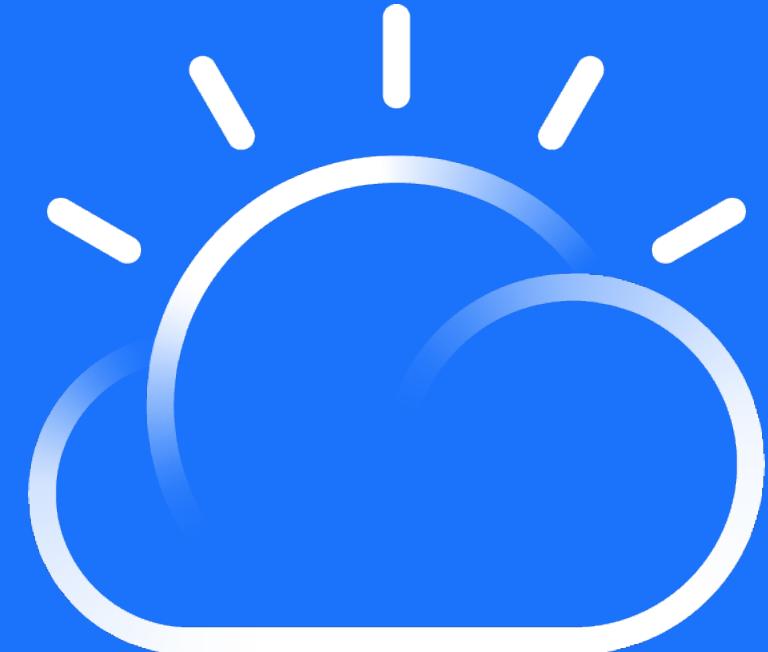


Kafka for the Enterprise: What the docs don't mention



IBM Cloud

IBM

Who are we?

Mark Lucente

- Technical Director & EA leader
- Swears Kubernetes is only for hipsters
- Call sign Grandpa Zilker
- Leads our Blockchain practice and secretly likes Kafka, Hyperledger, Node.js, and containers.

Nathan Olson

- Senior Consultant
- Will tell you Kafka can cure cancer
- Has a hipster beard as required for Kafka
- Probably going from this session directly to an online gaming competition
- Key contributor on multiple high profile Kafka clients



Zilker Technology Overview

Enterprise Solutions for a Digital World

IBM Platinum Partner, Garage Partner, Global Elite Partner

250+ People working on many kinds of projects

- Cloud Integration Platform: APIC, MQ, DP, ACE-IIB, ACE-CI, Event Streams
- Mngt & Platform: ICP, DevOps, Mobile, Docker
- Enterprise Blockchain

Lots of Awards: Beacon, Watson Build NA 2018 and others

Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



Stories from Kafka land – don't do these



Many Kafka platforms. Why not try them all?

Since when is Kafka a database?

Pretend “Kafka” is “Beetlejuice” or “Candyman”. Don’t say it too many times.

High speed logging is awesome. But on VMs?

Most companies are not Netflix. Obvious, right?

Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



ilities for Enterprise Architects



Discovery: Hashicorp
Consul

Extensibility/Platform
Compatibility: Spring Cloud
Streams

Transparency: IBM Event
Streams

Security: IBM Event
Streams Security

Usability: Producers,
Streams, & Consumer API

Durability/Scalability: IBM
Event Streams

Integrability: Kafka Multiple
consumer model

Transparency: Zipkin

Data Retention: Topic
Retention Policy

Data governance and avoiding product name

IBM

- Data governance - “Wait I thought sharing was caring?”
 - When it comes to data people can get rather defensive
 - Offer Kafka as a means to facilitate enterprise level goals
- The problem with product name
 - People focus on the name rather than the idea
 - Advise, sell the idea to the business not the product unless you know that it will help strengthen your case

NO: “Kafka does that” or “The Kafka team does that”.

YES: “We need the data to reliably synchronize to 3 different places”.



Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



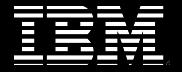
Poor development patterns: Got 99 problems

IBM

- Reinventing the REST Proxy
 - I've met Ewen, he's more than likely smarter than you
- Dual Write Problem
 - Service that write's to Kafka and somewhere else
 - Kafka anti-pattern
- Multiple supported clients
 - Node.js, Golang, Java Producer/Consumer Clients, Spring Kafka, Spring Cloud Streams
 - Unless you have a compelling reason for multiple clients, identify your development stack and design good, reusable patterns.



How we fix it: Good Development Patterns



- Enterprise Integration Patterns
Book → Buy it
- Event-Carried State Transfer
 - Event-driven materialized views
 - Benefits: Data models that fit your client's needs, blazing fast reads
 - Drawbacks: Lots of data, everywhere.
- Event sourcing
 - Audit of changes to entities within your domain
 - Benefits: All changes to entities are tracked within the domain
 - Drawbacks: Data governance and retention policies
- CQRS
 - Separate read and write's
 - Benefits: Autonomously scalable services which adapt to system requirements
 - Drawbacks: Added complexity and difficult

Source: <https://martinfowler.com/articles/201701-event-driven.html>

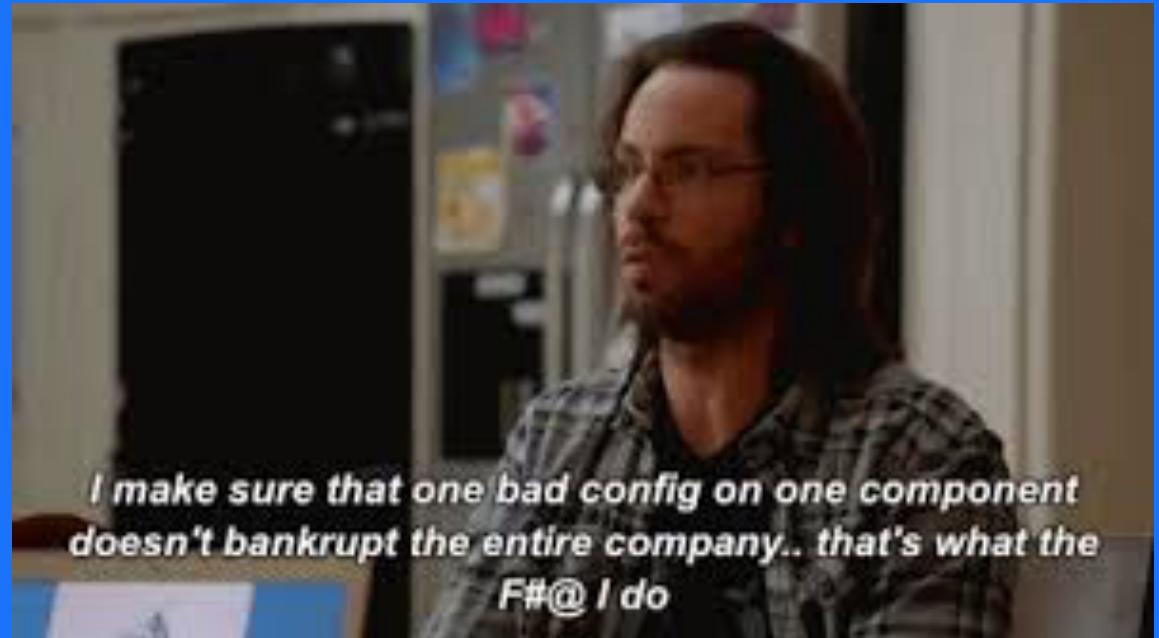
Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



- Select hardware per use case
 - High throughput: Physical
 - Lower throughput: Virtualized
 - Do what your enterprise does best
- Containers with Kafka are pets not cattle.
- Segment zookeeper and broker
 - No need to colocate, spec servers according to needs
- Consensus protocol w/ Kafka
 - 3, 5, 7
 - Eventually Consistent
- `auto.create.topics=false`
- Kafka is upgraded often
- Kafka is rack aware
 - Spread replicas across racks

- Mini-projects:
 - Retention Policies
 - Partitioning strategy
 - Naming conventions
- Use deadletter same as MQ and JMS.
- max.message.bytes - keep the default
 - Optimized for small payloads and high throughput



Staffing Concept - Enterprise Managed Service



Do you like MQ Hotel? Integration COE? Most shared cloud concepts? Any other COE? Then you are going to like Kafka as an enterprise managed service.

Concept:

- Single team that can provide a smooth immersion process for many project teams that will start using Kafka in their designs.
- Team covers: Ops, DevOps, Design Patterns, Project Estimations, L2-L3 Support.
- Team should initially be formed by donated time from: EA, Dev, Ops
- Team could be funded out of EA or Ops budgets in most companies without causing too much of a stir.
- Recommendation: Over time this team could disperse or move to DevOps. But for now until late 2020 try to concentrate expertise in your enterprise.
- Develop Common Frameworks, Do lunch 'n learns, etc.

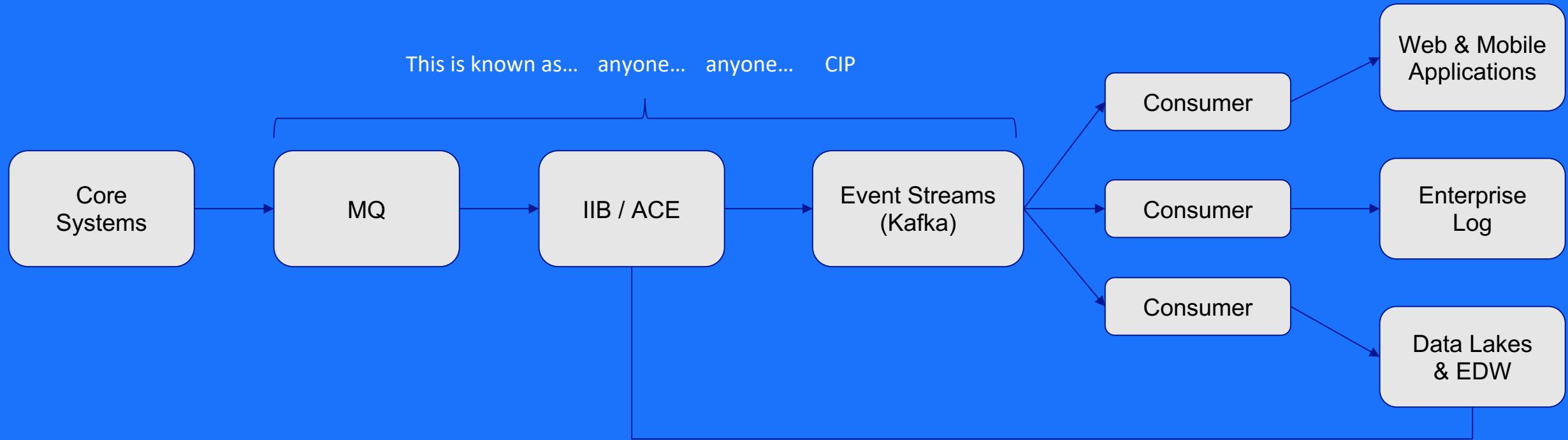


Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



Easy Fit in an IBM Integration Shop



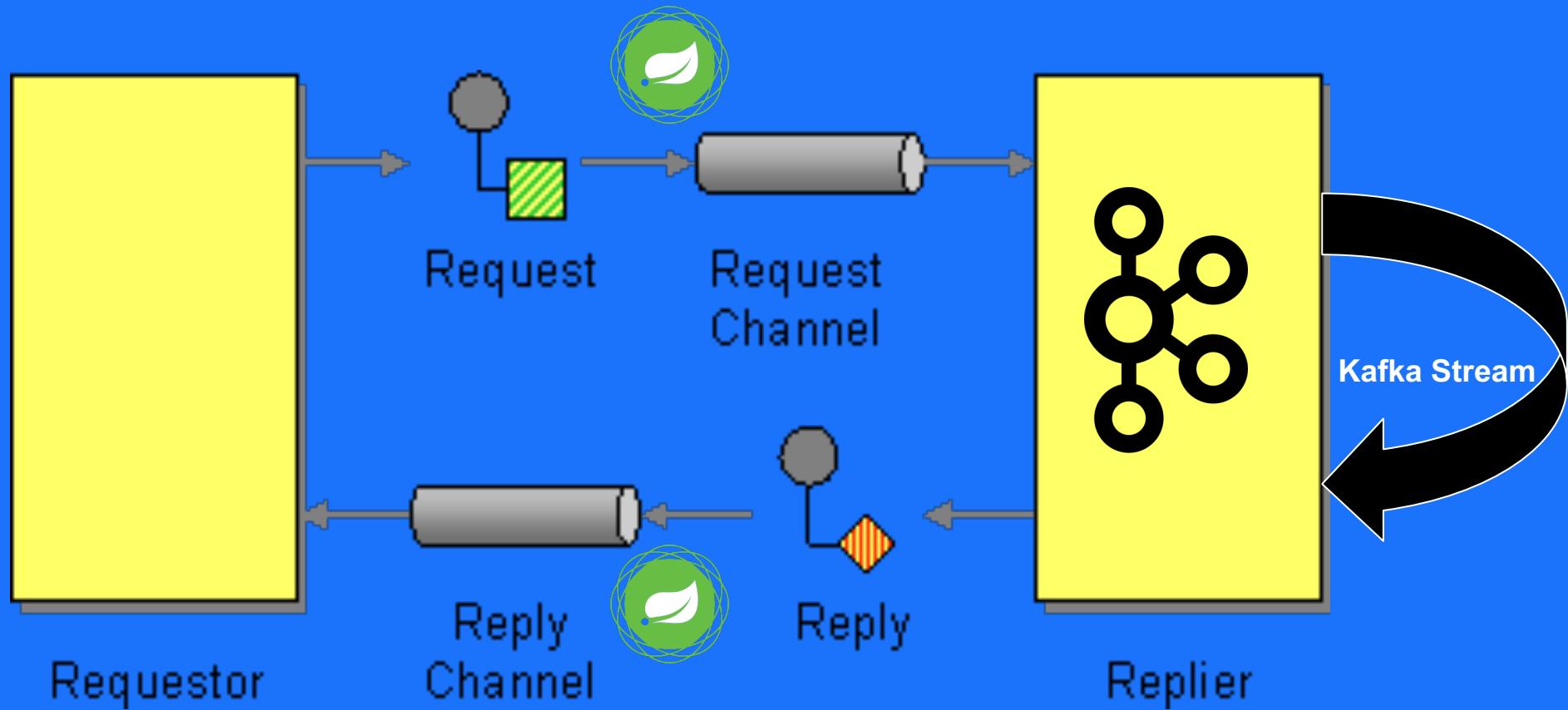
Topics:

- Story Time
- EA Viewpoint
- Developer Viewpoint
- DevOps & Ops Viewpoint
- Among IBM Integration Tools
- Demo Time



Demo Time + Request-Reply thought exercise

IBM



Source: <https://www.enterpriseintegrationpatterns.com/patterns/messaging/RequestReply.html>

Thank You

