

MODERN DATA STREAMING AND PROCESSING WITH APACHE KAFKA AND JNOSQL

BY LEONARDO LIMA (@LEOMRLIMA)



ABOUT THE PRESENTER

- **LEONARDO LIMA**
- **JAVA CHAMPION**
- **FROM CAMPINAS, BRAZIL**
- **DEVELOPING INDUSTRIAL INTERNET OF THINGS SOLUTIONS SINCE 2005**
- **@LEOMRLIMA**

A MODERN ARCHITECTURE IS...

- **RESILIENT**
- **SCALABLE**
- **UPDATEABLE**
- **SECURE**
- **POLYGLOT**

EVENT DRIVEN ARCHITECTURE

- **EVERYTHING IS AN EVENT**
- **100% ASYNCHRONOUS**
- **THERE'S NO DIRECT COMMUNICATION BETWEEN SERVICES**
 - THERE'S NO REST!
- **ALL COMMUNICATION GO THRU AN EVENT BROKER**

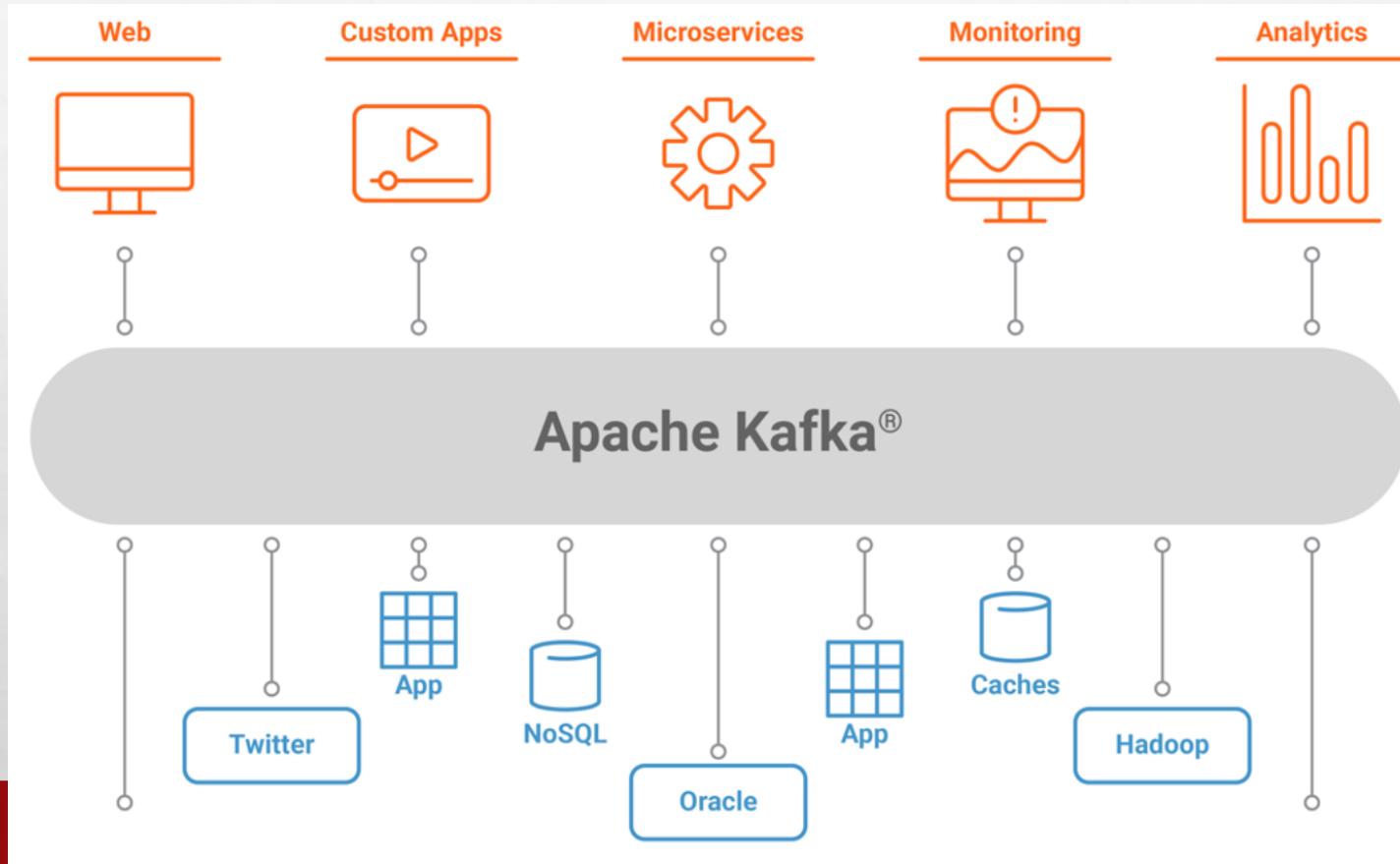
EVENT DRIVEN ARCHITECTURE

- **DEVICE REGISTERED**
 - JSON BODY WITH DETAILS, GLOBAL UNIQUE ID IS ATTRIBUTED (DEVICE ID)
- **DEVICE CONNECTED**
- **DEVICE SENT DATA**
- **SEND DATA TO DEVICE**
- **DEVICE DISCONNECTED**

APACHE KAFKA

- STARTED AS A VERY DISTRIBUTED TRANSACTION LOG INSIDE LINKEDIN
- NOW IS A LOT OF OTHER THINGS...
 - DISTRIBUTED DATABASE
 - MESSAGE SYSTEM
 - STORAGE SYSTEM
 - STREAM PROCESSING
- SO... AN EVENT BROKER!
- PRODUCER(S) SEND DATA TO A TOPIC, CONSUMER(S) CONSUME FROM TOPICS

APACHE KAFKA



NOSQL DATABASES

- NOT STRUCTURED DATABASES
- GOOD WHEN YOU NEED FLEXIBILITY AND CHOICE
- YOU CAN SACRIFICE ONE ASPECT IN FAVOR OF ANOTHER - BREWER'S CAP THEOREM
- NOT ACID – BUT BASE

DEMO – DEVICE STATUS BOARD

AN EXAMPLE OF MICROSERVICE AND EVENT DRIVEN ARCHITECTURE, WITH APACHE KAFKA AS EVENT BROKER AND REDIS NOSQL DATABASE AS BACKEND

IMPROVING THE IMPLEMENTATION

- CURRENTLY, APPLICATION AND INFRA ARE MIXED, SO THERE'S A LOT OF LOCK-IN
 - APPLICATION DEPEND ON KAFKA, REDIS, GSON...
- THAT CAN BE OK, BUT THERE ARE BETTER WAYS TO ISOLATE APPLICATION CODE FROM INFRASTRUCTURE CODE
- ALSO, USE OF STANDARDS ALLOWS CHOICE
 - WHERE WE CAN'T USE A STANDARD, WE'LL ISOLATE THRU ONE!

ECLIPSE JNOSQL

- DEVELOPING STANDARD FOR CONNECTING (JAKARTA EE) APPLICATIONS TO NOSQL DATABASES
- THINK JPA AND JDBC, BUT FOR NOSQL
- JOIN THE FUN! IT'S AN OPEN STANDARD IN DEVELOPMENT

CDI EVENTS AND EXTENSIONS

- THERE'S ONE STANDARD FOR MESSAGES: JMS
- BUT... IT CAN'T REALLY EXPOSE KAFKA'S SPECIFIC SEMANTICS TO THE APPLICATION
- WE'LL ABSTRACT THAT WITH CDI EVENTS
 - SO EVENTS CAN COME FROM POTENTIALLY ANYWHERE, AS WELL AS COME FROM ANYWHERE TOO

DEMO – IMPROVED STATUS BOARD

AN EXAMPLE OF MICROSERVICE AND EVENT DRIVEN ARCHITECTURE, USING CDI EXTENSIONS WITH APACHE KAFKA AND JNOSQL CONNECTING TO REDIS NOSQL DATABASE

QUESTIONS?

THANKS!

YOU CAN REACH OUT WITH QUESTIONS, SUGGESTIONS AND MORE AT TWITTER/@LEOMRLIMA

CODE IS AT GITHUB.COM/LEOMRLIMA/KAFKA-NOSQL