



Connecting the Edge

Project Flogo as Ultra-lightweight IoT Apps framework

Anshul Sharma

03 Feb 2018 – FOSDEM'18

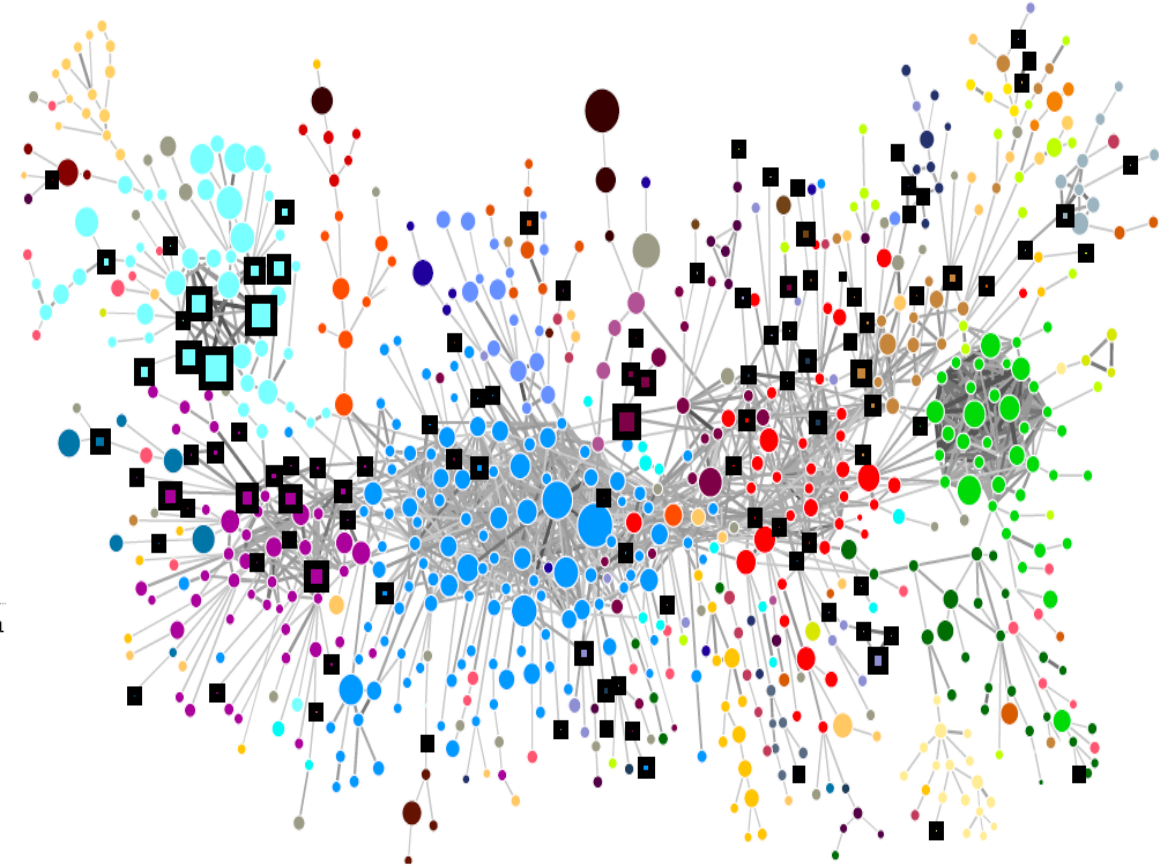


**CONNECTED
INTELLIGENCE**

CONFIDENTIALITY

The following information is confidential information of TIBCO Software Inc. Use, duplication, transmission, or republication for any purpose without the prior written consent of TIBCO is expressly prohibited.

What (hyper)connectivity looks like ...

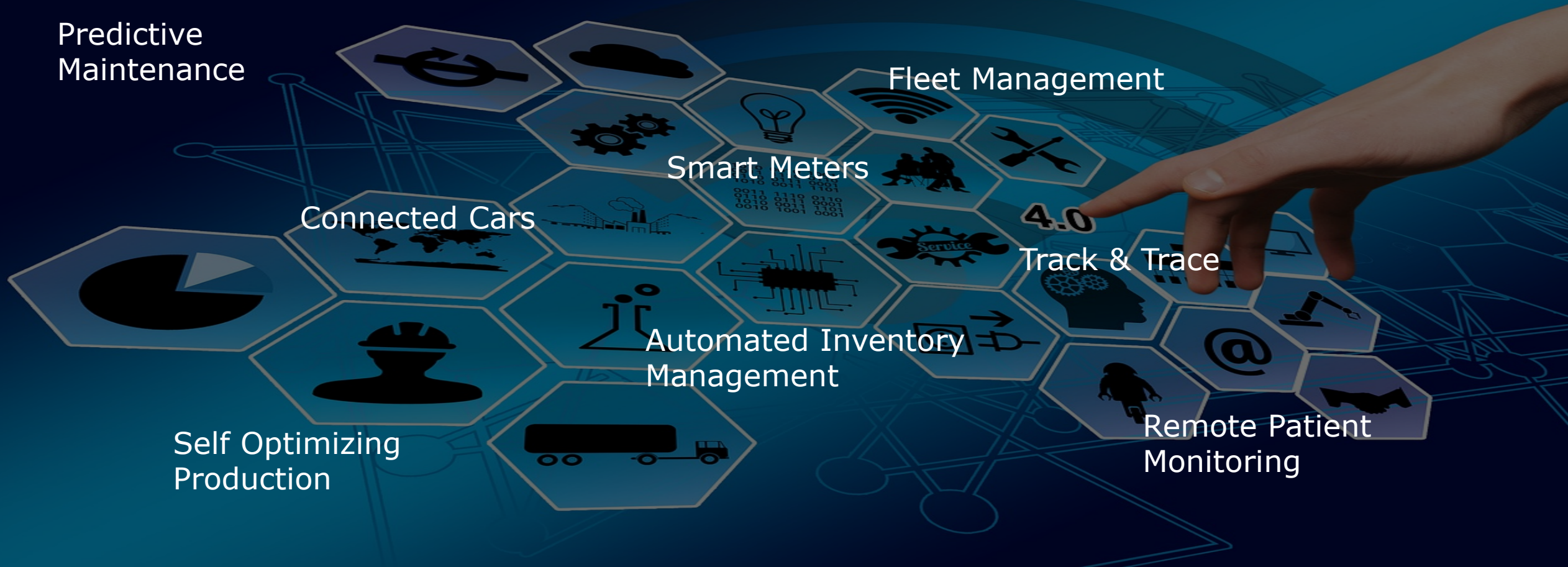


In 2015, we had 4.9 billion connected things, by 2020, the number of Internet-connected things will reach or even exceed **50 billion**.

GE believes that the “Industrial Internet” (their term for IoT) will add **\$10 to \$15 trillion** to global GDP in the next 20 years.

According to estimations by the McKinsey Global Institute, the IoT will have a total economic impact of up to **\$11 trillion** by 2025.

Consumer & Industrial Use Cases



The architecture of today



- Cloud-centric, Centralized
- Data transported from Devices to Cloud over network
- Processing happens in the Cloud

Embracing edge computing



- Massive amount of data needs to be processed in real-time
- Cloud-centric IoT is not reliable
- Push computing, analytics & decision making to **EDGE!!**

What is Project Flogo?

Project Flogo™ Ultralight Edge Microservices Framework



flogo.io



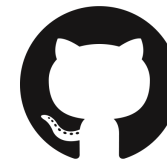
10-50x lighter than
Java, .NET or Node.js



Visually model & test
flows as functions



Event-driven design
with 1st-class support
for AWS Lambda

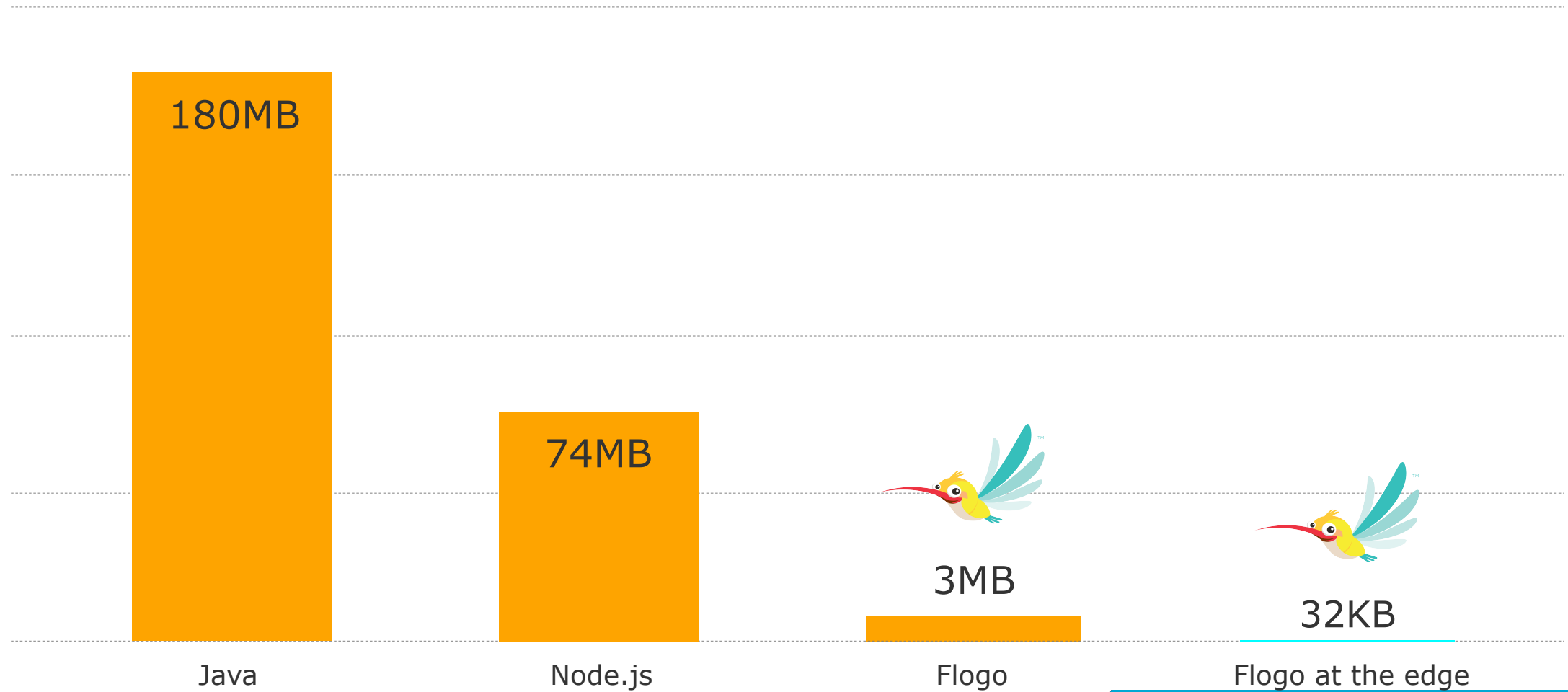


100% Open Source
with zero lock-in



Build smarter with Edge
Machine Learning

How small?

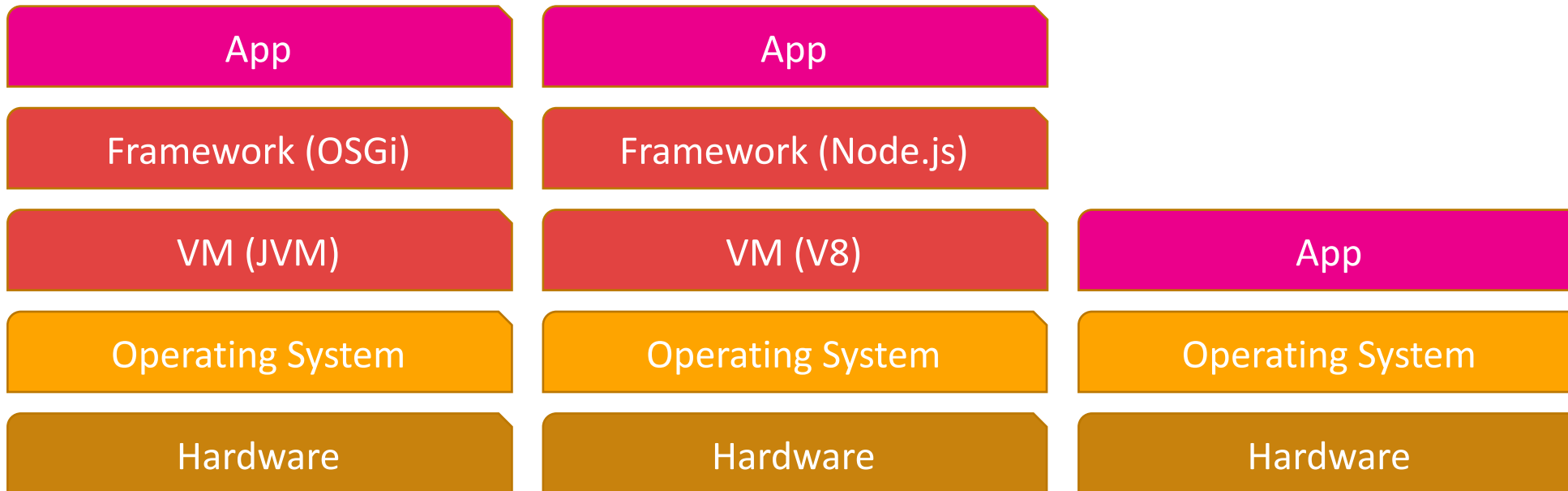


How did we get so small?

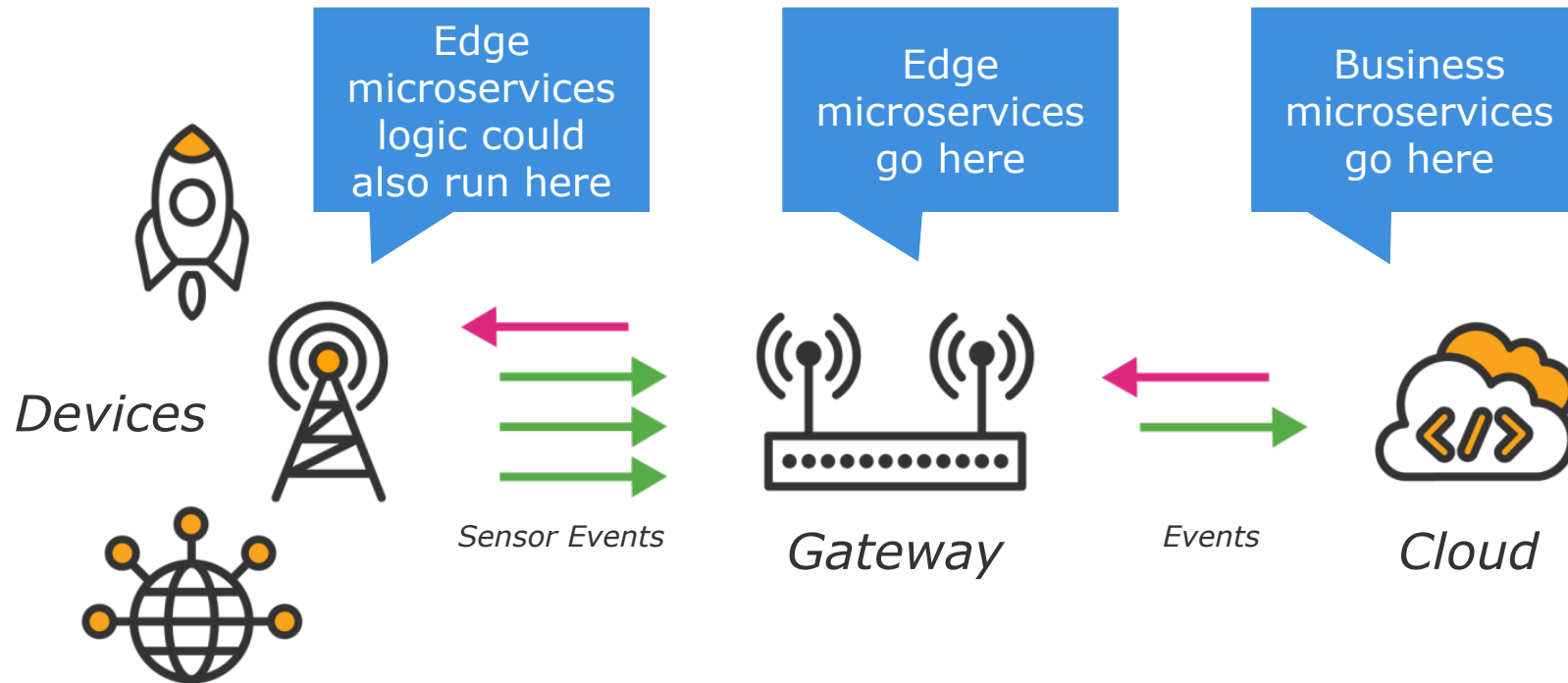
Java, NodeJS are great, but too large for resource constrained environments

Why **Golang** for Project Flogo-

- Compiles natively and runs natively.
- Only the required dependencies are built into the application.
- Static linking enables zero OS dependencies.



A better IoT integration blueprint



Benefits of edge integration & event processing

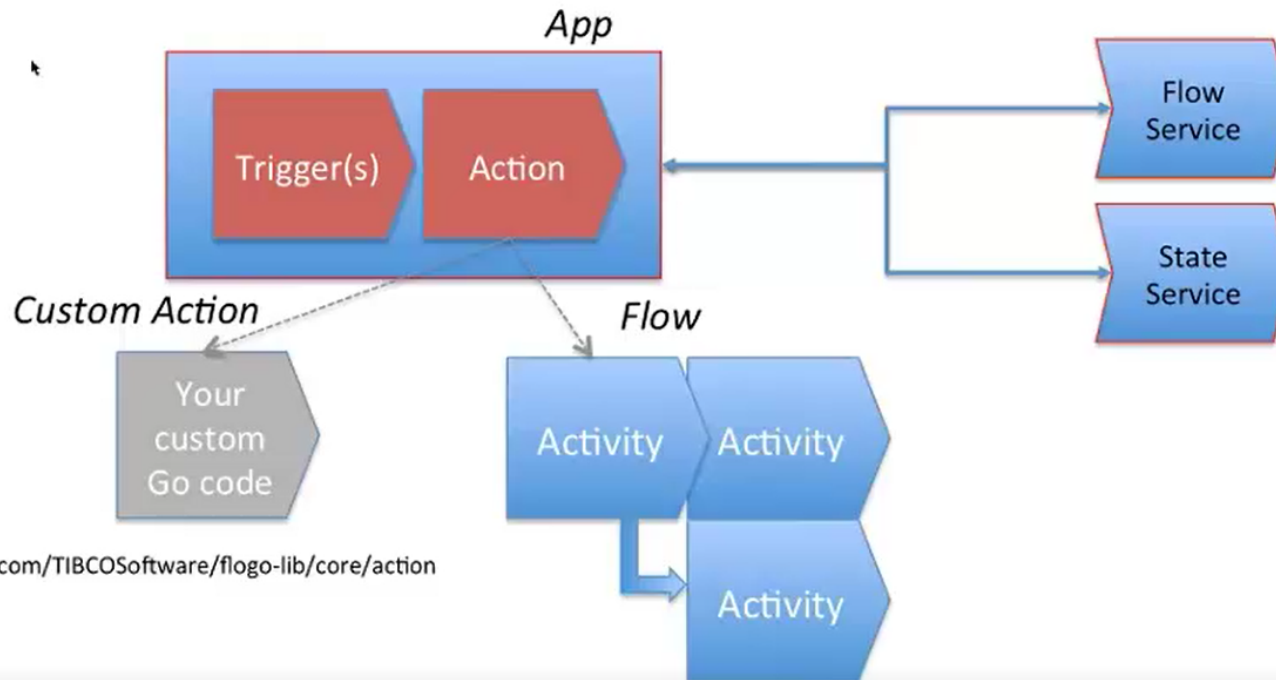
- Local control = more reliable
- Less bandwidth & lower TCO
- Fewer hops, less latency

Event-driven by design

App = Trigger + Actions

Action = Flow of Activities

Flow = Activities + Transitions + Error Handling



github.com/TIBCOSoftware/flogo-lib/core/action

Event Processing

- Triggers emit events
- Activities collect events, process them and emit output events for further processing

What is Project Flogo?



Flogo at the Edge

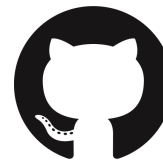
flogo.io



Microcontrollers



Operating Systems

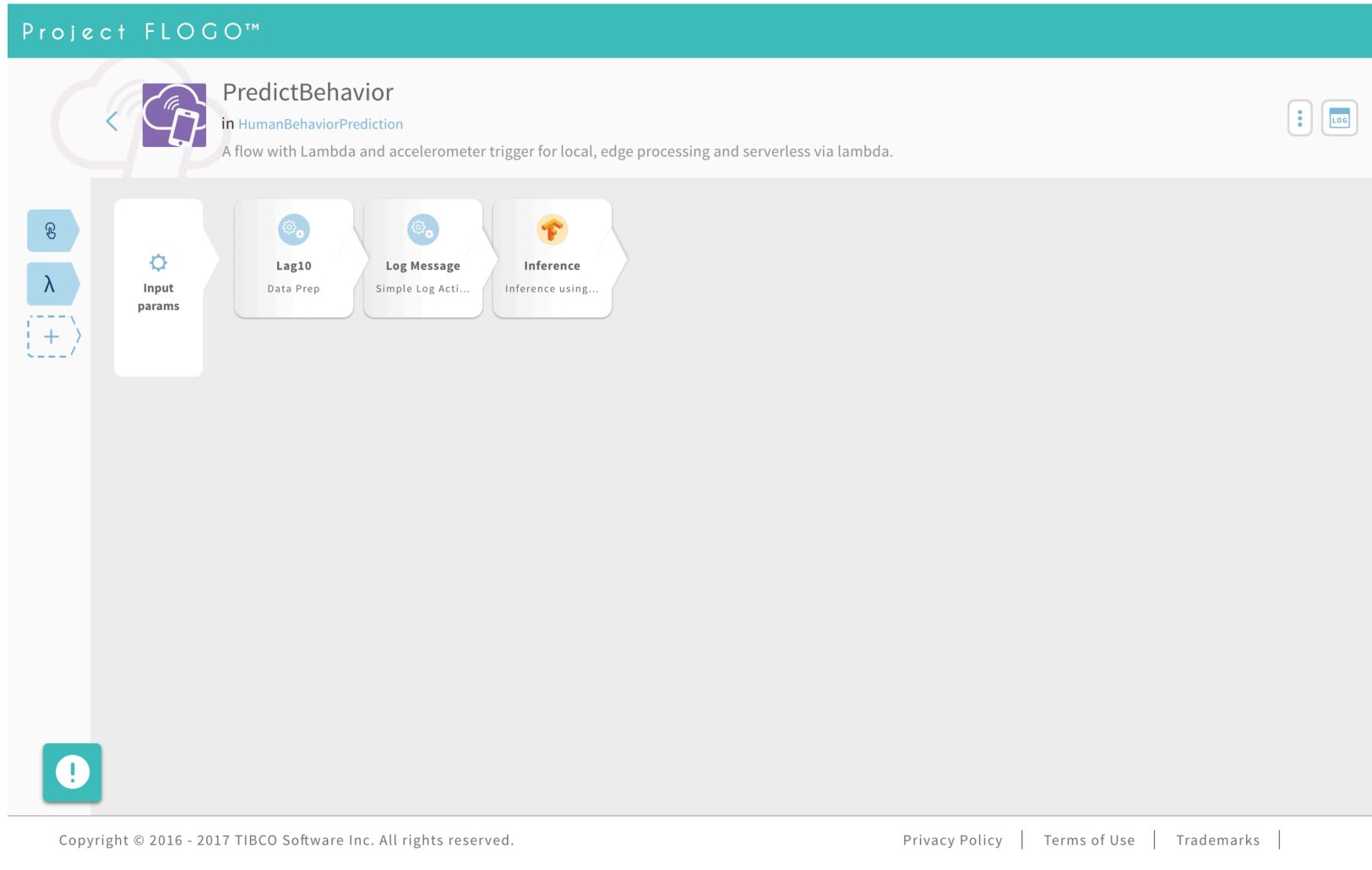


100% Open Source
with zero lock-in



TIBCO Statistica Models

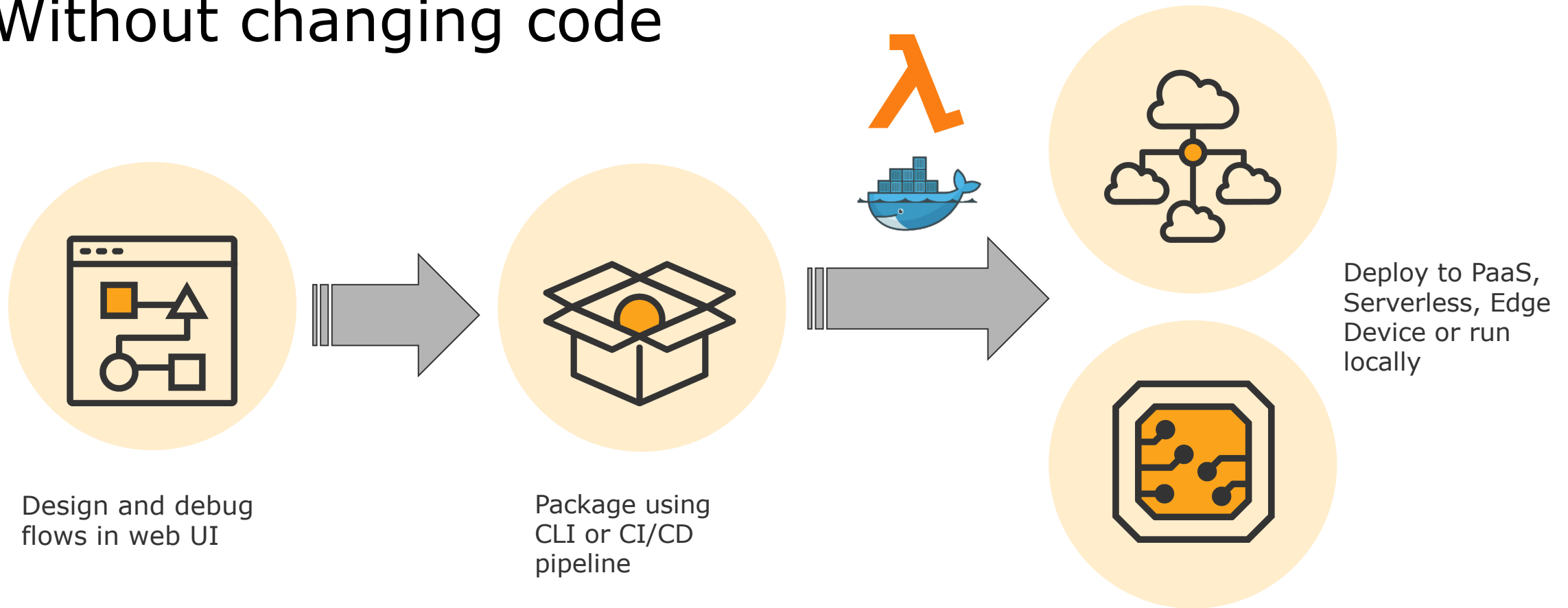
Flogo Web UI



- Low friction web-native UX
 - Express app logic using rich flows, not just data or request pipelines
 - Inline data transformations
 - Built-in web-based debugger
 - Build for target platform directly from UI
 - Available on Docker Hub or Flogo.io

Deploy Flogo Apps where you need them...

Without changing code



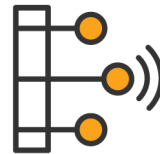
Introducing Edge ML



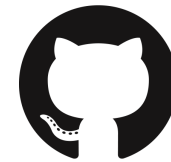
Edge Machine Learning



Execute TensorFlow
Models Natively in Flogo
Flows



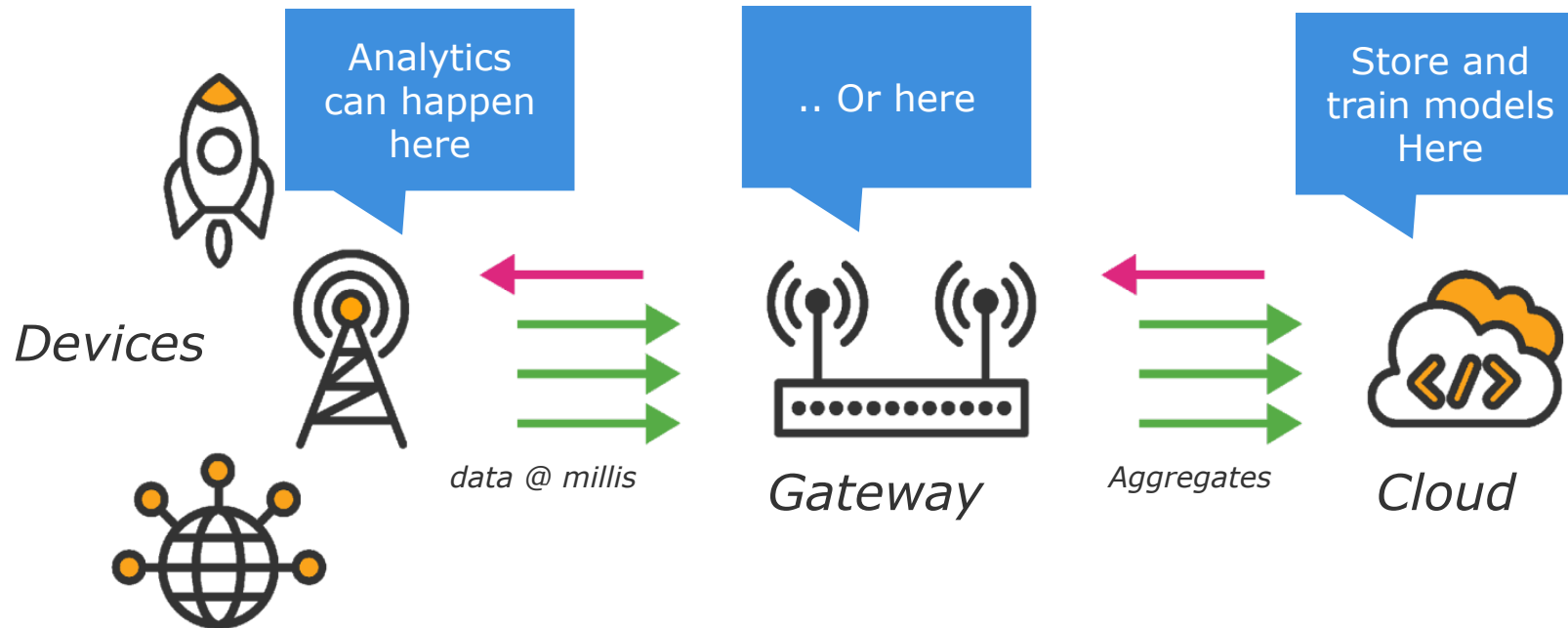
Streaming data
constructs



100% Open Source
with zero lock-in

Why ML @ the Edge

- Data Volume, Generation
 - Data collection exceeds ability to transport
- Intelligent Aggregation
 - Reduces transfer & storage costs
 - Smarter, more efficient networks
- Predictions
 - Smarter Device Actions, Less Network Latency
 - Actions resilient to network connectivity issues



The Issues:

- Prediction Lag
- Massive Data Transfers
- Connectivity Requirements

..ML Challenges Amplify The Issues of IoT Integration!

Project Flogo Value

Easily build and deploy microservices to any cloud-based PaaS, FaaS or the smallest of IoT edge devices



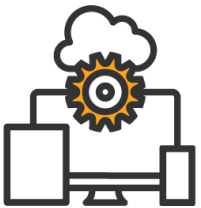
Build Microservices Faster

- Create microservice and function flows with intuitive, browser-based visual tooling
- Easy to install, without a lot of dependencies
- Available as an open source project to get started quickly and without any hurdles



Leverage FaaS and Serverless Platforms

- Lower your TCO by employing serverless and FaaS architecture
- Migrate existing or new services to pure cloud FaaS platforms like AWS Lambda



Deploy Anywhere Including Edge Devices

- Deploy microservice flows anywhere: on-premise, cloud, hybrid and edge devices
- Run on microcontrollers and sensors, with 50x smaller footprint than Node.js/Java



What Next?

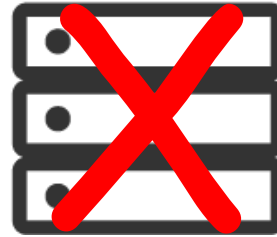
Flogo Roadmap



Model Lifecycle Mgmt

Control the lifecycle of models on running devices, initiate champion-challenger bake offs, push updated models.

Simplified and async cloud delivery of target metrics (to improve model training)



Serverless

Additional Serverless platform support.

Additional tooling to accommodate Serverless deployments



Dev Experience

Enhanced tooling and scaffolding to enable a larger spectrum of flow and activity/trigger developers

Improved Documentation



What can you do next?

- Stay back and grab a beer? 😊



- Docker Pull? Go to flogo.io
- Check out the code – github.com/tibcosoftware/flogo





@anshuldsharma



@anshuldsharma



ansharma@tibco.com

Thank You!