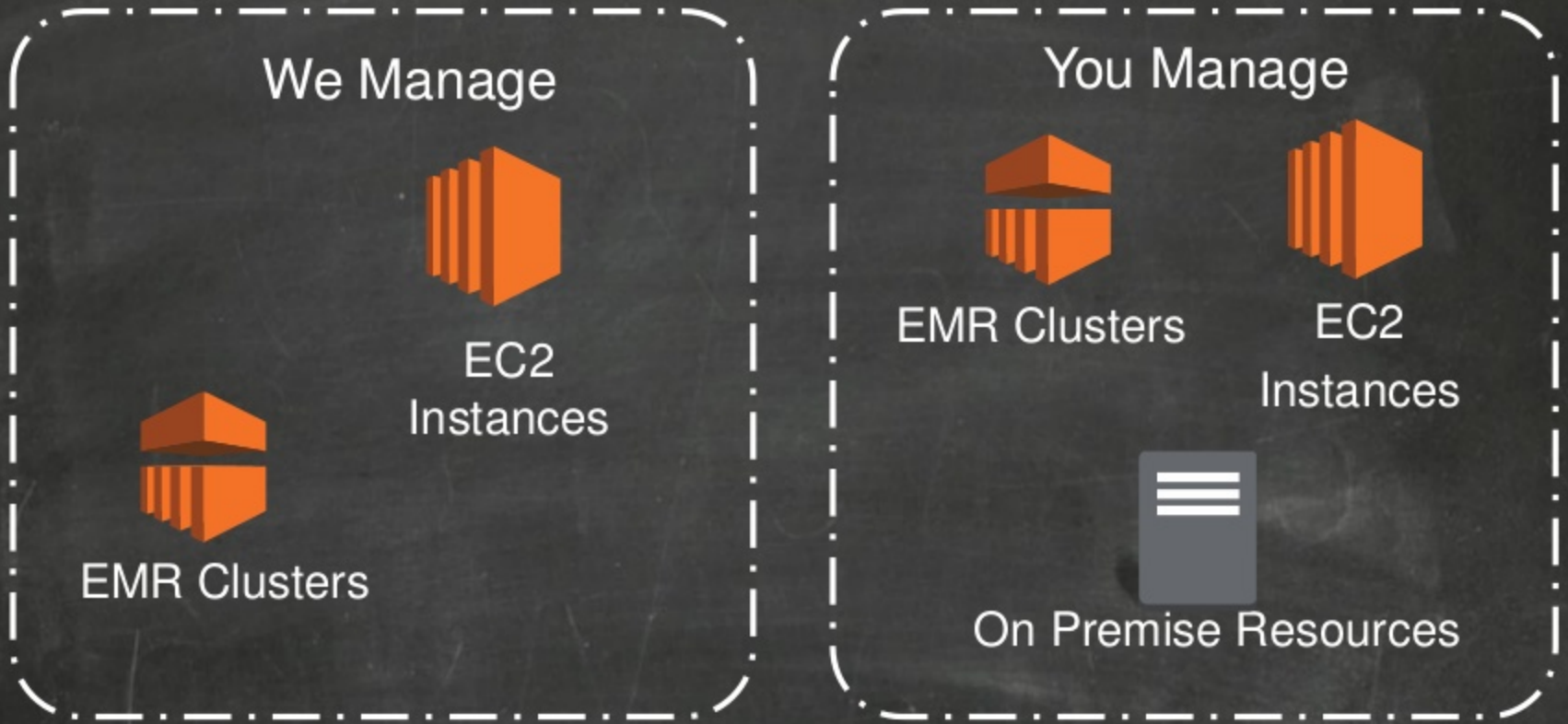
Targeting: Building the pipeline



Supported Activities

- Arbitrary Linux applications – anything that you can run from the shell
- Copies between different data source combinations
- SQL queries
- User-defined Amazon EMR jobs

Where can you run activities?



Data Pipeline–managed resources

- Data Pipeline launches the resource when your pipeline needs it
- Data Pipeline shuts it down when your pipeline is done with it
- Even easier to pay only for what you use

Resources you manage

- Control by installing Java-based task agent that polls the web service for work
- Run whatever manual configuration you need on the hosts
- Run your Data Pipeline activities on existing resources – zero incremental footprint
- Run on on-premise resources

On premise resources

- Task agent works on any *nix machine with Java & Internet access
- Agent includes logic to run local software or copy data between on-premise and AWS
- Agent activities are scheduled & managed just like any other activity

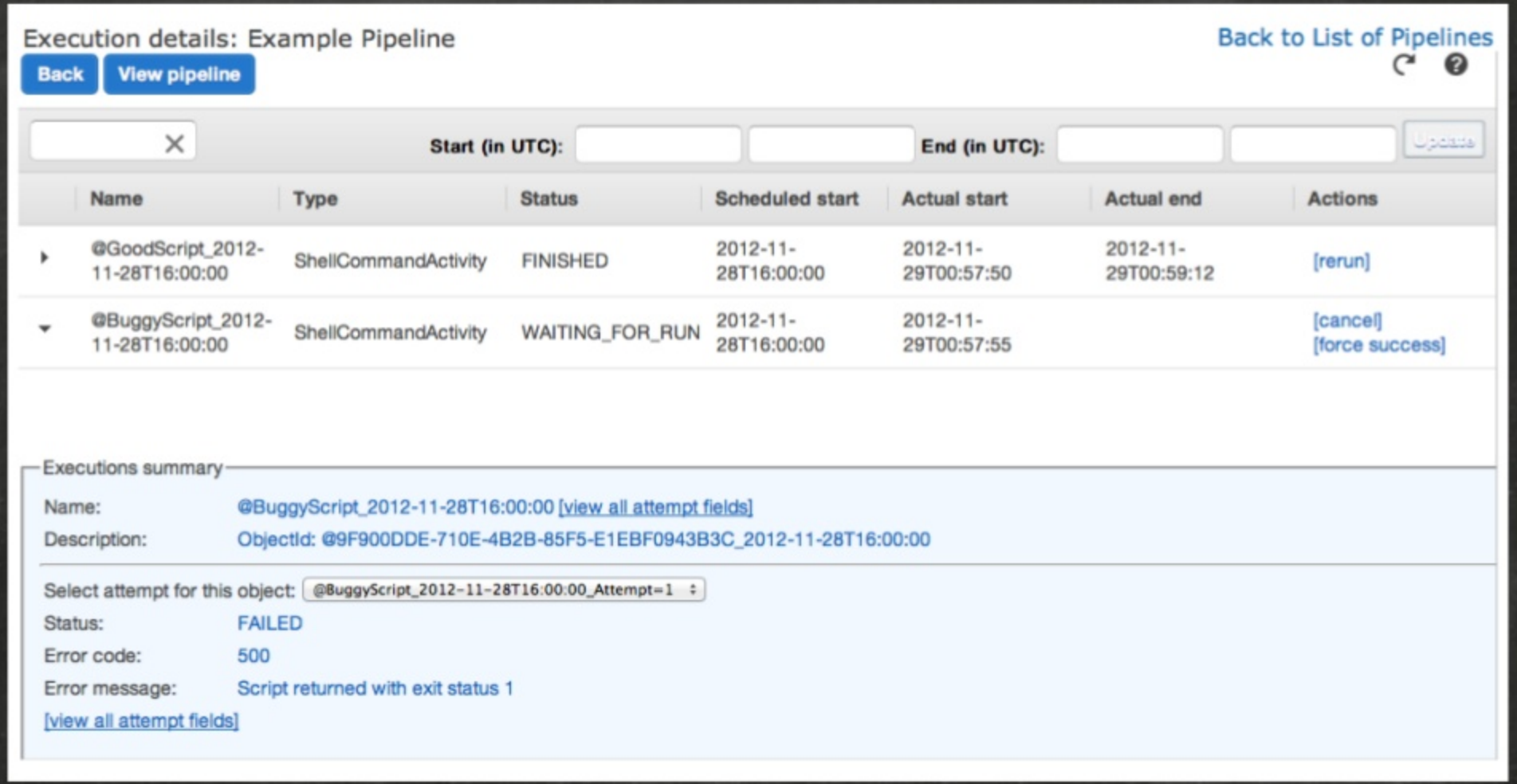
Console Templates

Console Provisioning

Provisioning (CLI/Pipeline Definition)

```
{
  "objects" : [
      {
        "name" : “My Copy”,
        "type" : “Copy Action”,
        “input”: {“ref” : “My RDS Data”},
        “output”: {“ref” : “My S3 Data”},
        ”runsOn” : {“ref”: “My Instance”},
        "schedule" : { "ref" : “My Schedule" } },
      {
        "name" : ”My Instance”,
        "type" : ”EC2Instance”,
        "instanceType" : "m1.small”,
        "schedule" : { "ref” : “My Schedule" } },
.....
}
```

Monitoring


Execution details: Example Pipeline
Back to List of Pipelines
Back
View pipeline
Start (in UTC):
End (in UTC):
Update
Name
Type
Status
Scheduled start
Actual start
Actual end
Actions
@GoodScript_2012-11-28T16:00:00
ShellCommandActivity
FINISHED
2012-11-28T16:00:00
2012-11-29T00:57:50
2012-11-29T00:59:12
[rerun]
@BuggyScript_2012-11-28T16:00:00
ShellCommandActivity
WAITING_FOR_RUN
2012-11-28T16:00:00
2012-11-29T00:57:55
[cancel]
[force success]
Executions summary
Name:
@BuggyScript_2012-11-28T16:00:00 [view all attempt fields]
Description:
ObjectId: @9F900DDE-710E-4B2B-85F5-E1EBF0943B3C_2012-11-28T16:00:00
Select attempt for this object:
@BuggyScript_2012-11-28T16:00:00_Attempt=1
Status:
FAILED
Error code:
500
Error message:
Script returned with exit status 1
[view all attempt fields]

Pricing/Free Tier

Cost is per activity or precondition in active pipelines

	On AWS	On Premise
High Frequency	$1/month	$2.50/month
Low Frequency	$.60/month	$1.50/month

Free Tier: 3 low frequency on-AWS preconditions & 5 low frequency on-AWS activities

The Goals

- ✓ What's the problem?
- ✓ How does AWS Data Pipeline solve it?
- ✓ Technical examples

Current Status

- Currently in private beta
- Public beta coming soon!

We are sincerely eager to hear your **feedback** on this presentation and on re:Invent.

Please fill out an evaluation form when you have a chance.