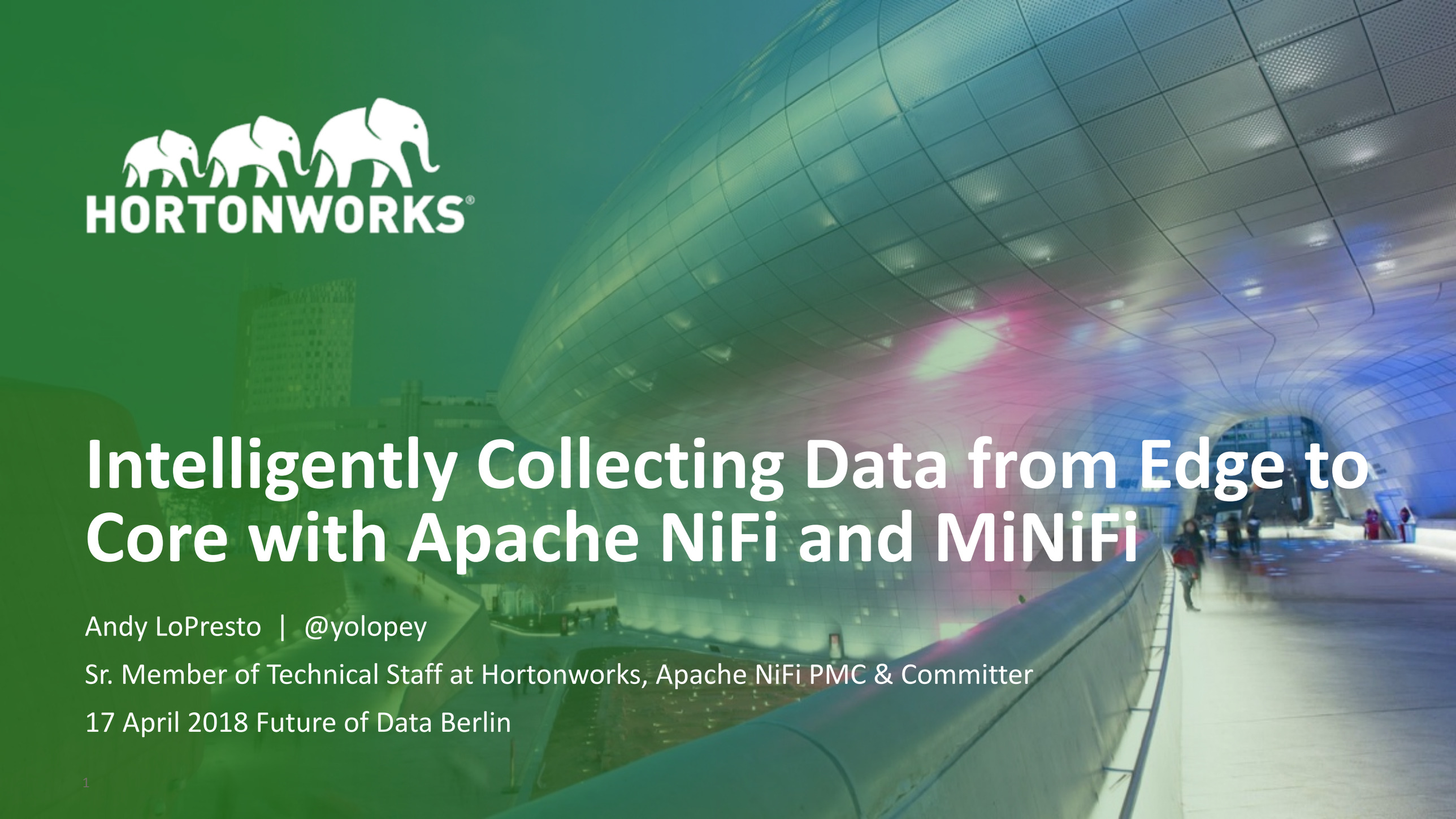


Intelligently Collecting Data from Edge to Core with Apache NiFi and MiNiFi

Andy LoPresto | @yolopey

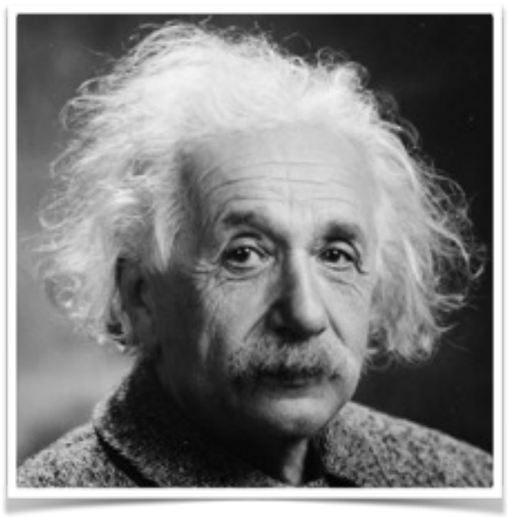
Sr. Member of Technical Staff at Hortonworks, Apache NiFi PMC & Committer

17 April 2018 Future of Data Berlin

Gauging Audience Familiarity With NiFi

"What's a NeeFee?"

No experience with dataflow
No experience with NiFi



"I can pick this up pretty quickly"

Some experience with dataflow
Some experience with NiFi

"I refactored the Ambari integration endpoint to allow for mutual authentication TLS during my coffee break"

Forgotten more about NiFi than most of us will ever know

HORTONWORKS

Agenda

- Introduction
- What is dataflow?
- What are NiFi & MiNiFi?
- What's next?

- All slides provided online, so no need to transcribe

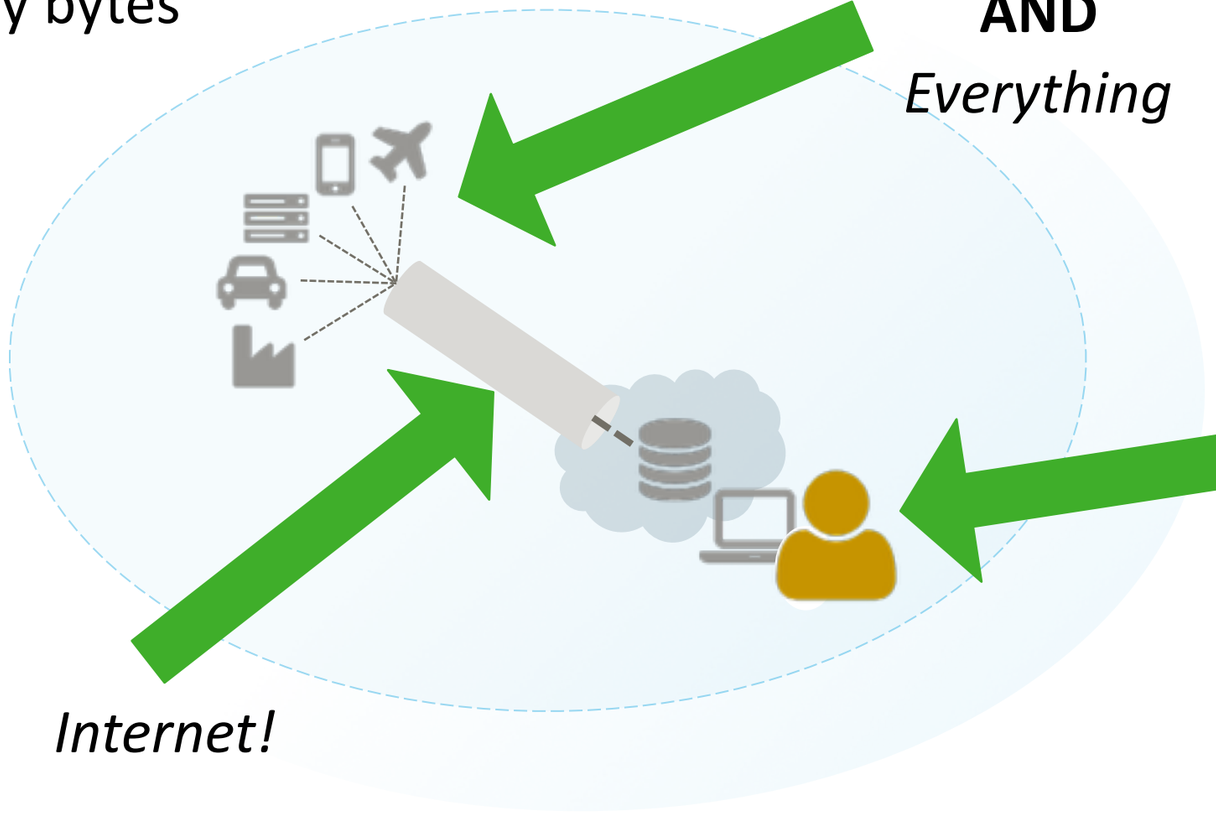
HORTONWORKS

What is dataflow?

HORTONWORKS

What is dataflow?

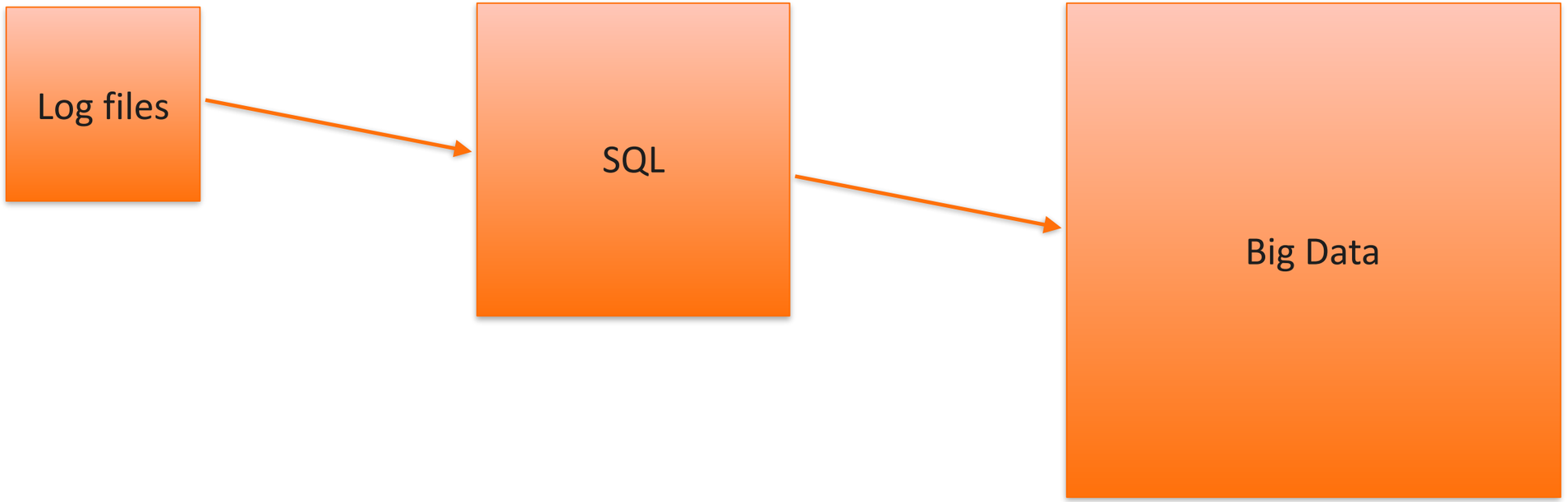
- Moving some content from A to B
- Content could be any bytes
  - Logs
  - HTTP
  - XML
  - CSV
  - Images
  - Video
  - Telemetry



HORTONWORKS

Connecting Data Points Is Easy

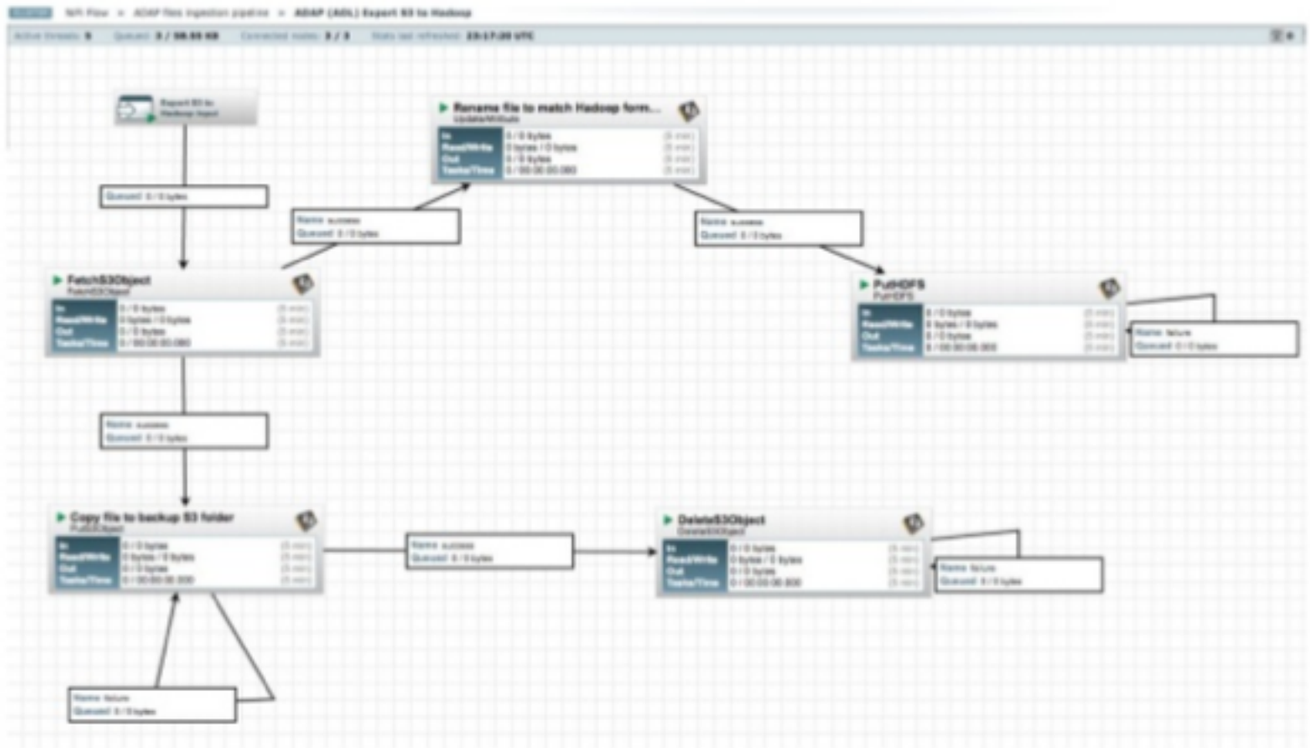
- Simple enough to write a process
  - Bash/Ruby/Python
  - SQL proc
  - etc.



HORTONWORKS

Big Data Is About Scale…

- …and this doesn't scale
- Example use case:
  - AOL Data Processing
  - AWS -> HDFS
  - 20 TB ingested/day
  - Lev Brailovskiy, "Data Ingestion and Distribution with Apache NiFi", Slide 27, 02/2017
  - https://www.slideshare.net/LevBrailovskiy/data-ingestion-and-distribution-with-apache-nifi



HORTONWORKS

Dataflow Challenges In 3 Categories

Data

- Standards
- **Formats**
- Protocols
- Veracity
- Validity
- Schemas
- Partitioning/ Bundling

Infrastructure

- “Exactly Once” Delivery
- Ensuring Security
- **Overcoming** Security
- Credential Management
- Network

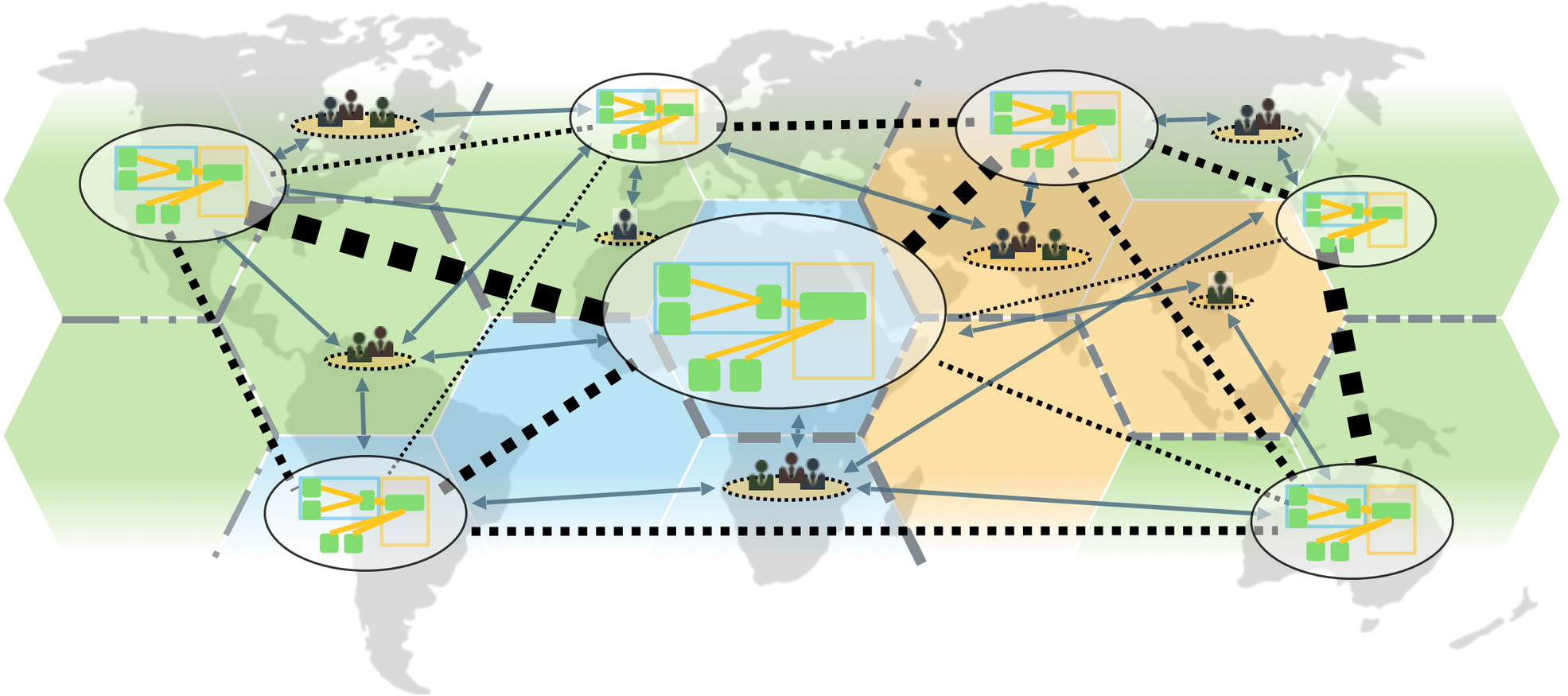
People

- Compliance
- “**That** [person|team|group]”
- **Consumers Change**
- **Requirements Change**
- “Exactly Once” Delivery

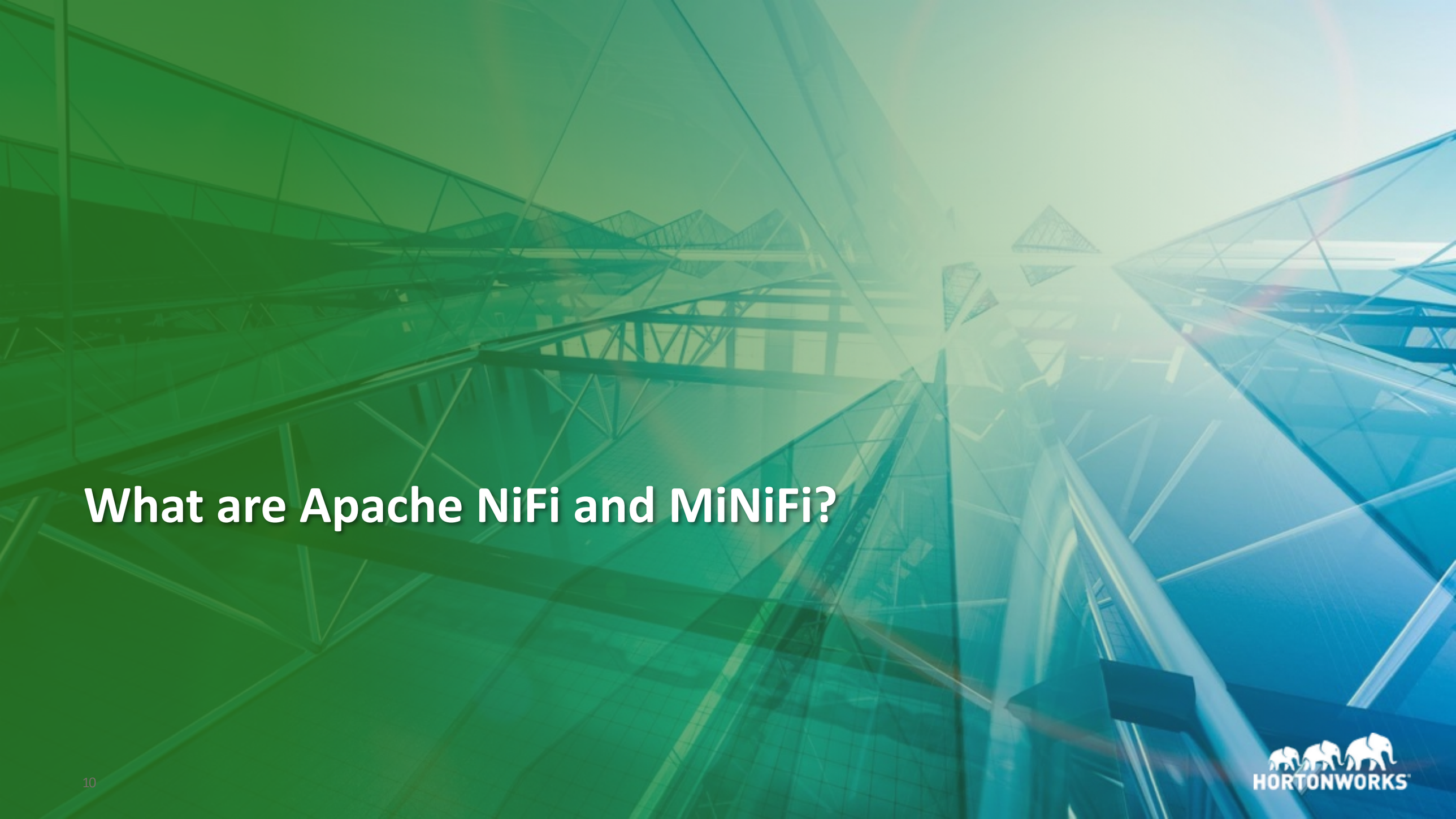
HORTONWORKS

Let’s Connect Lots of As to Bs to As to Cs to Bs to Δs to Cs to φs

Raise your hand if you want to maintain Python scripts for the rest of your life



HORTONWORKS



What are Apache NiFi and MiNiFi?

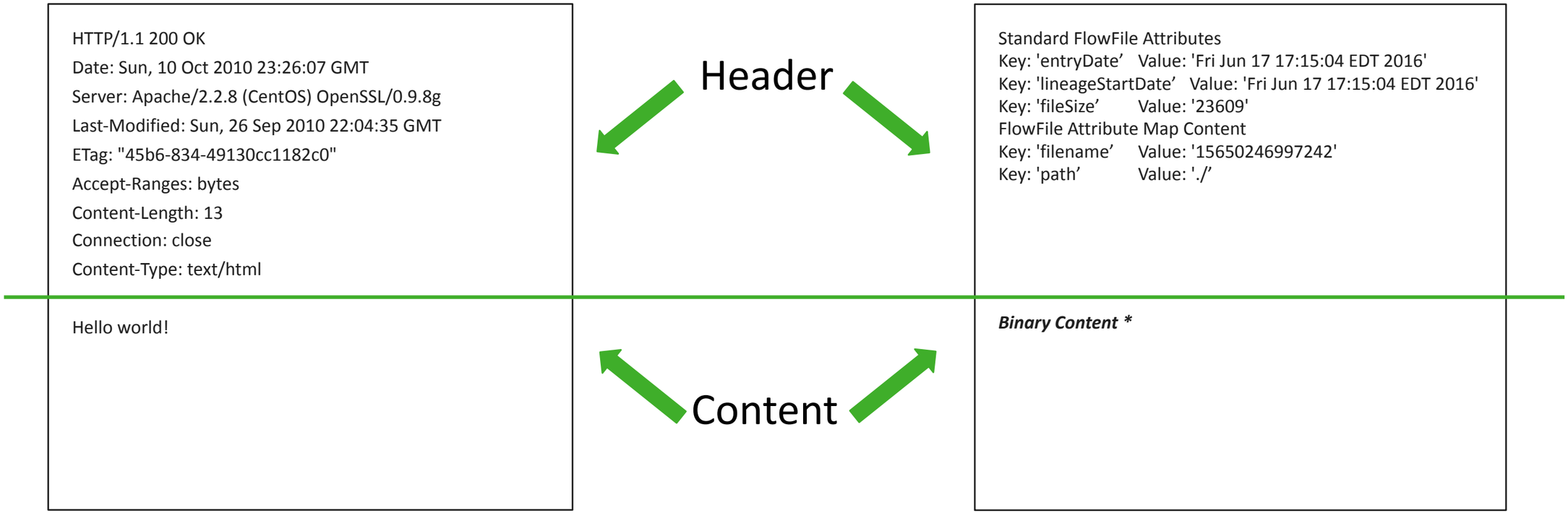
Apache NiFi

Key Features

- Guaranteed delivery
  - Data buffering
  - Backpressure
- Pressure release
- Prioritized queuing
- Flow specific QoS
  - Latency vs. throughput
  - Loss tolerance
- Data provenance
- Supports push and pull models
- Recovery/recording
  a rolling log of fine-grained history
- Visual command and control
- Flow templates
- Pluggable, multi-tenant security
- Designed for extension
- Clustering

HORTONWORKS

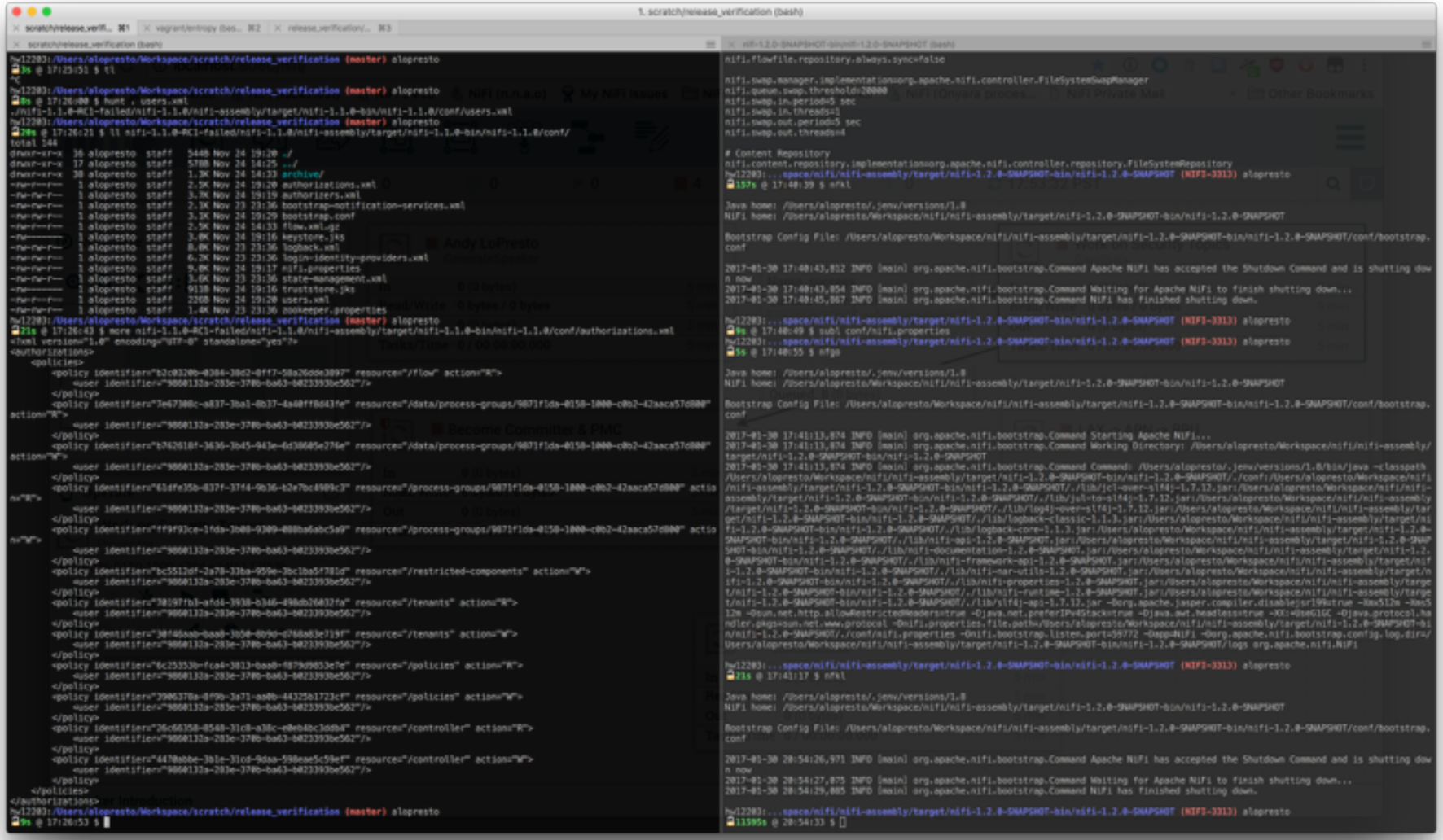
Flowfiles Are Like HTTP Data



HORTONWORKS

User Interface

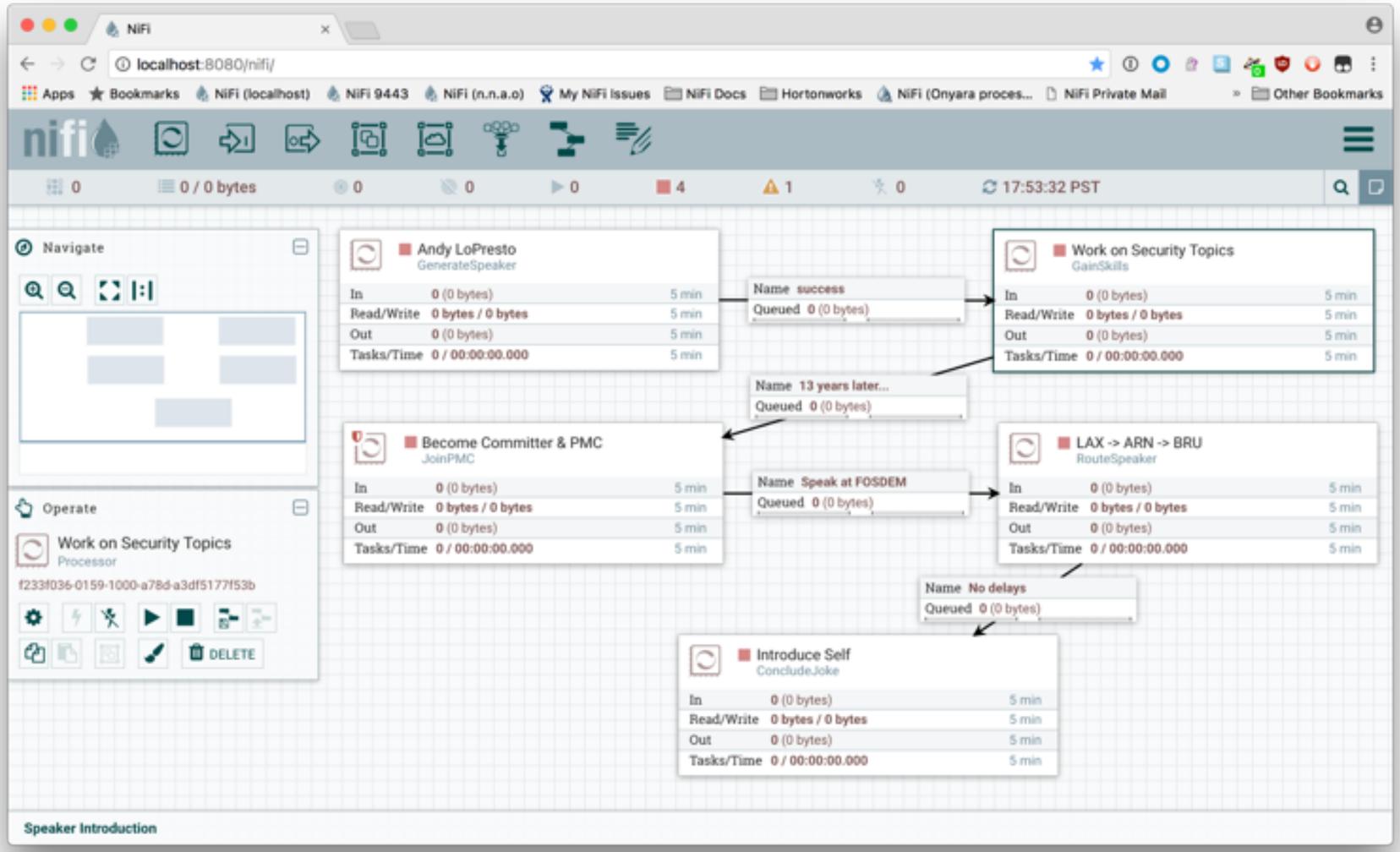
Less of this…



HORTONWORKS

User Interface

Less of this… **… more of this**



HORTONWORKS

Deeper Ecosystem Integration: 260+ Processors, 48 Controller Services

- FTP
- SFTP
- HL7
- UDP
- XML
- ⋮
- HTTP
- WebSocket
- Email
- HTML
- Image
- Syslog
- AMQP

Hash	Encrypt	GeoEnrich
Merge	Tail	Scan
Extract	Evaluate	Replace
Duplicate	Execute	Translate
Split	Fetch	Convert
⋮	⋮	

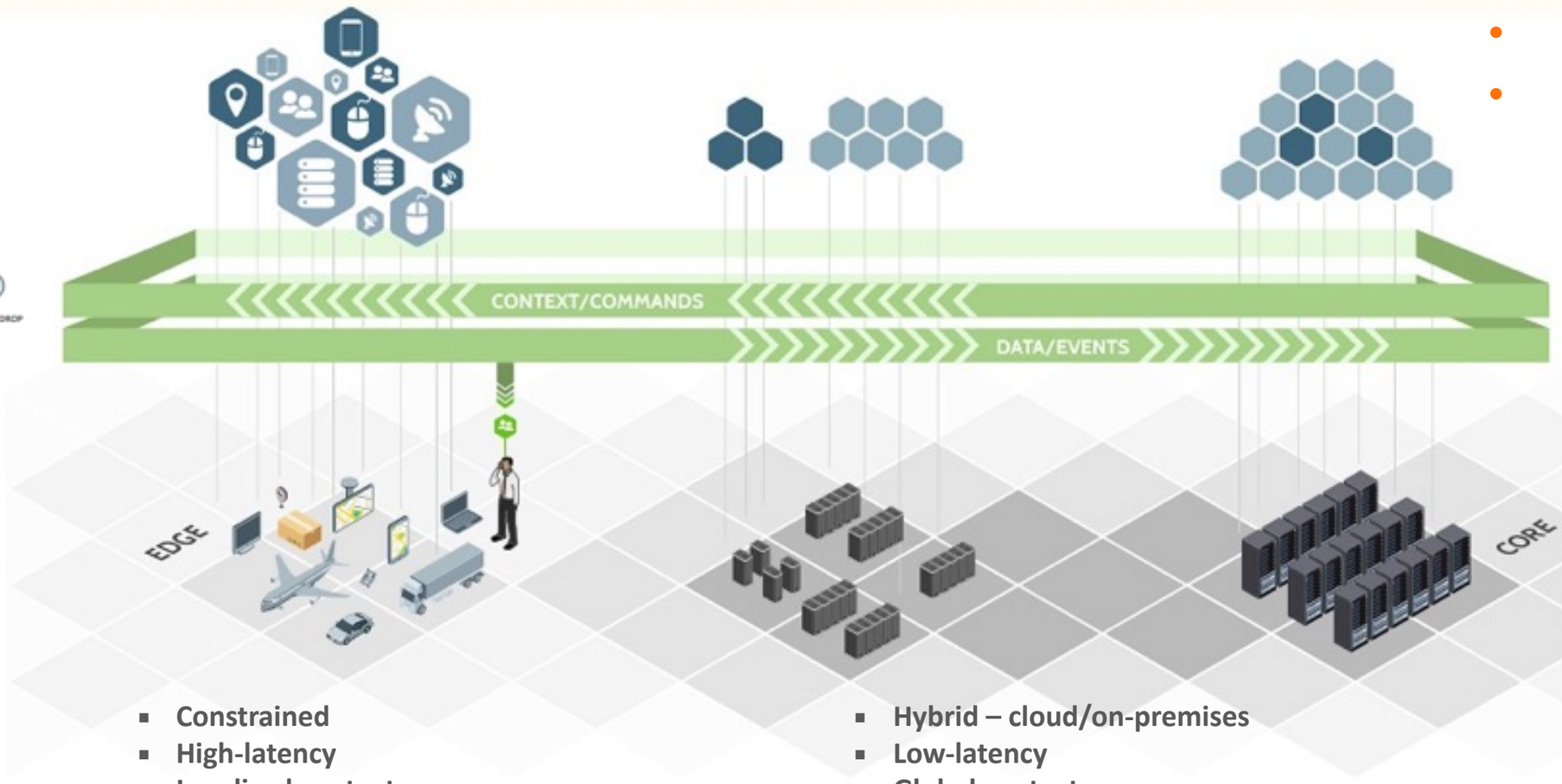
Parse Records	Convert Records
Route Text	Distribute Load
Route Content	Generate Table Fetch
Route Context	Jolt Transform JSON
Control Rate	Prioritized Delivery

HORTONWORKS

Data Provenance

Origin – attribution
Replay – recovery

Evolution of topologies
Long retention



Types of Lineage

- Event
- Configuration

- Constrained
- High-latency
- Localized context

- Hybrid – cloud/on-premises
- Low-latency
- Global context

HORTONWORKS

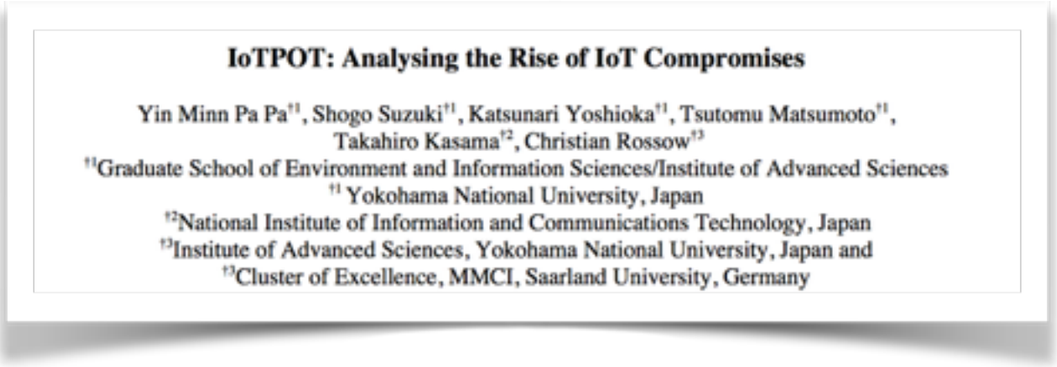
IoT Challenges

- Limited computing capability
- Limited power/network
- Restricted software library/platform availability
- No UI
- Physically inaccessible
- Not frequently updated
- Competing standards/protocols
- Scalability
- Privacy & Security

HORTONWORKS

Recent Examples

- When the Mirai attack has its own Wikipedia page, that's not good
- Hackers stole high-roller database from casino via aquarium thermometer connected to internet (04/2018)



IoTPOT: Analysing the Rise of IoT Compromises

Yin Minn Pa Pa[†1], Shogo Suzuki[†1], Katsunari Yoshioka[†1], Tsutomu Matsumoto[†1], Takahiro Kasama[†2], Christian Rossow[†3]
[†1]Graduate School of Environment and Information Sciences/Institute of Advanced Sciences
[†1] Yokohama National University, Japan
[†2]National Institute of Information and Communications Technology, Japan
[†3]Institute of Advanced Sciences, Yokohama National University, Japan and
[†3]Cluster of Excellence, MMCI, Saarland University, Germany

HORTONWORKS

NiFi Solves Everything*

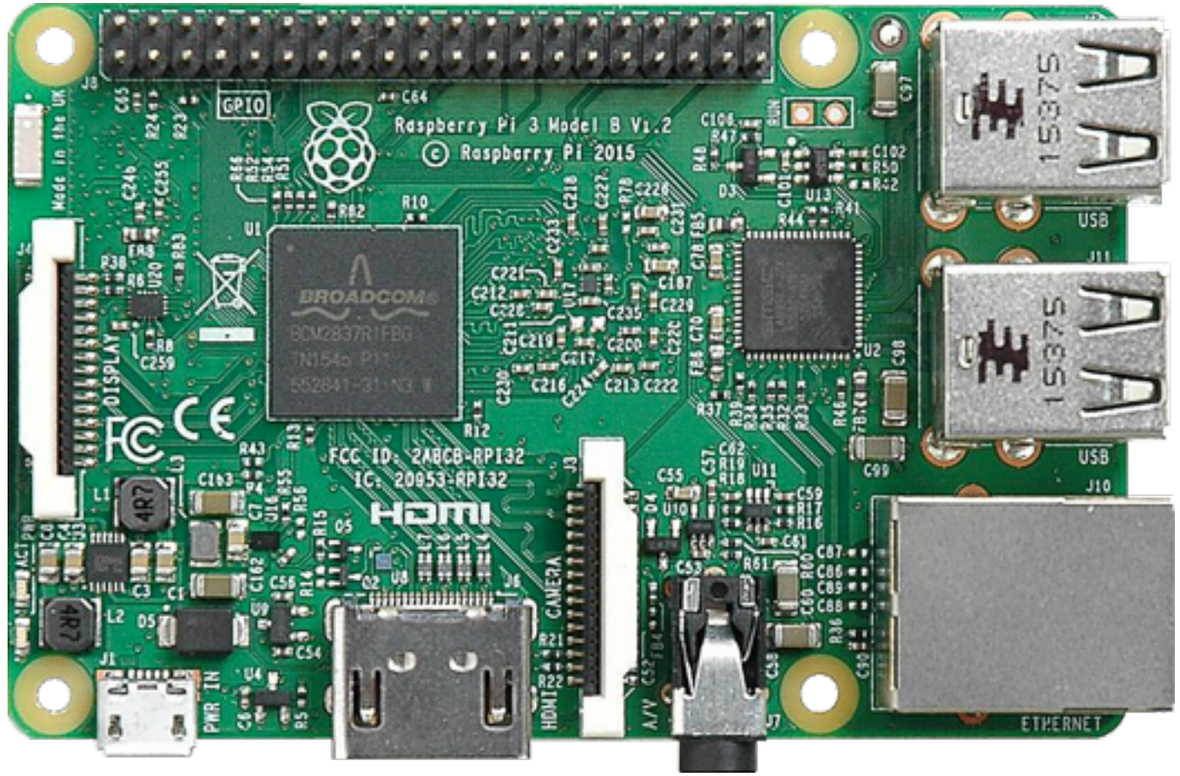
- Runs on JVM
- Provides UI for flow design & monitoring
- Security built-in
  - TLS, authentication/authorization, encrypted data
- Handles practically any format/protocol

HORTONWORKS

NiFi for IoT

- NiFi supports AMQP, MQTT, UDP, TCP, HTTP(S), CEF, JMS, (S)FTP, *AWSIoT*
- With a little pruning, NiFi can run on a Raspberry Pi

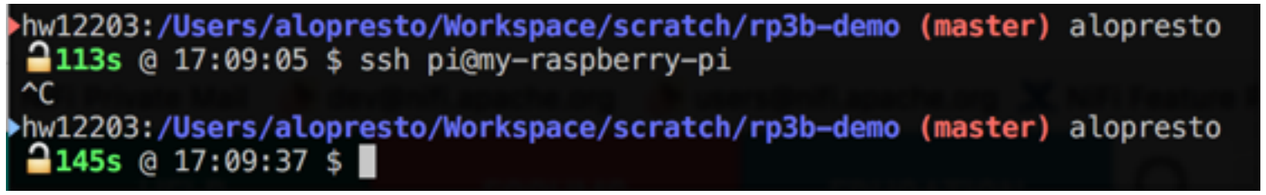
```
.
├── bootstrap
├── jcl-over-slf4j-1.7.12.jar
├── jul-to-slf4j-1.7.12.jar
├── log4j-over-slf4j-1.7.12.jar
├── logback-classic-1.1.3.jar
├── logback-core-1.1.3.jar
├── nifi-api-0.6.1.jar
├── nifi-documentation-0.6.1.jar
├── nifi-framework-nar-0.6.1.nar
├── nifi-html-nar-0.6.1.nar
├── nifi-http-context-map-nar-0.6.1.nar
├── nifi-jetty-bundle-0.6.1.nar
├── nifi-kerberos-iaa-providers-nar-0.6.1.nar
├── nifi-ldap-iaa-providers-nar-0.6.1.nar
├── nifi-nar-utils-0.6.1.jar
├── nifi-properties-0.6.1.jar
├── nifi-provenance-repository-nar-0.6.1.nar
├── nifi-runtime-0.6.1.jar
├── nifi-scripting-nar-0.6.1.nar
├── nifi-ssl-context-service-nar-0.6.1.nar
├── nifi-standard-nar-0.6.1.nar
├── nifi-standard-services-api-nar-0.6.1.nar
├── nifi-update-attribute-nar-0.6.1.nar
└── slf4j-api-1.7.12.jar
```



HORTONWORKS

So Why Do We Need A Different Solution?

- NiFi is designed to “own the box”
- NiFi 0.7.x started up in about 10-15 minutes on RP3 (593 MB)
- NiFi 1.x started up in about 30 minutes on RP3 (760 MB)
  - 33 new processors
  - Rewrite for multi tenant authorization
  - Complete UI overhaul



HORTONWORKS

Apache NiFi Subproject: MiNiFi

- Get the key parts of NiFi close to where data begins and provide bidirectional communication
- NiFi lives in the data center — give it an enterprise server or a cluster of them
- MiNiFi lives as close to where data is born and is a guest on that device or system
  - IoT
  - Connected car
  - Legacy hardware

HORTONWORKS

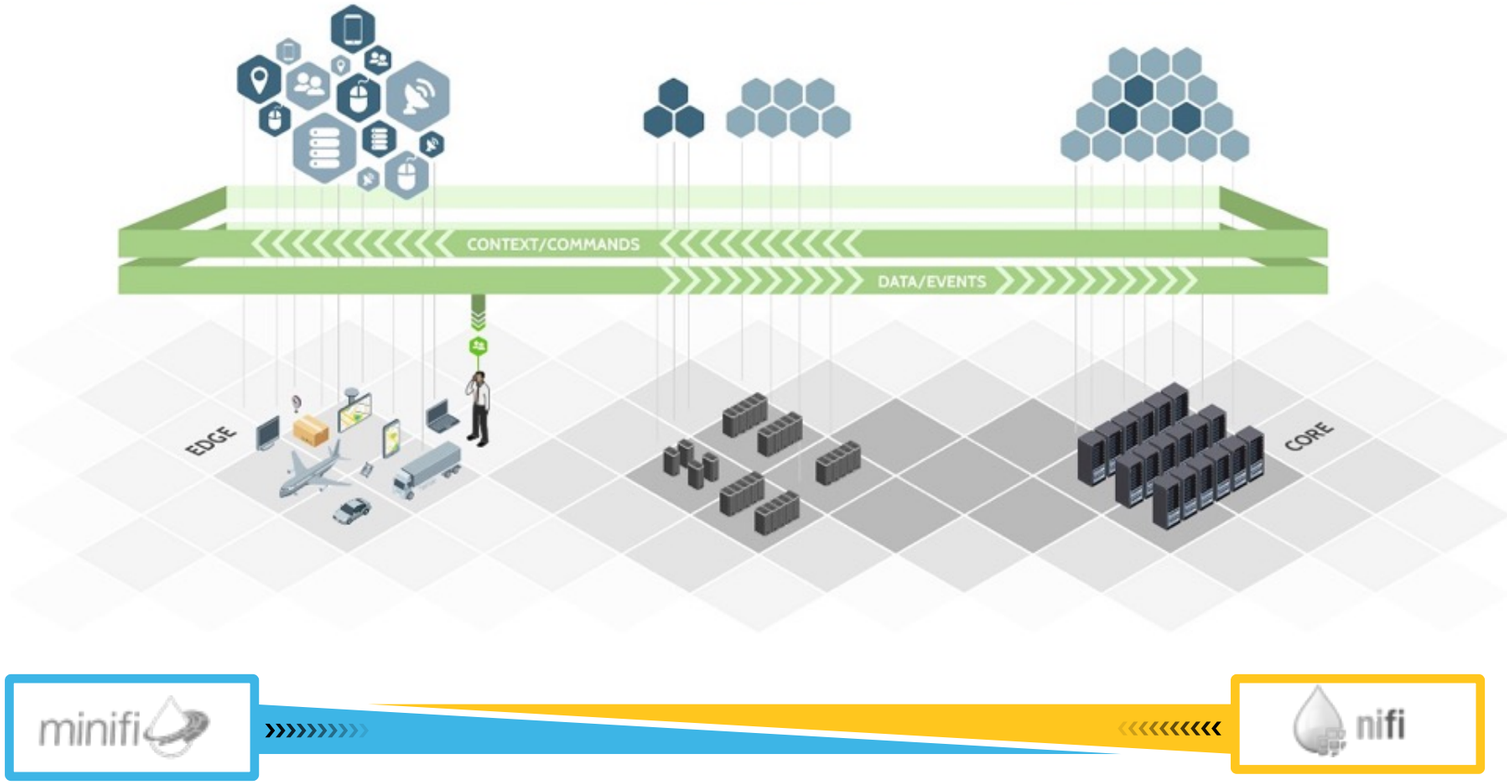
Why build MiNiFi?

- NiFi is big
  - 1.6.0 release is 1.2 GB compressed
  - Can be modified to run in restricted environments, but requires manual surgery
  - Provides UI, provenance query, etc.
  - Runs on dedicated machines/clusters — “owns the box”
- MiNiFi lives at the edge
  - No UI
  - 0.4.0 Java binary is 65 MB, C++ binary is 4.5 MB **(0.2.0 fits on a floppy disk)**
  - “Good guest”

HORTONWORKS

How Does MiNiFi Interact With NiFi?

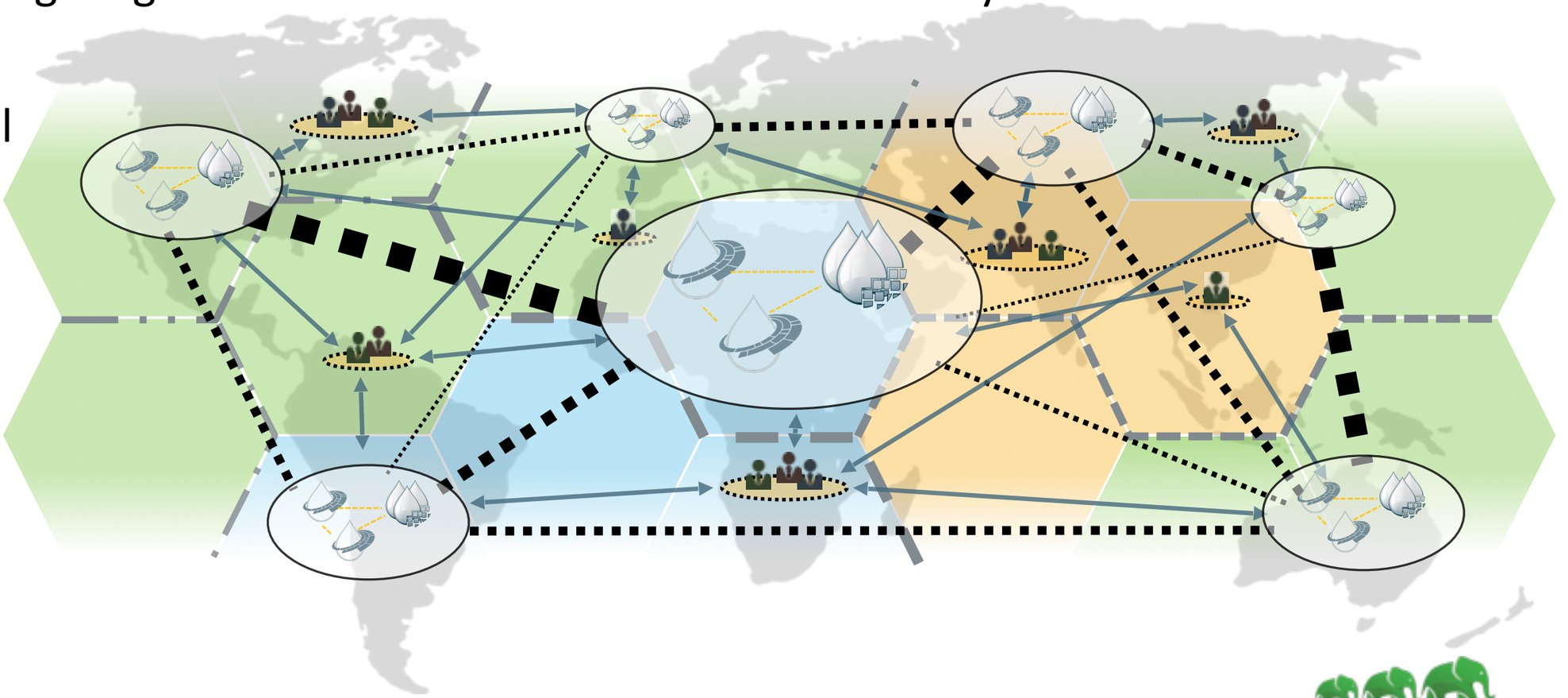
- NiFi
  - Design flows
  - Aggregate data from many sources
  - Perform routing/analysis/SEP
- MiNiFi
  - Receive flows
  - Collect data
  - Send for processing



HORTONWORKS

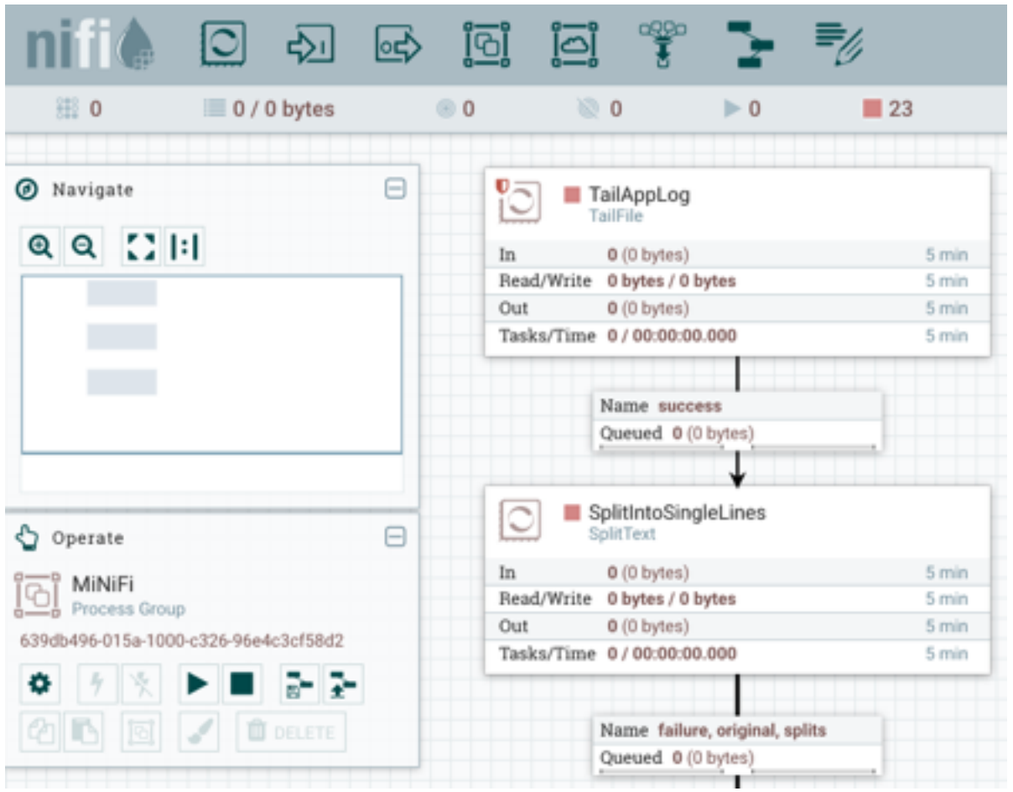
Let’s Add Dimensionality

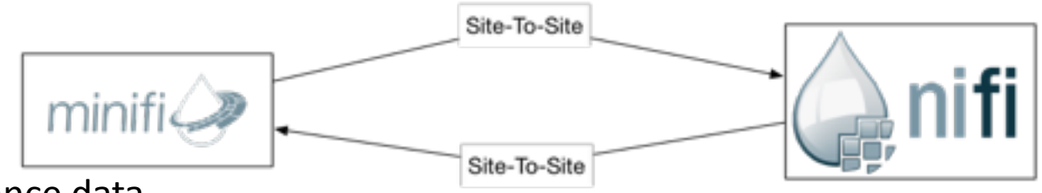
- We’ve been imagining EDGE to CORE as a bi-directional linear system
- Let’s expand that to the real world



Flavors of MiNiFi

- MiNiFi Java (v0.4.0)
  - Modified version of NiFi
    - No UI
    - YAML configuration
  - Reduced processor count
    - 110+ by default, more available with additional NARs
- MiNiFi C++ (v0.4.0)
  - Written from scratch
  - 28 processors by default
  - Bi-directional site-to-site & provenance data



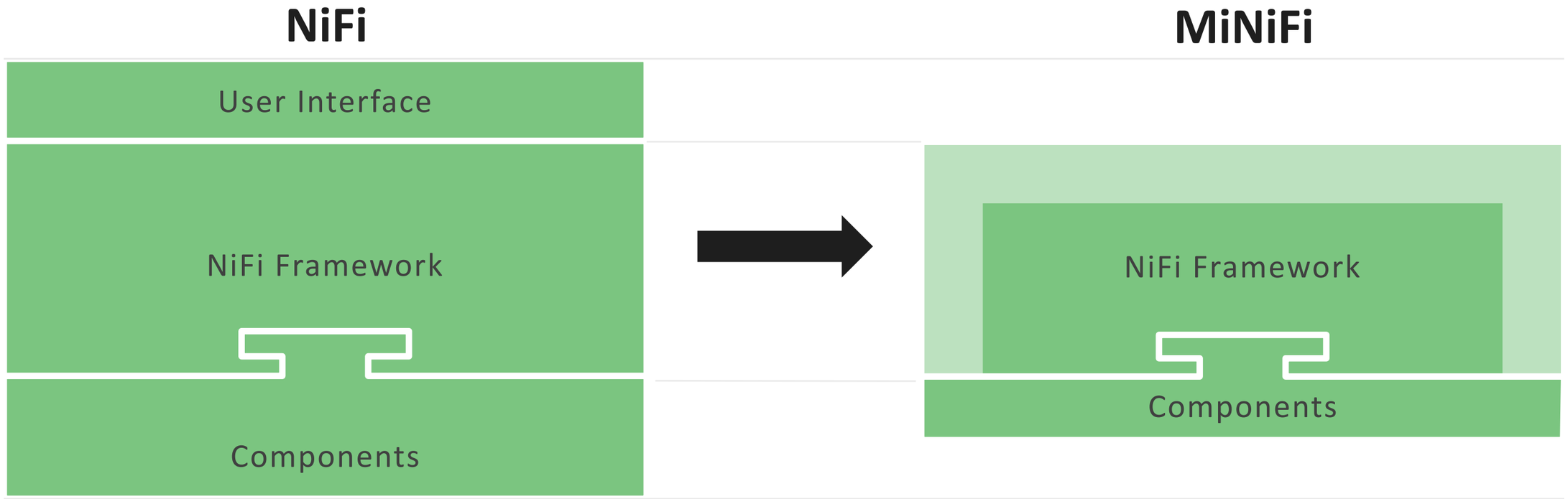



```
Security Properties:
    keystore: /tmp/ssl/localhost-ks.jks
    keystore type: JKS
    keystore password: localtest
    key password: localtest
    truststore: /tmp/ssl/localhost-ts.jks
    truststore type: JKS
    truststore password: localtest
    ssl protocol: TLS
    Sensitive Props:
        key:
        algorithm: PBEWITHMD5AND256BITAES-CBC-OPENSSL
        provider: BC

Processors:
    - name: TailAppLog
      class: org.apache.nifi.processors.standard.TailFile
      max concurrent tasks: 1
      scheduling strategy: TIMER_DRIVEN
      scheduling period: 10 sec
      penalization period: 30 sec
      yield period: 1 sec
      run duration nanos: 0
      auto-terminated relationships list:
      Properties:
          File to Tail: logs/minifi-app.log
          Rolling Filename Pattern: minifi-app*
          Initial Start Position: Beginning of File
    - name: SplitIntoSingleLines
      class: org.apache.nifi.processors.standard.SplitText
      max concurrent tasks: 1
      scheduling strategy: TIMER_DRIVEN
      scheduling period: 0 sec
      penalization period: 30 sec
      yield period: 1 sec
      run duration nanos: 0
      auto-terminated relationships list:
          - failure
          - original
      Properties:
          Line Split Count: 1
          Header Line Count: 0
          Remove Trailing Newlines: true
```


HORTONWORKS

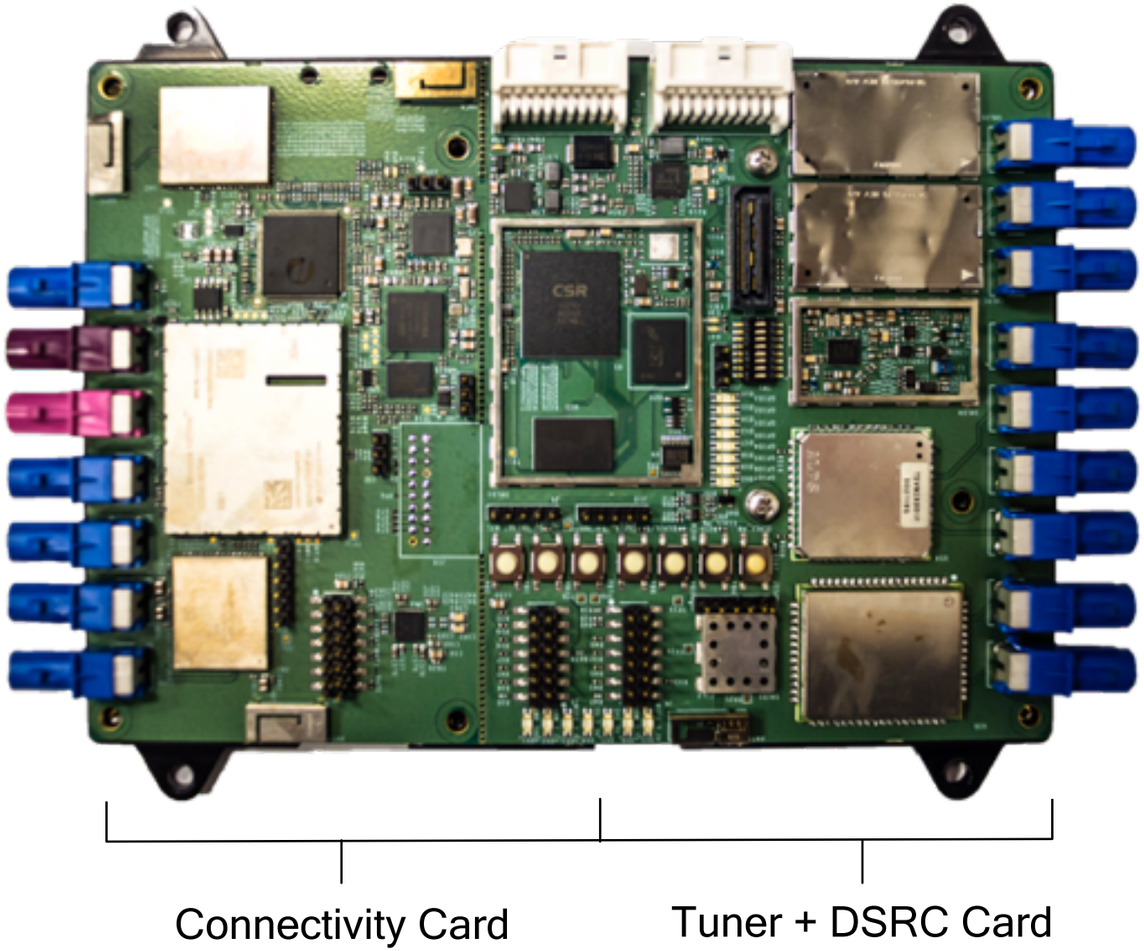
NiFi vs MiNiFi Java Processes



What does MiNiFi provide?

- Data tagging/provenance
- Governance from edge (geopolitical restrictions)
- Security (encryption, certificate-based authentication)
- Low latency (immediate reactions & decision-making)

Connected Car Reference Platform Box



MiNiFi Exfil

- Site-to-Site
  - NiFi protocol
  - Two implementations
    - Raw socket
    - HTTP(S) **(Java only)**
- Secured with mutual authentication TLS
  - HTTP(S), (S)FTP, JMS, Syslog, File, Email, Process **(Java only)**

HORTONWORKS

What’s Next?

HORTONWORKS

New Announcements

Introducing Apache NiFi Registry

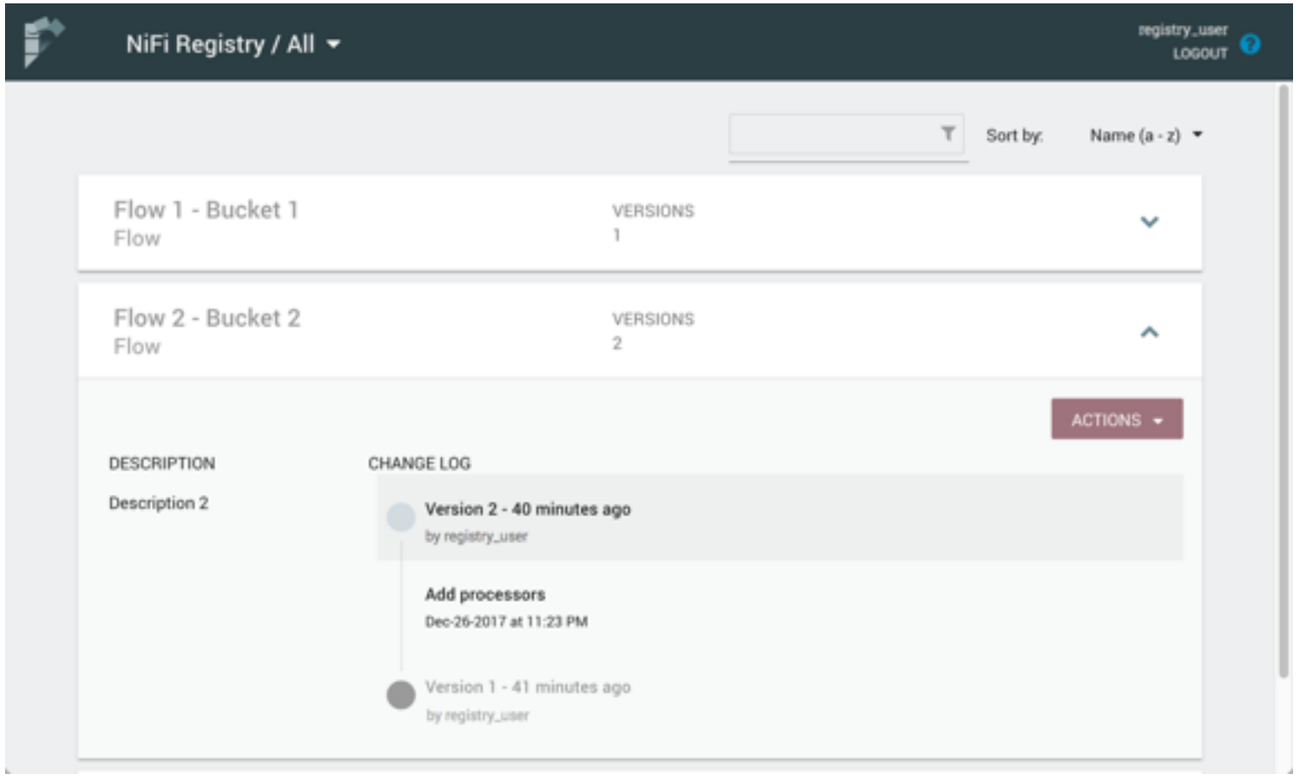
- NiFi 1.6.0 — 08 April 2018
  - MongoDB, InfluxDB, Druid, HBase components
  - Granular @Restricted components
- MiNiFi C++ 0.4.0 — 27 January 2018
- MiNiFi Java 0.4.0 — 22 January 2018
- NiFi Registry 0.1.0 — 1 January 2018

HORTONWORKS

NiFi Registry for Dataflows

Introducing Apache NiFi Registry 0.1.0

- Previously, flows were exported via XML templates
  - Didn’t contain sensitive values
  - Couldn’t be updated in-place
  - No tracking system
- NiFi Registry brings asset management as first-class citizen to NiFi
- Flows can be versioned
- Flows can be promoted between environments



Learn more at **Forget Duplicating Local Changes: Apache NiFi and the Flow Development Lifecycle (FDLC)**
Thursday 19/4 @ 1600, Room II

HORTONWORKS

Community Health

apache / nifi

Unwatch 145 | Unstar 896 | Fork 856

Code | Pull requests 130 | Projects 0 | Insights

Mirror of Apache NiFi

4,337 commits | 49 branches | 60 releases | 174 contributors | Apache-2.0

Unresolved: By Priority

Priority	Issues	Percentage
Blocker	2	
Critical	29	2%
Major	788	51%
Minor	647	42%
Trivial	75	5%

HORTONWORKS

Learn more and join us

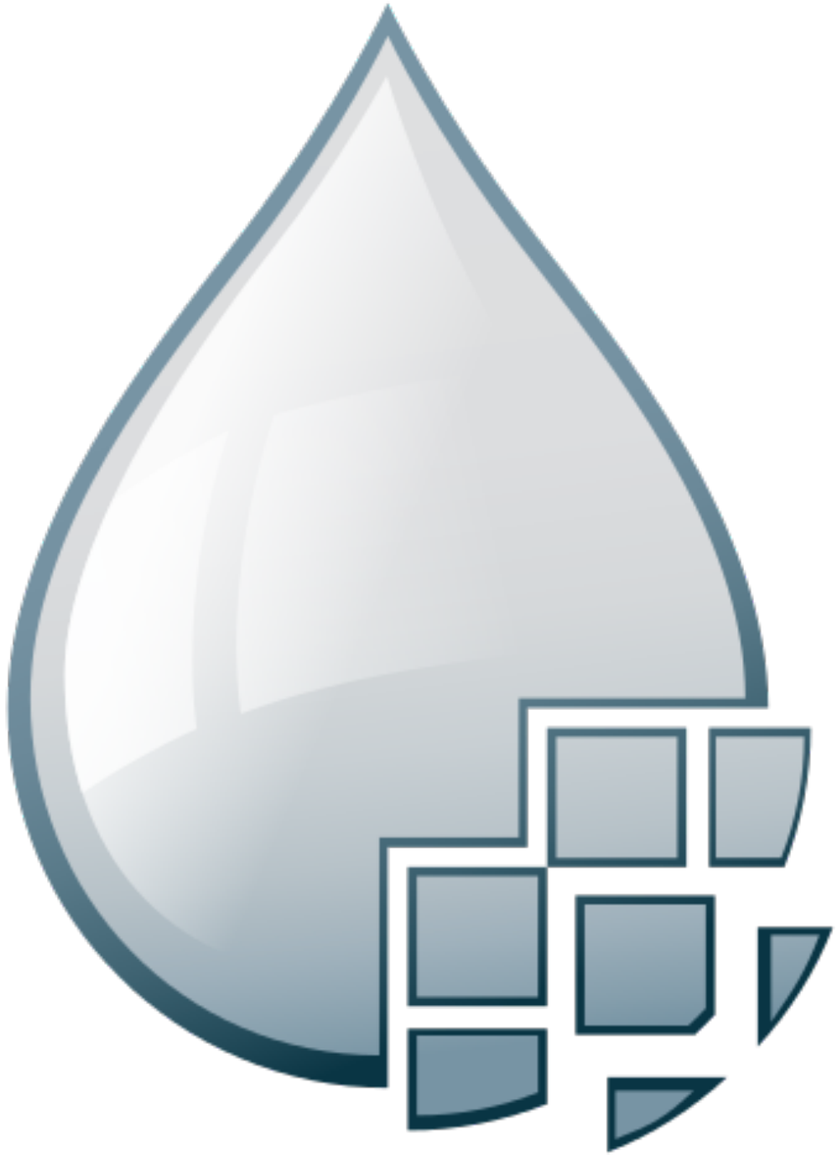
Apache NiFi site
https://nifi.apache.org

Subproject MiNiFi site
https://nifi.apache.org/minifi/

Subscribe to and collaborate at
dev@nifi.apache.org
users@nifi.apache.org

Submit Ideas or Issues
https://issues.apache.org/jira/browse/NIFI

Follow us on Twitter
@apachenifi



HORTONWORKS

More NiFi This Week…

Title	Room	Time	Speaker(s)
Apache NiFi Crash Course	Hall I - D	1115 - 1345	Andy LoPresto, Tim Spann
IoT with Apache MXNet and Apache NiFi and MiNiFi	Hall I - C	1150 - 1230	Tim Spann
Best practices and lessons learnt from Running Apache NiFi at Renault	Europe	1650 - 1730	Adel Gacem, Abdelkrim Hadjidj
From an experiment to a real production environment	Room V	1650 - 1730	Jeroen Wolffensperger, Martijn Groen
IoT, Streaming, and Dataflow Birds of a Feather	Room I	1740 - 1855	George Vetticaden, Davor Bonaci, Andy LoPresto, Stephan Ewen
Intelligently Collecting Data at the Edge — Intro to Apache MiNiFi	Room II	1100 - 1140	Andy LoPresto
The Power of Intelligent Flows: Realtime IoT Botnet Classification with Apache NiFi	Hall I - C	1400 - 1440	Andy LoPresto
Forget Duplicating Local Changes: Apache NiFi and the Flow Development Lifecycle (FDLC)	Room II	1600 - 1640	Andy LoPresto

HORTONWORKS

Questions?

(Just kidding, hold them until after Tim)

HORTONWORKS

Thank you

alopresto@hortonworks.com | alopresto@apache.org | @yolopey

github.com/alopresto/slides