Amazon S3, Ruby and Rails

Scott Patten
spattendesign.com

spatten design
Building web apps and web app teams

Amazon Web Services

- Jeff Bezos wants other people to be able to create web apps without the pain he went through
- No fixed costs
- Your costs scale with your usage and (hopefully!) your profits.

Amazon Web Services

- Simple Storage Service (S3)
- Elastic Compute Cloud (EC2)
- Mechanical Turk
- Flexible Payments Service
- Simple Queue Service
- Alexa Web Services

Simple Storage Service

- on-line storage using Amazon's hardware
- Simple
- Reliable
- Scalable
- Flexible
- Pretty cheap

Costs

Storage

$0.15 per GB-Month of storage used

Data Transfer

$0.10 per GB - all data transfer in
$0.18 per GB - first 10 TB / month data transfer out
$0.16 per GB - next 40 TB / month data transfer out
$0.13 per GB - data transfer out / month over 50 TB

Requests

$0.01 per 1,000 PUT or LIST requests
$0.01 per 10,000 GET and all other requests*
* No charge for delete requests

Use Cases

Backup

Personal
- music
- photos
- documents

Rails application
- databases
- svn repositories
- user generated content

Use Cases

Asset host

- Store the public directory of your Rails app on S3
- make it readable by anyone
- point something like assets.plotomatic.com at it

Use Cases

Streaming large Files

Use Cases

Zunior

- Delivers their MP3s from S3

SmugMug

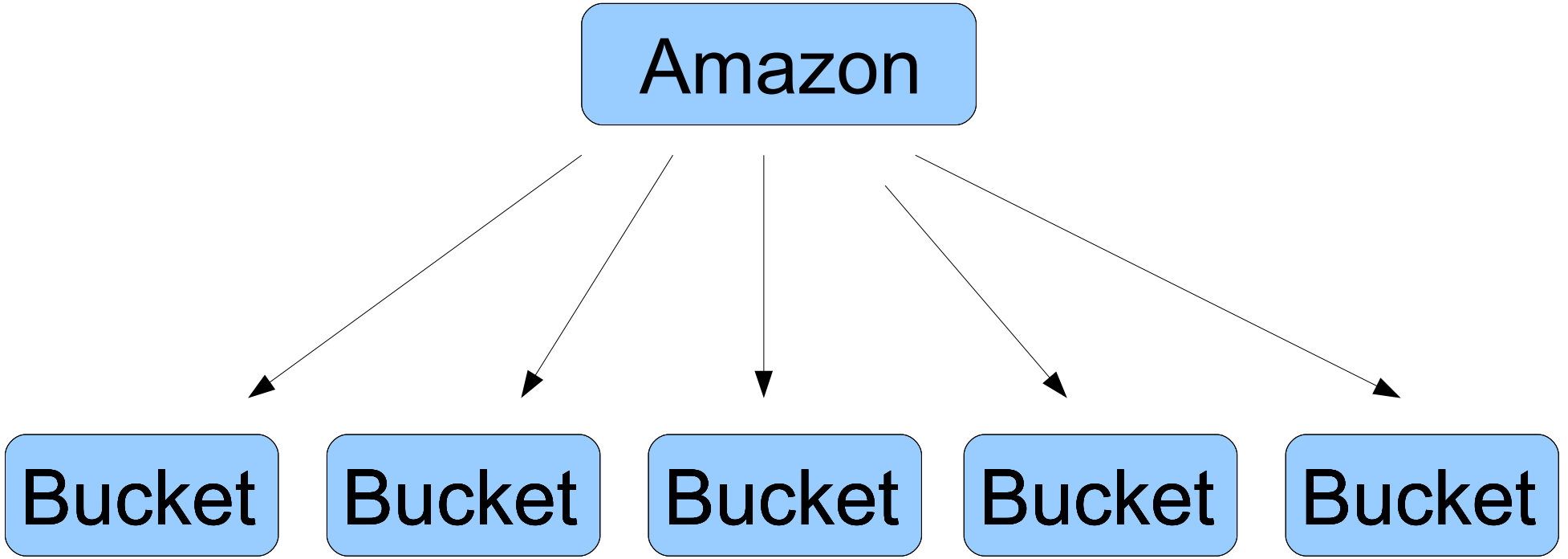
- estimates they save $500k / year
- > 100 TB of data stored

37Signals

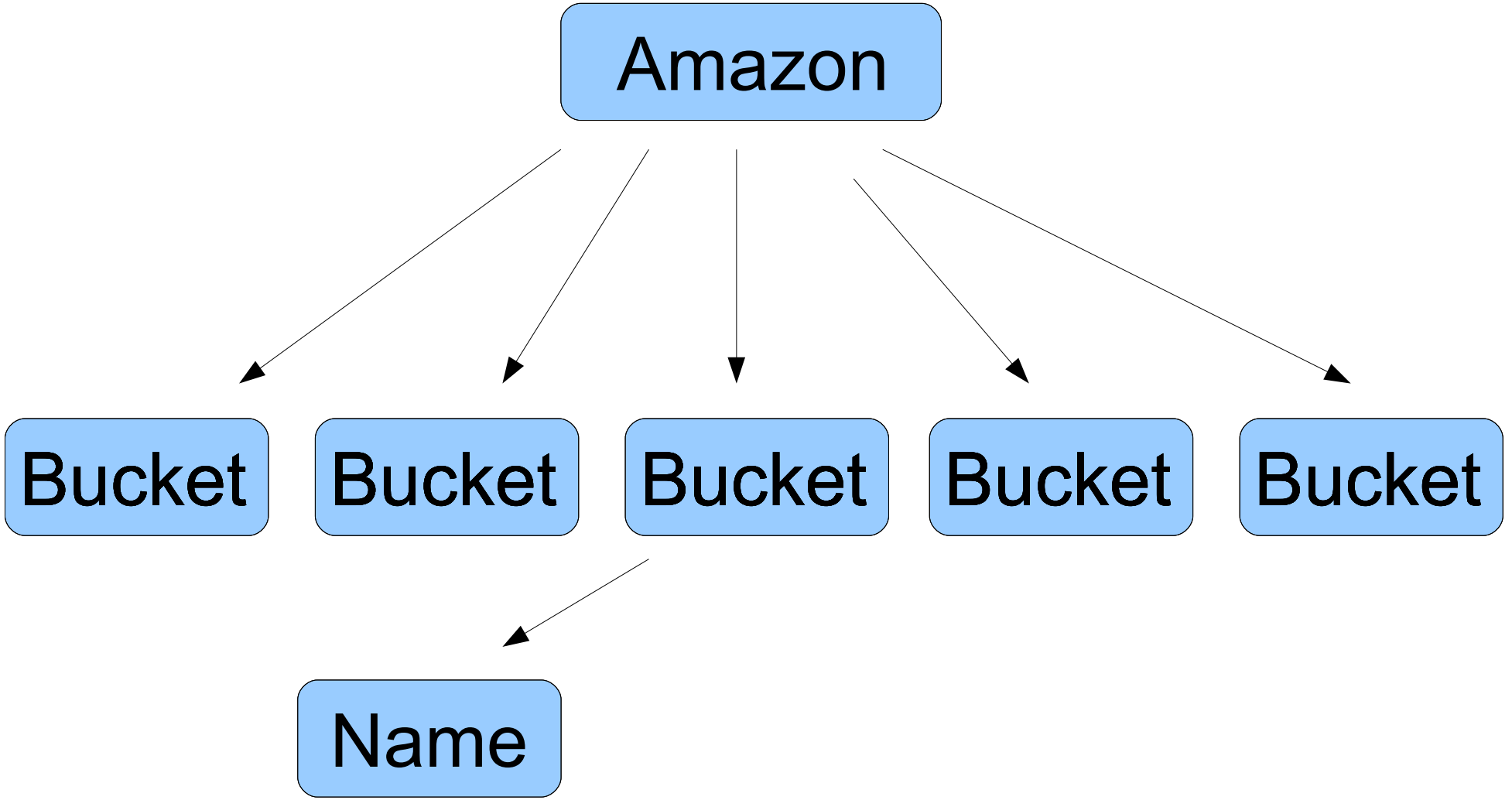
- All campfire uploads get uploaded to S3
- > 1 TB data stored

S3 Architecture

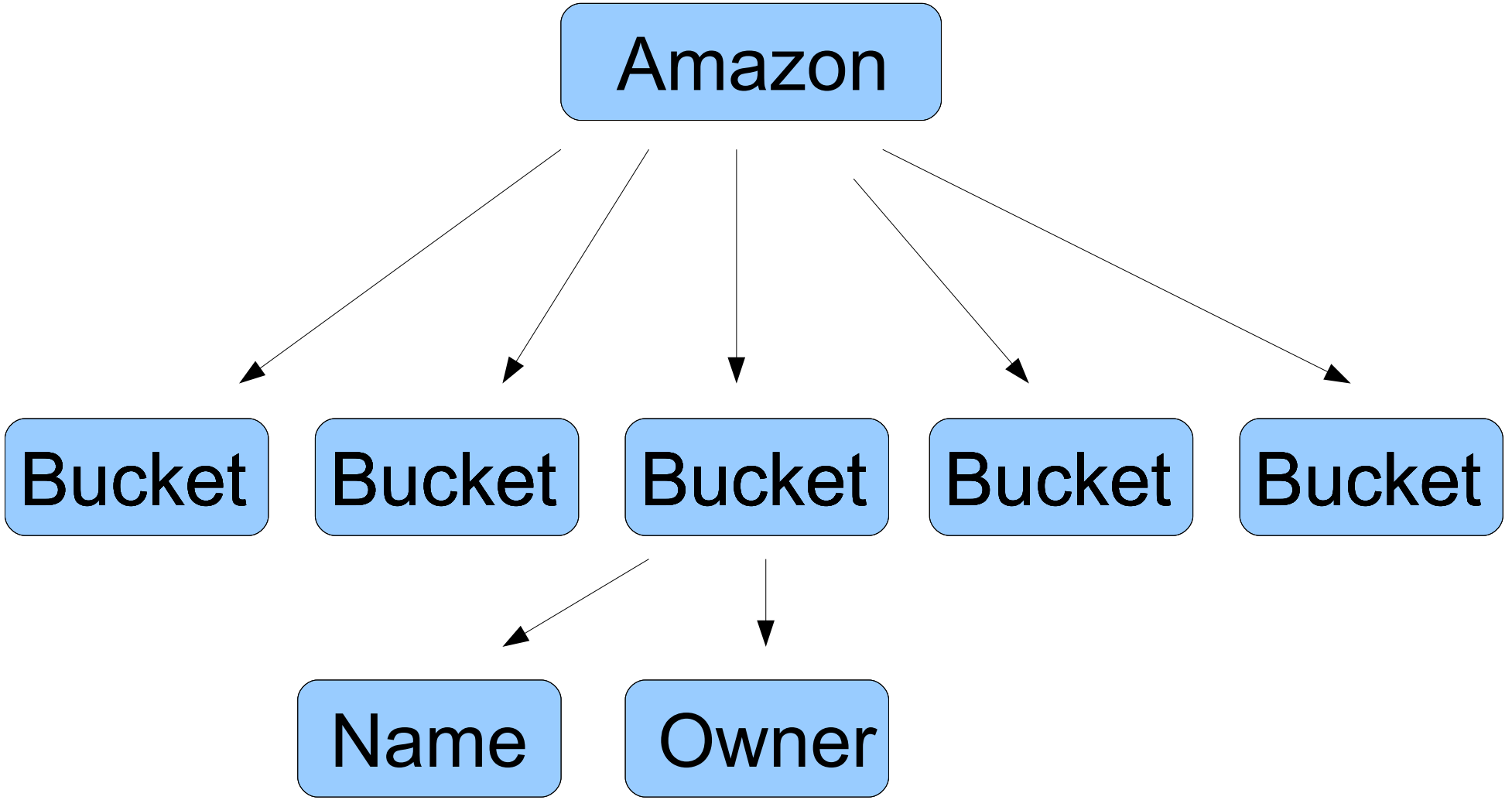
S3 Architecture



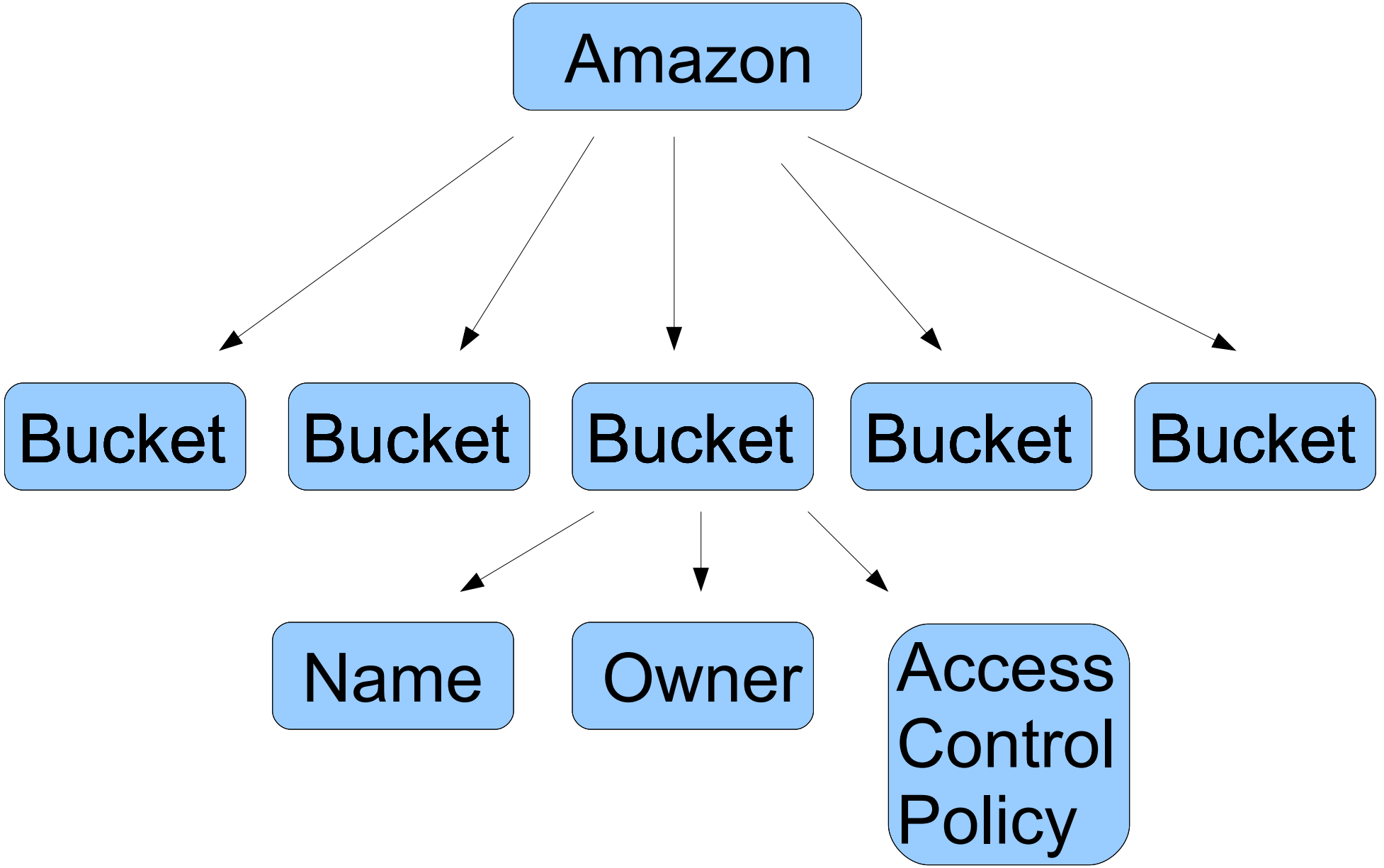
S3 Architecture



S3 Architecture

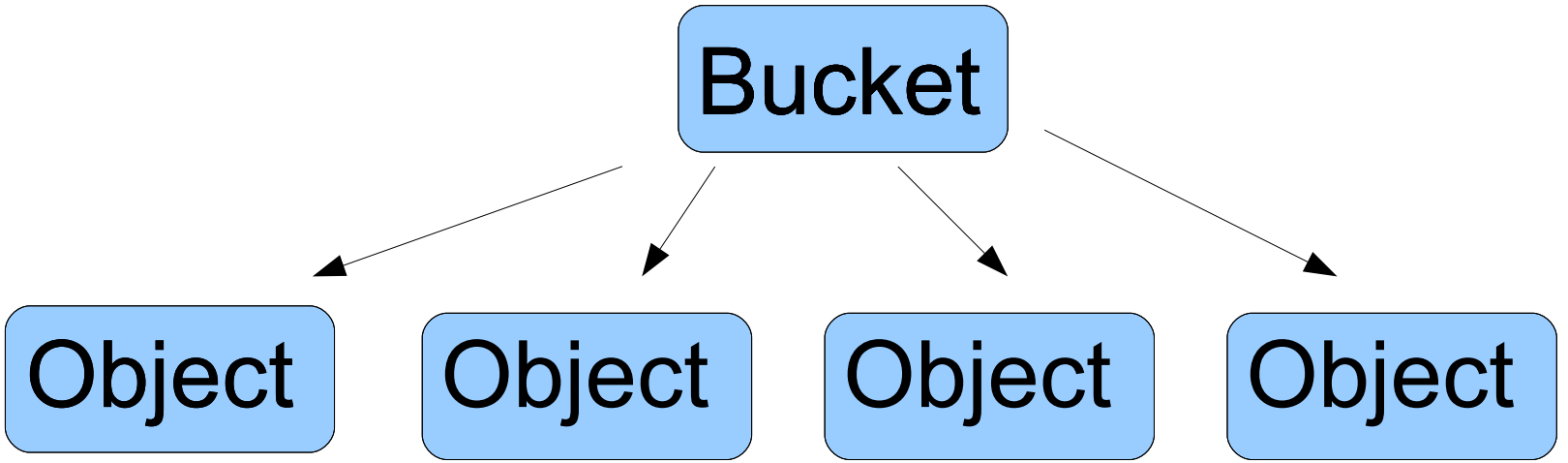


S3 Architecture

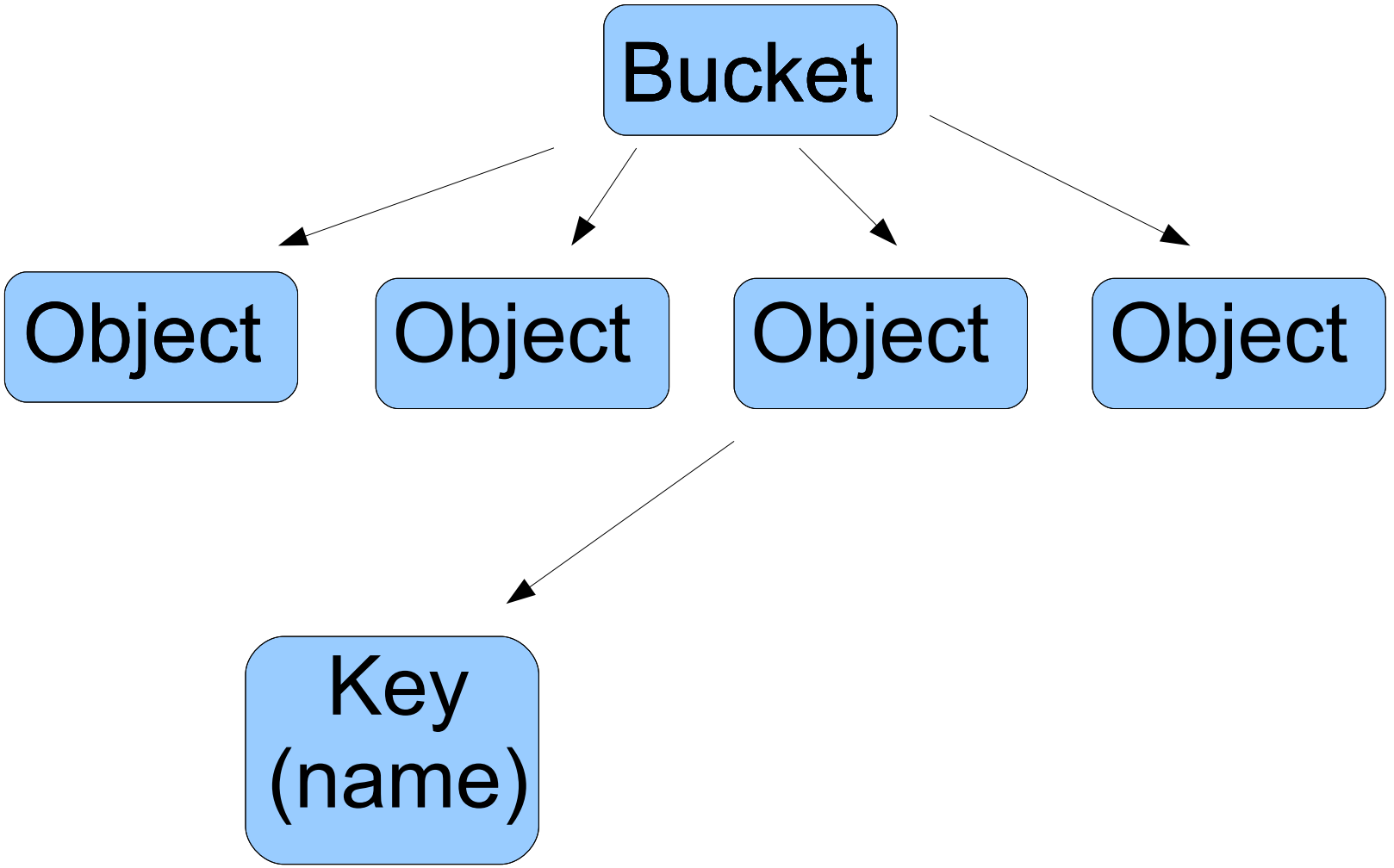


S3 Architecture

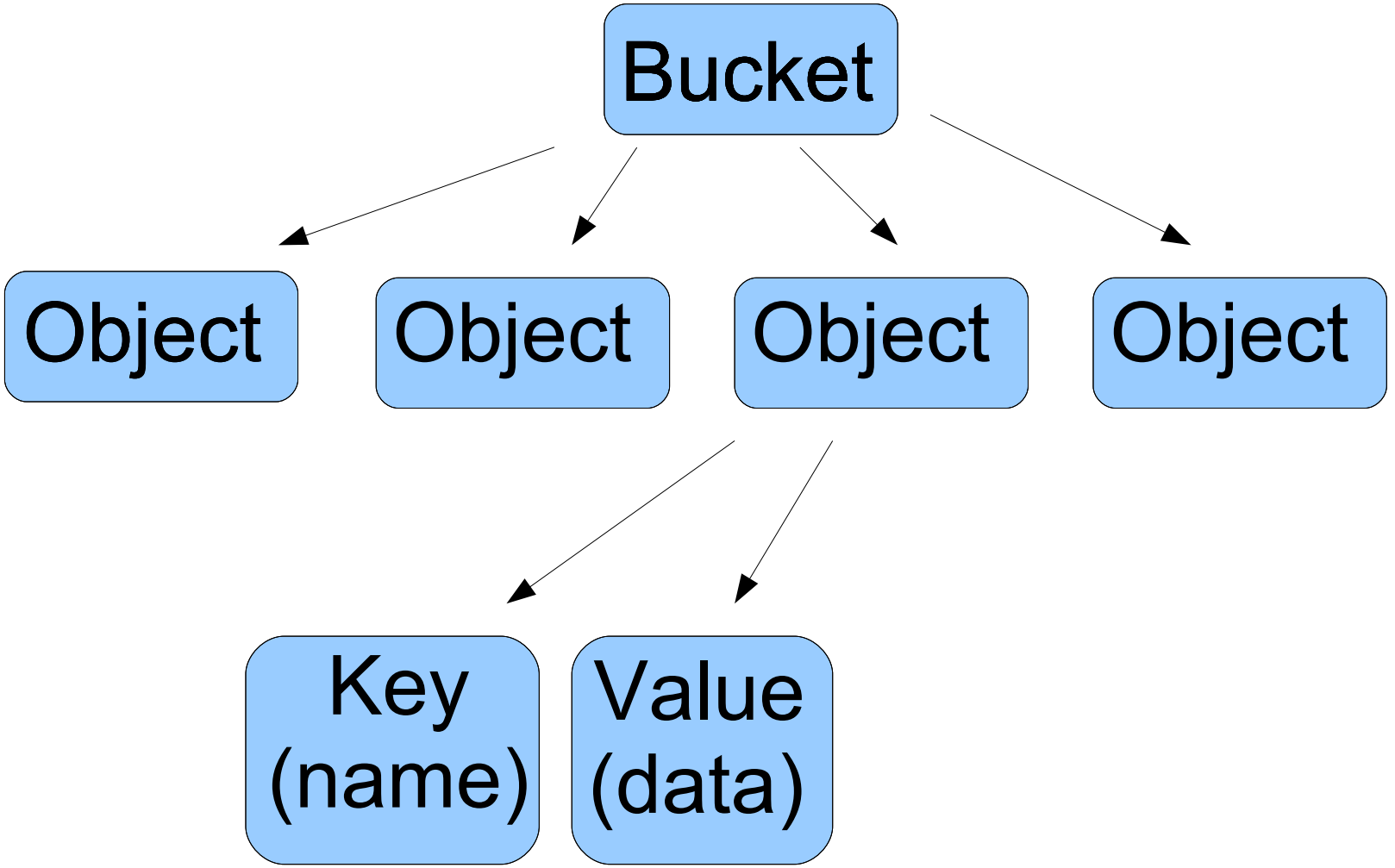
S3 Architecture



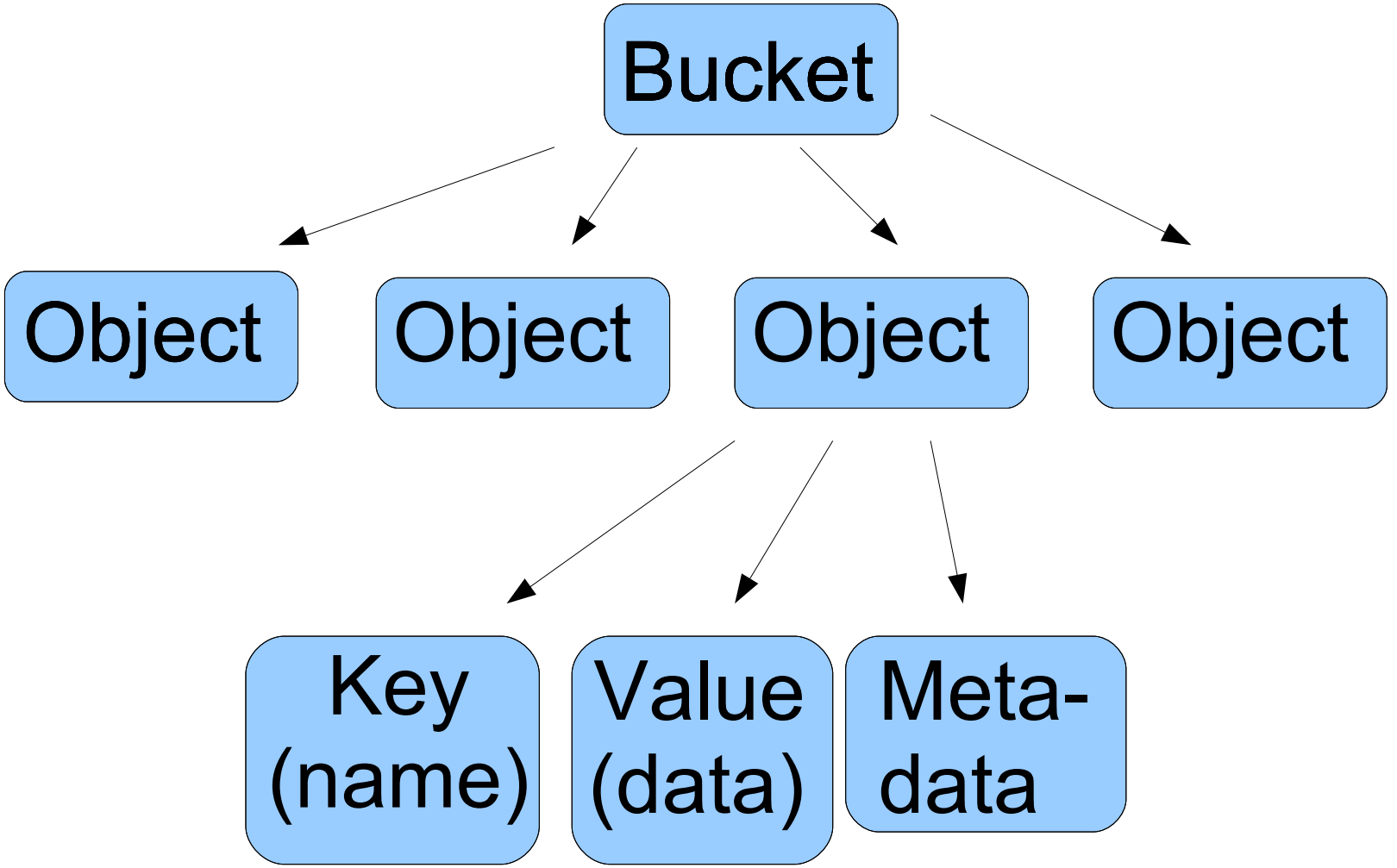
S3 Architecture



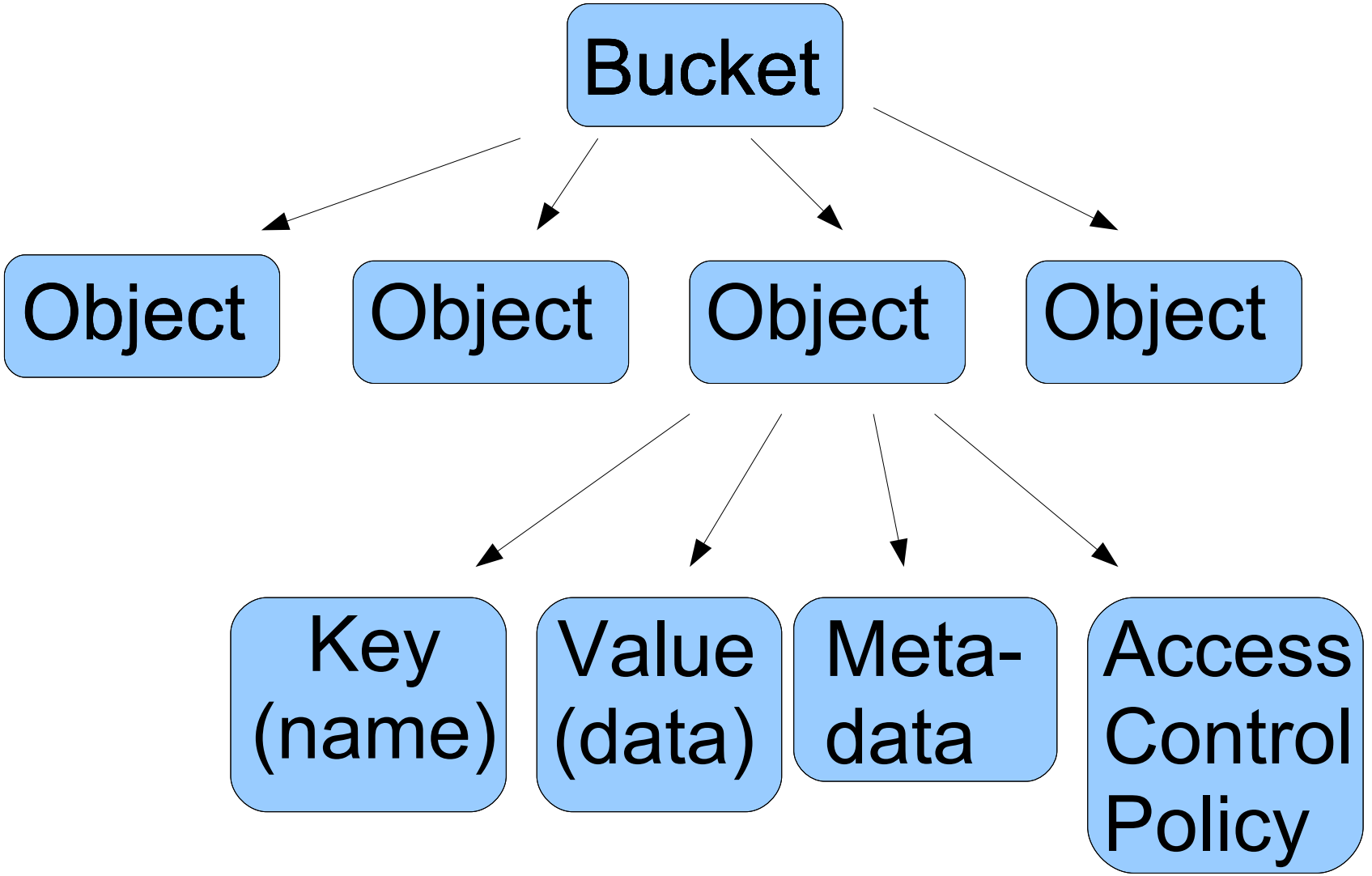
S3 Architecture



S3 Architecture



S3 Architecture

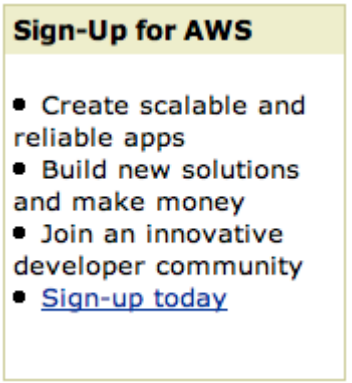


S3 Architecture

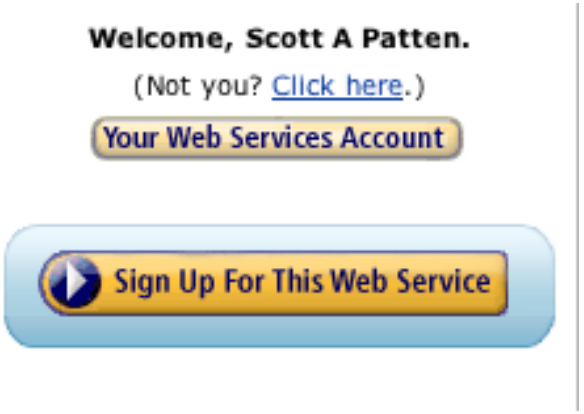
No Directory Structure

Getting started with S3

Sign up at http://aws.amazon.com



Sign up for S3

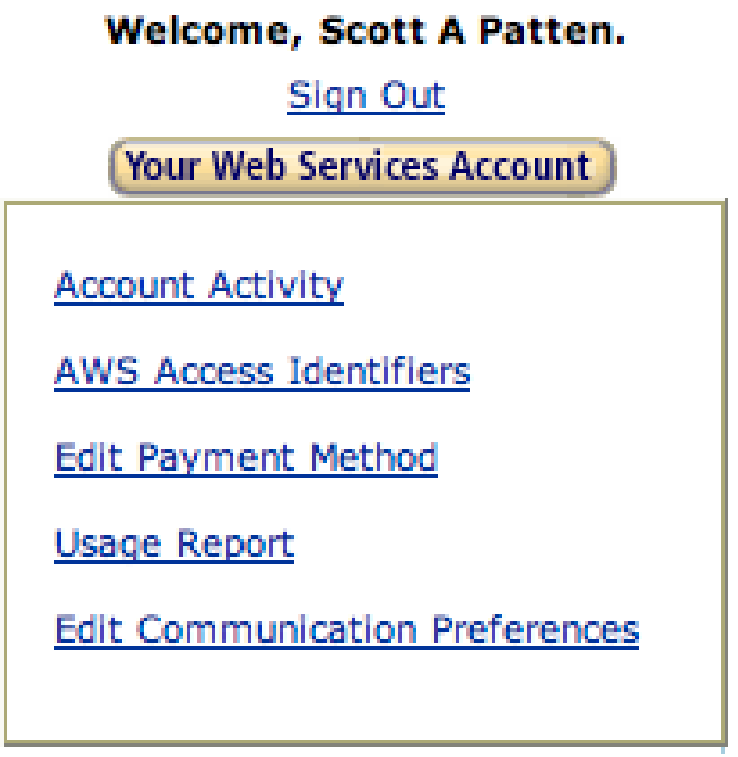


Getting started with S3

Get your keys

Hover over the 'web services account' button
Click on 'AWS Access Identifiers'

You need two keys:
Access key
Secret access key



Getting started with S3

Install the AWS-S3 gem

sudo gem install aws-s3

http://amazon.rubyforge.org

Getting started with S3

Set up your environment variables

```
export AMAZON_ACCESS_KEY_ID='abcdefghijklmnop'

export AMAZON_SECRET_ACCESS_KEY='12367891012345'
```

S3SH Demo

Accessing your files

Normal method

key vampire.jpg

bucket spatten_s3demo

URL http://s3.amazonaws.com/spatten_s3demo/vampire.jpg

Accessing your files

Bucket First

key vampire.jpg

bucket spatten_s3demo

URL http://spatten_s3demo.s3.amazonaws.com/vampire.jpg

Accessing your files

CNAME Method

key vampire.jpg

bucket photos.spattendesign.com

URL http://photos.spattendesign.com/vampire.jpg

For this to work, you need to create a CNAME entry that points photos.spattendesign.com to s3.amazonaws.com

Cool, how did they make irb do that?

s3sh

```ruby
#!/usr/bin/env ruby
s3_lib   = File.dirname(__FILE__) + '/../lib/aws/s3'
setup    = File.dirname(__FILE__) + '/setup'
irb_name = RUBY_PLATFORM =~ /mswin32/ ? 'irb.bat' : 'irb'

exec "#{irb_name} -r #{s3_lib} -r #{setup} --simple-prompt"
```

setup.rb

```ruby
#!/usr/bin/env ruby
if ENV['AMAZON_ACCESS_KEY_ID'] &&
   ENV['AMAZON_SECRET_ACCESS_KEY']
  AWS::S3::Base.establish_connection!(
    :access_key_id     => ENV['AMAZON_ACCESS_KEY_ID'],
    :secret_access_key => ENV['AMAZON_SECRET_ACCESS_KEY']
  )
end

require File.dirname(__FILE__) + '/../test/fixtures'
include AWS::S3
```

Example: S3Syncer

A script that
synchronizes a local directory
with an S3 bucket

S3, Rails and Asset Hosting

S3 setup

- Create a bucket called assets.example.com
- Point the bucket at s3.amazonaws.com
- Put everything in /public in the bucket

S3, Rails and Asset Hosting

Rails setup

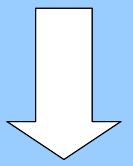
add the following line to RAILS_ROOT/config/environments/production.rb:

```
config.action_controller.asset_host = "http://assets.example.com"
```

S3, Rails and Asset Hosting

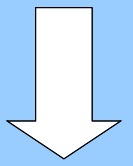
Rails setup

```
<%= image_tag("image.png") -%>
```



```
<img src="http://assets.example.com/images/image.png" alt="image" />
```

```
<%= javascript_include_tag  'prototype' %>
```



```
<script src="http://assets.example.com/javascripts/prototype.js"
 type="text/javascript"></script>
```

S3, Rails and Asset Hosting

Rails setup

Install the synch_s3_asset_host plugin

http://spattendesign.com/2007/11/6/synching-your-amazon-s3-asset-host-using-capistrano

One more cool thing

You can generate a Bit Torrent file
for any S3 Object by accessing it like this:

http://s3.amazonaws.com/bucket/file?torrent

This will automatically generate the torrent

You only pay when the bit torrent clients
download directly from Amazon

The end

any questions?