

CLOUD DEVELOPMENT WITH APACHE CAMEL AND AMAZON WEB SERVICES

@robinhowlett

ABOUT ME

- **Robin Howlett**
- Senior Architect at Silver Chalice (Boulder, CO)
- Lead for Advanced Media Platform

ABOUT SILVER CHALICE

- Work with sports and media entities to build digital businesses
- Focus on mobile, digital sales, video, social and digital networks, and new media technologies
- Backed by Chicago White Sox ownership
- Clients are major media and sports companies

WHY CAMEL + AWS?

- Startup: small team, big project, quick deadline
- Existing products in market; needed to scale on day 1
- Complex data ecosystem; variety of providers, data formats and distribution methods
- Needed expressive, powerful integration framework

STYLE

- Brief overviews for context
- Technical detail over broad descriptions
- Code-first approach; fewer slides
- Increasing complexity
- Personal experience
- Q&A best for business and management questions

AGENDA

- Introduction to Amazon Web Services
- Details of selected AWS products and architectures
- Overview of AWS SDK for Java and AWS Toolkit for Eclipse
- Demo: Camel's AWS components
- Scaling Demo: Camel and AWS Elastic Beanstalk
- Cloud Routing Demo: Camel and AWS Simple Workflow (SWF)
- Q&A

AMAZON WEB SERVICES

- Pay-as-you-go
- Scalable-on-demand resources
- HA
- Global
- Computing, Storage, Middleware, Database, Networking, CDN, Security, DevOps
- Open source technologies prominent
- Web service APIs and SDKs
- IaaS, PaaS, SaaS



AWS MAP

IMPLICATIONS

- Rapid engineering
- Smaller teams with greater scope
- Budgeting
- SLAs
- Security
- Non-traditional engineering
- Complexity

AWS PRODUCTS & SERVICES

- EC2 & EBS
- S3 & CloudFront
- RDS, SimpleDB & DynamoDB
- SQS
- SNS
- SWF
- IAM
- EMR
- SES
- Elastic Beanstalk
- ELB & Auto-Scaling
- Route 53
- ElastiCache
- CloudSearch
- CloudFormation
- VPC
- CloudWatch
- AMIs

EC2

- Micro-instances to high-computing clusters
- On-demand, spot or reserved pricing options
- EBS vs Instance Stores: implications
- Templating with AMIs and CloudFormation
- Security: IAM users, key pairs and security groups
- Load Balancing and Elastic IPs

S3

- What's cooler than a trillion objects?
- Buckets of objects, uniquely identified
- REST & SOAP interfaces; HTTP and BitTorrent
- SLA: 99.9% "uptime"; 99.9999999% durability
- Static content, publicly accessible
- Combine with CloudFront for an instant global CDN

RDS, SIMPLEDB & DYNAMODB

- RDS: MySQL or Oracle relational databases
- RDS: Invisible, auto-sharding, multi-AZ deployment, read replicas
- SimpleDB: non-relational (NoSQL) key-value-based data store; basically giant tables
- SimpleDB: automatically replicated and geographically distributed
- DynamoDB: NoSQL database service; columns of key-values like SimpleDB; non-binary data
- DynamoDB: SSDs! Only 64KB limit on item/row however and complex pricing arrangement

SQS

- Hosted, distributed queue
- Ordering, idempotency and approximations
- Consumers control message lifecycle with visibility timeouts
- Unlimited number of messages can be processed (64KB of text max)

SNS

- Pub-Sub web service
- Notification protocols include HTTP(S), SQS, email, SMS
- Subscribers must opt-in (SQS is automatic)
- Great for monitoring and mobile

AWS SDK FOR JAVA

- <https://github.com/amazonwebservices/aws-sdk-for-java>
- Apache 2.0 License; some Apache Commons dependencies
- Almost all products and services supported
- Samples included

AWS TOOLKIT FOR ECLIPSE

- Includes AWS SDK for Java
- AWS Explorer GUI administration tool
- Go through couple of AWS samples
- Demo toolkit

CAMEL-AWS

- AWS-SQS (2.6.0)
- AWS-SNS (2.8.0)
- AWS-S3 (2.8.0)
- AWS-SES (2.8.4)
- AWS-SDB (2.8.4)
- AWS-DDB (2.10.0*)

DEMO: CAMEL-AWS COMPONENTS

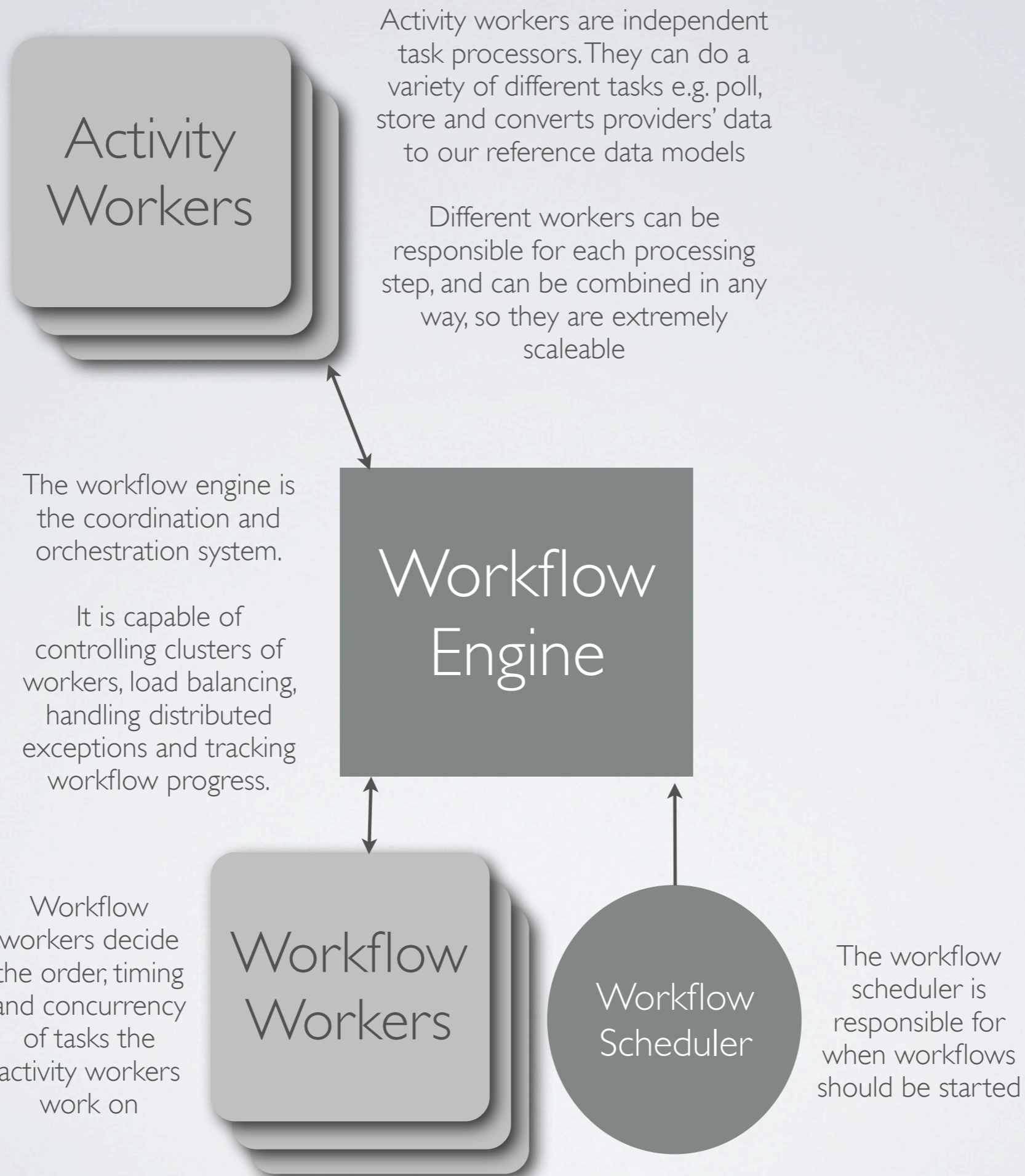
AWS ELASTIC BEANSTALK

- Platform-as-a-Service
- Provisioning, deployment, auto-scaling policies, load balancing, monitoring
- Leverages EC2, S3, SNS, Elastic Load Balancing, and Auto-Scaling
- Linux AMI, Apache Tomcat (6/7, 32- or 64-bit), Apache Web Server, Java EE
- Demo with AWS Toolkit

SCALING DEMO: CAMEL & ELASTIC BEANSTALK

SWF

- Simple Workflow Service
- Distributed coordination hub web service
- Manages state, tasks, execution dependencies, scheduling, monitoring and history
- Console provided
- Tasks, Workers and Deciders
- Camel complements it



AWS FLOW FRAMEWORK

- Java programming framework for SWF
- Translates method calls into schedule decisions; assigns tasks to workers
- Long running workflows, large media processing
- `@Workflow` and `@Activities` (show simple example)

CLOUD ROUTING DEMO: CAMEL & SWF

Q&A