Fluentd and v0.14 features

May 30, 2016

Masahiro Nakagawa

TREASURE DATA

What’s Fluentd?

- Data collector for unified logging layer
  - Streaming data transfer based on JSON / MessagePack
  - Written in Ruby and C extension
- Rubygems based various plugins
  - www.fluentd.org/plugins
- Working on lots of productions
  - www.fluentd.org/testimonials

CORE

- Divide & Conquer
- Buffering & Retries
- Error Handling
- Message Routing
- Parallelism

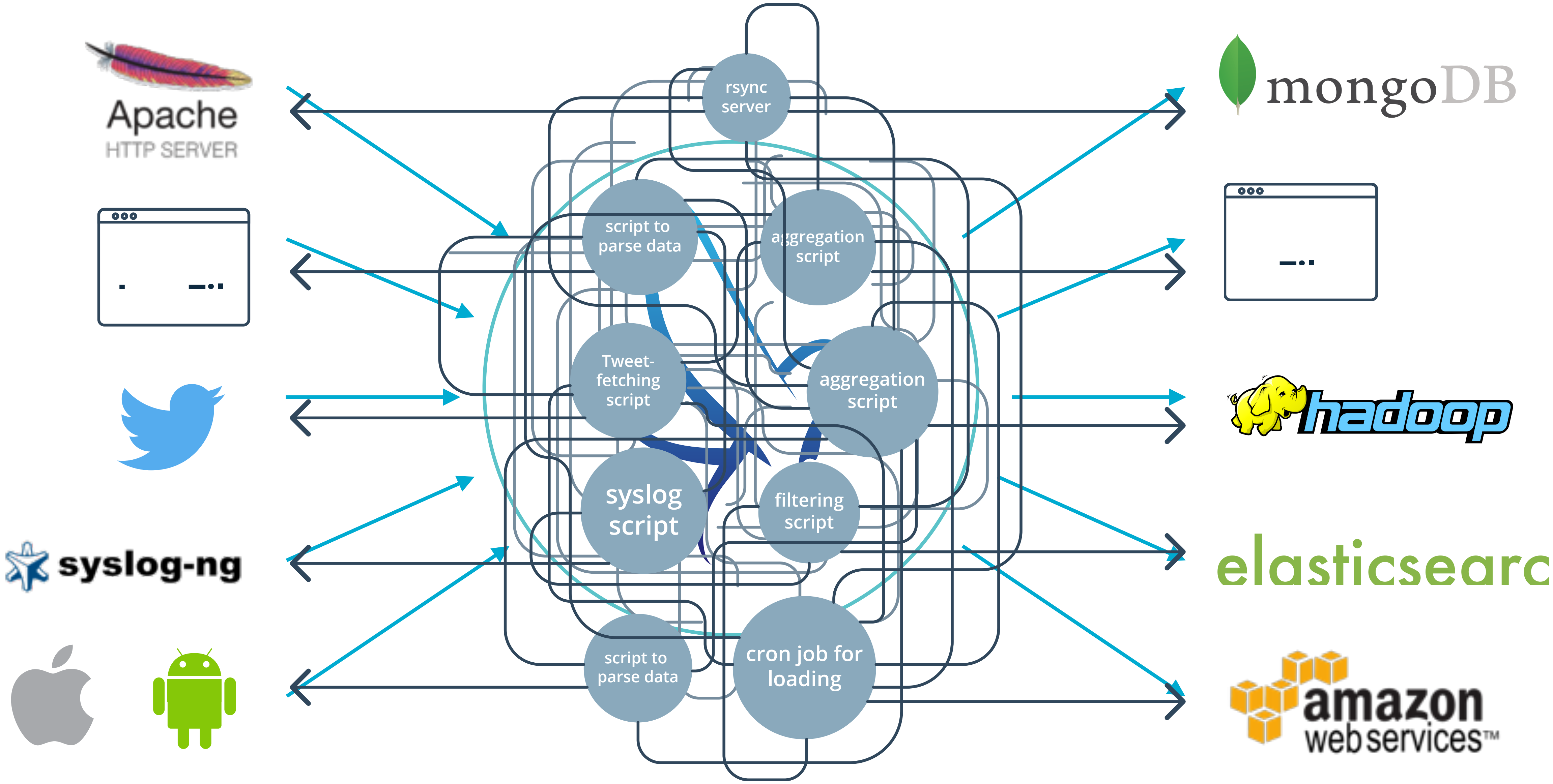
Common concerns

PLUGINS

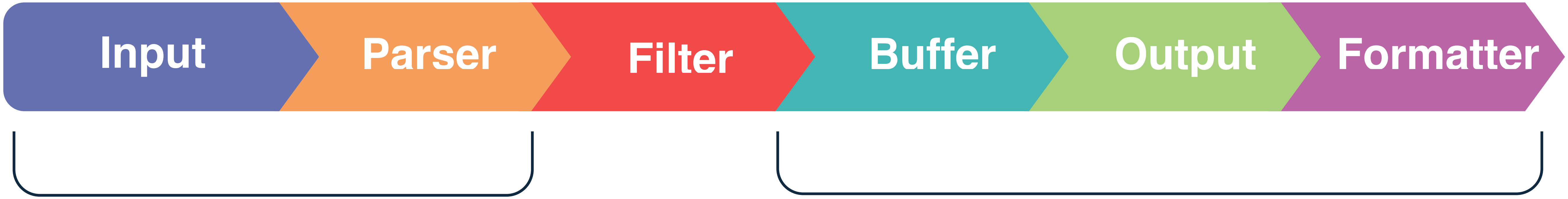
- Read Data
- Parse Data
- Buffer Data
- Write Data
- Format Data

Use case specific

Logging is a mess: M x N → M + N

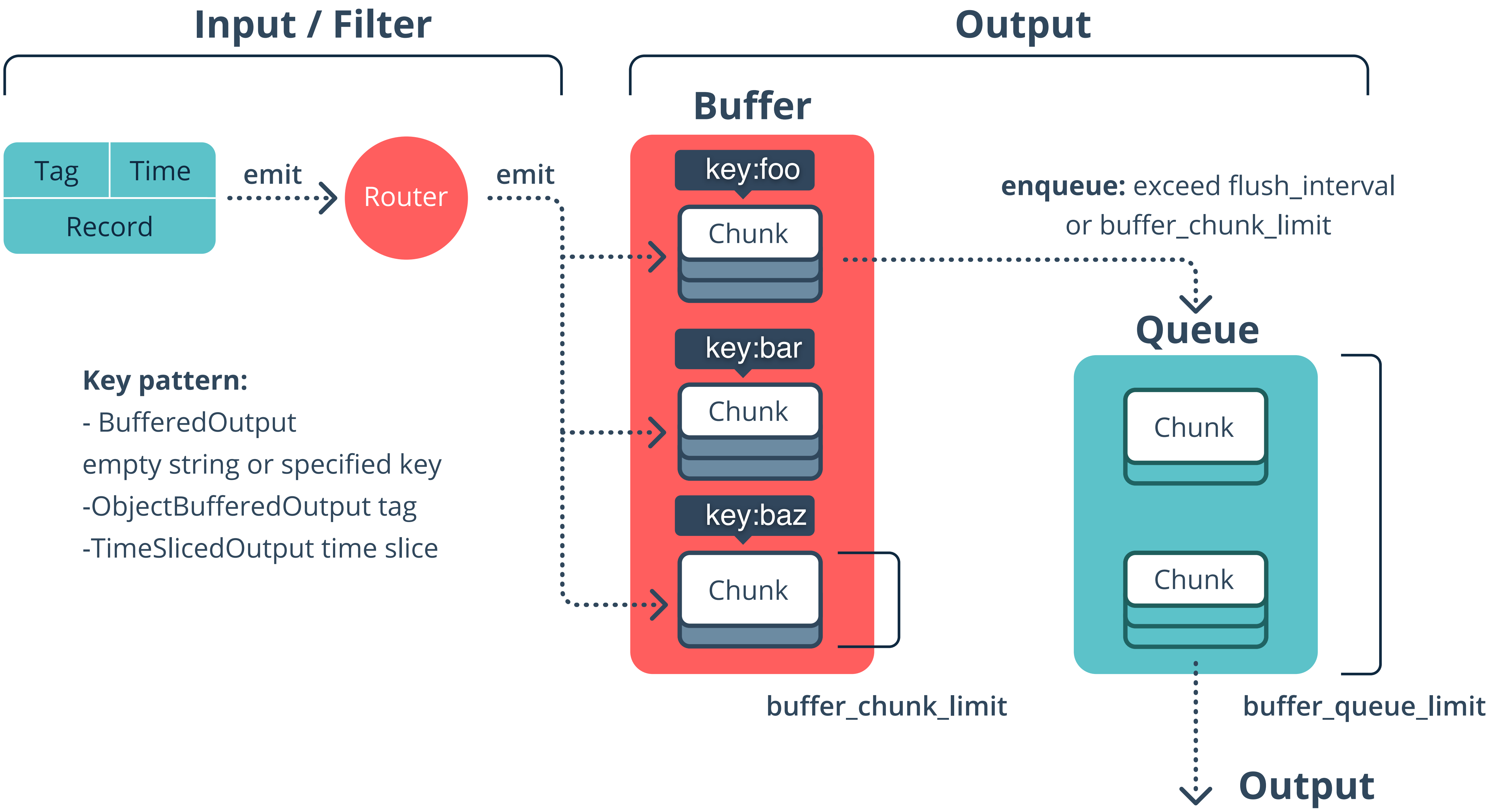


Internal Architecture



“input-ish”

“output-ish”



Input / Filter

Tag | Time

Record

emit

Router

emit

Output

Buffer

key:foo

Chunk

key:bar

Chunk

key:baz

Chunk

buffer_chunk_limit

enqueue: exceed flush_interval
or buffer_chunk_limit

Queue

Chunk

Chunk

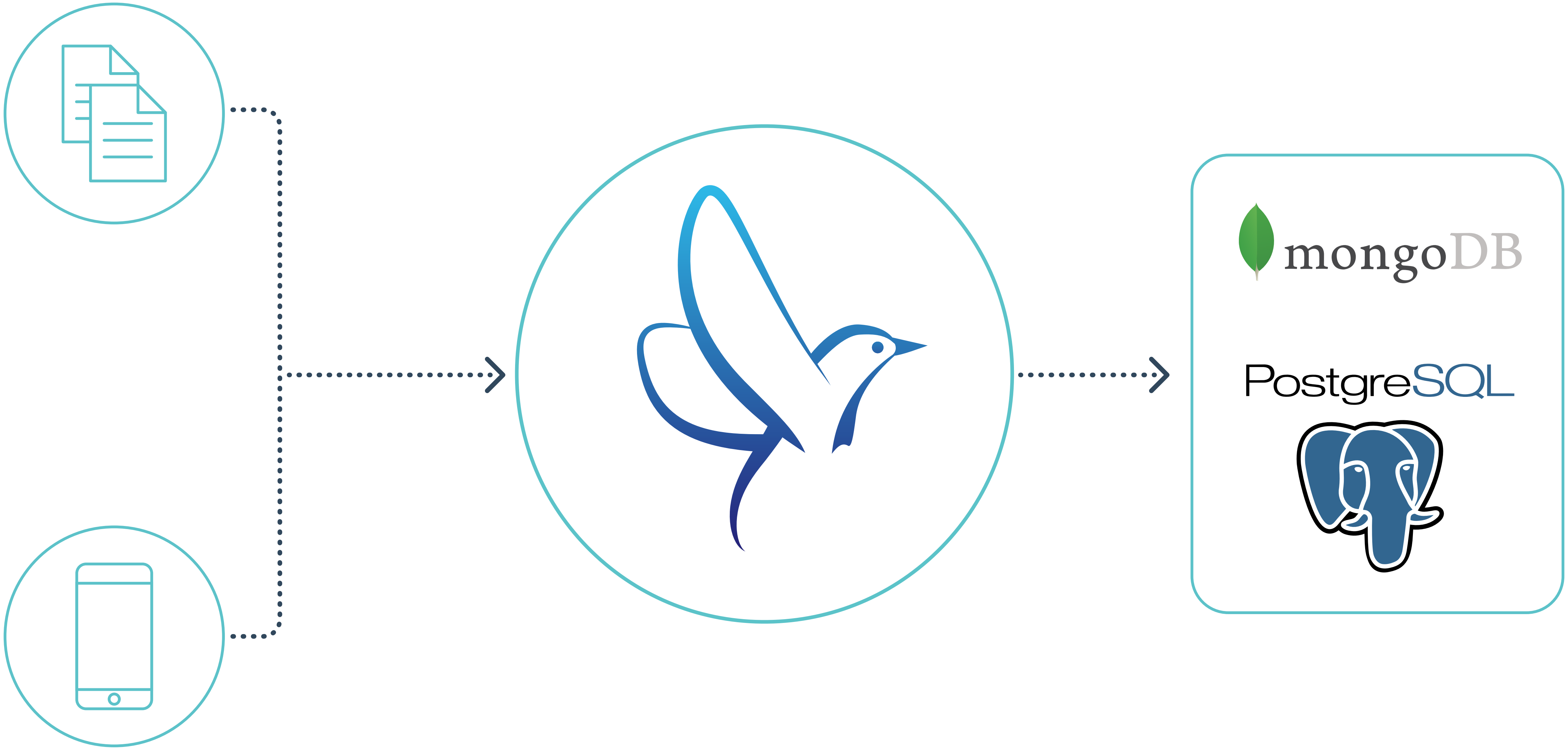
buffer_queue_limit

Output

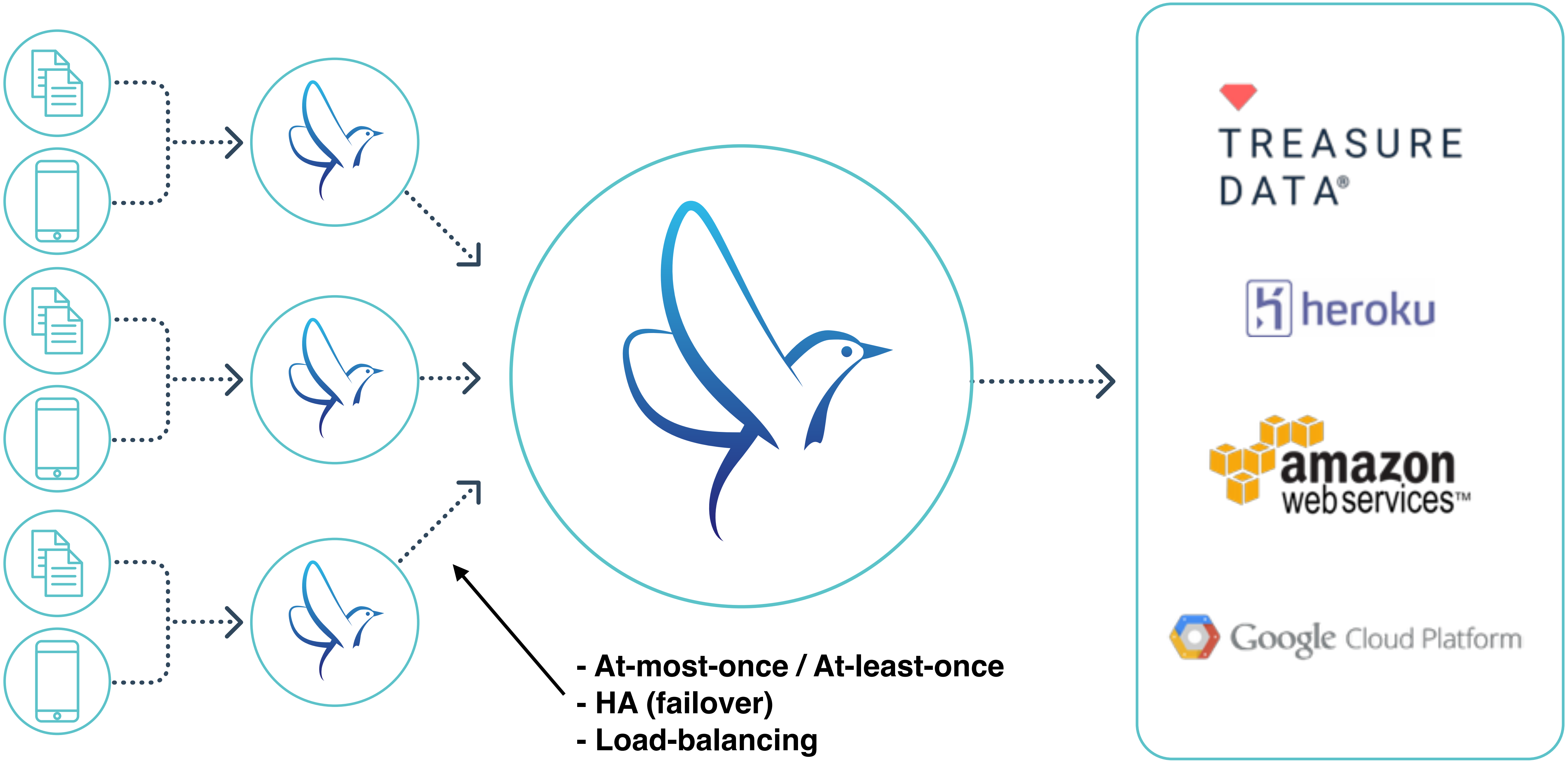
Key pattern:
- BufferedOutput
empty string or specified key
-ObjectBufferedOutput tag
-TimeSlicedOutput time slice

Use cases

Simple Forwarding

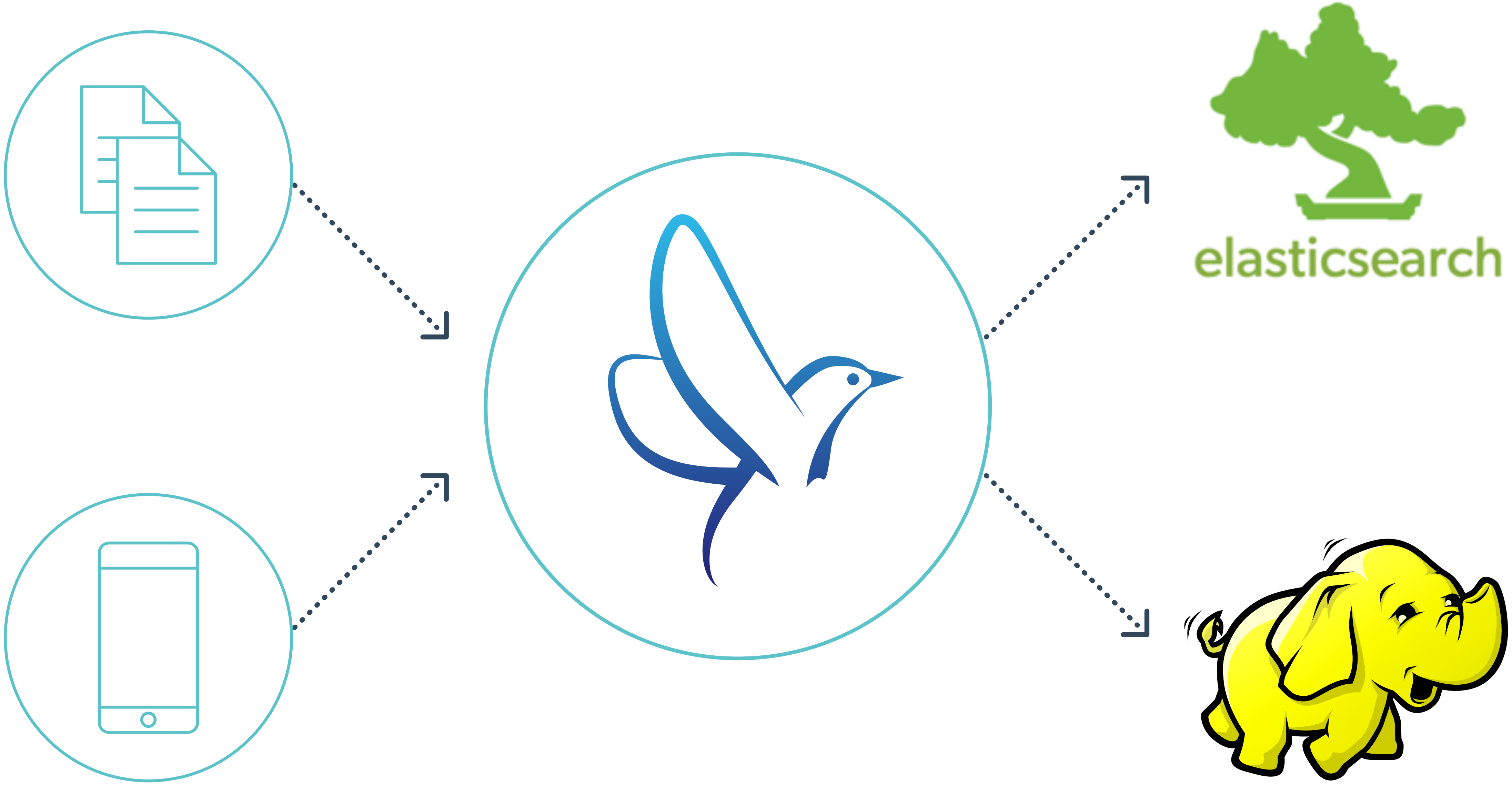


Less Simple Forwarding

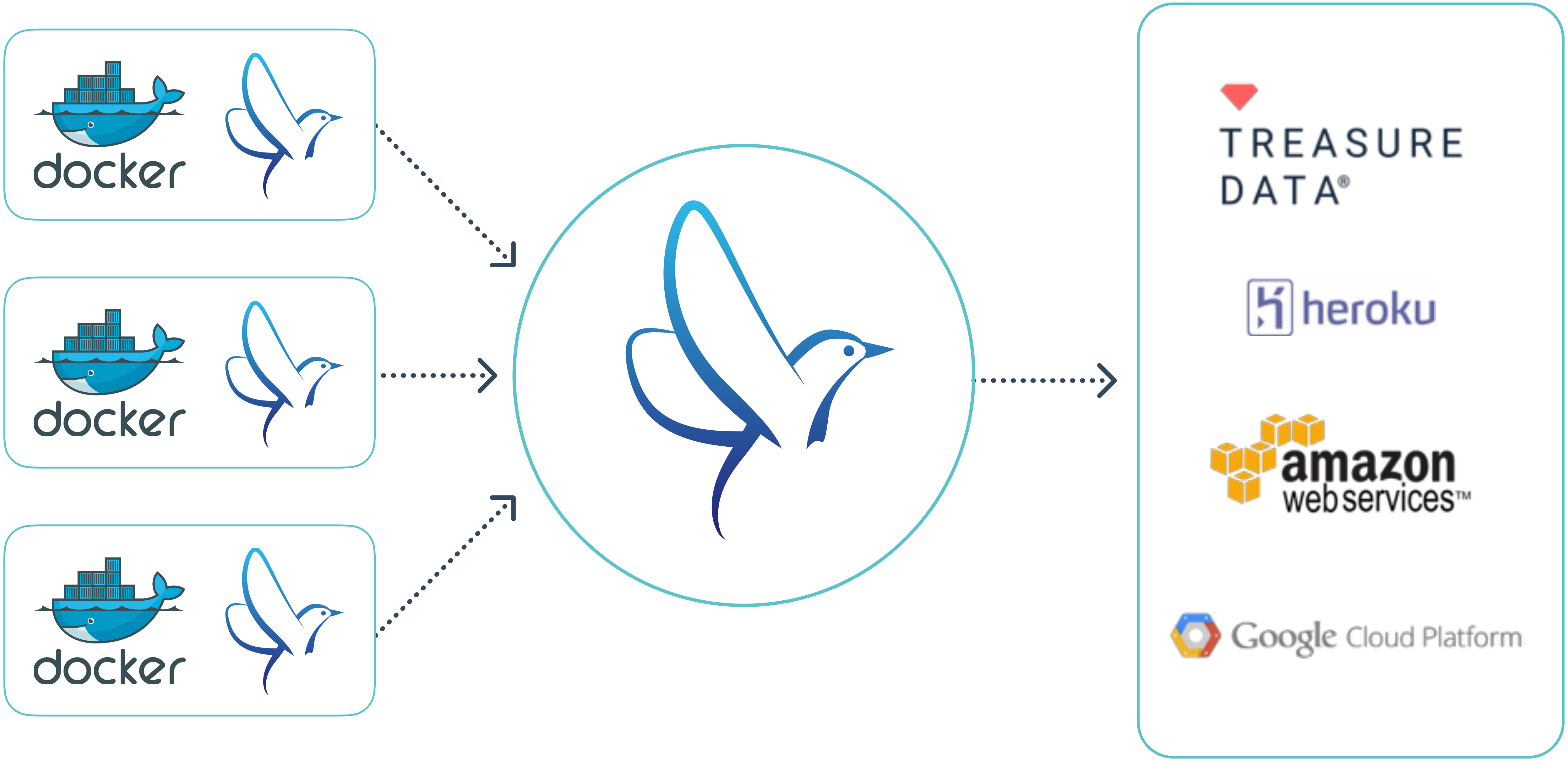


- At-most-once / At-least-once
- HA (failover)
- Load-balancing

Lambda Architecture



Container Logging



Roadmap

- v0.10.0 (In Oct 2011)
- v0.12.0 (In Dec 2014) -> Current stable
- v0.14.0 (In June 2016)
- v0.14.x (some versions in 2016)
- v1 (3Q or 4Q in 2016)

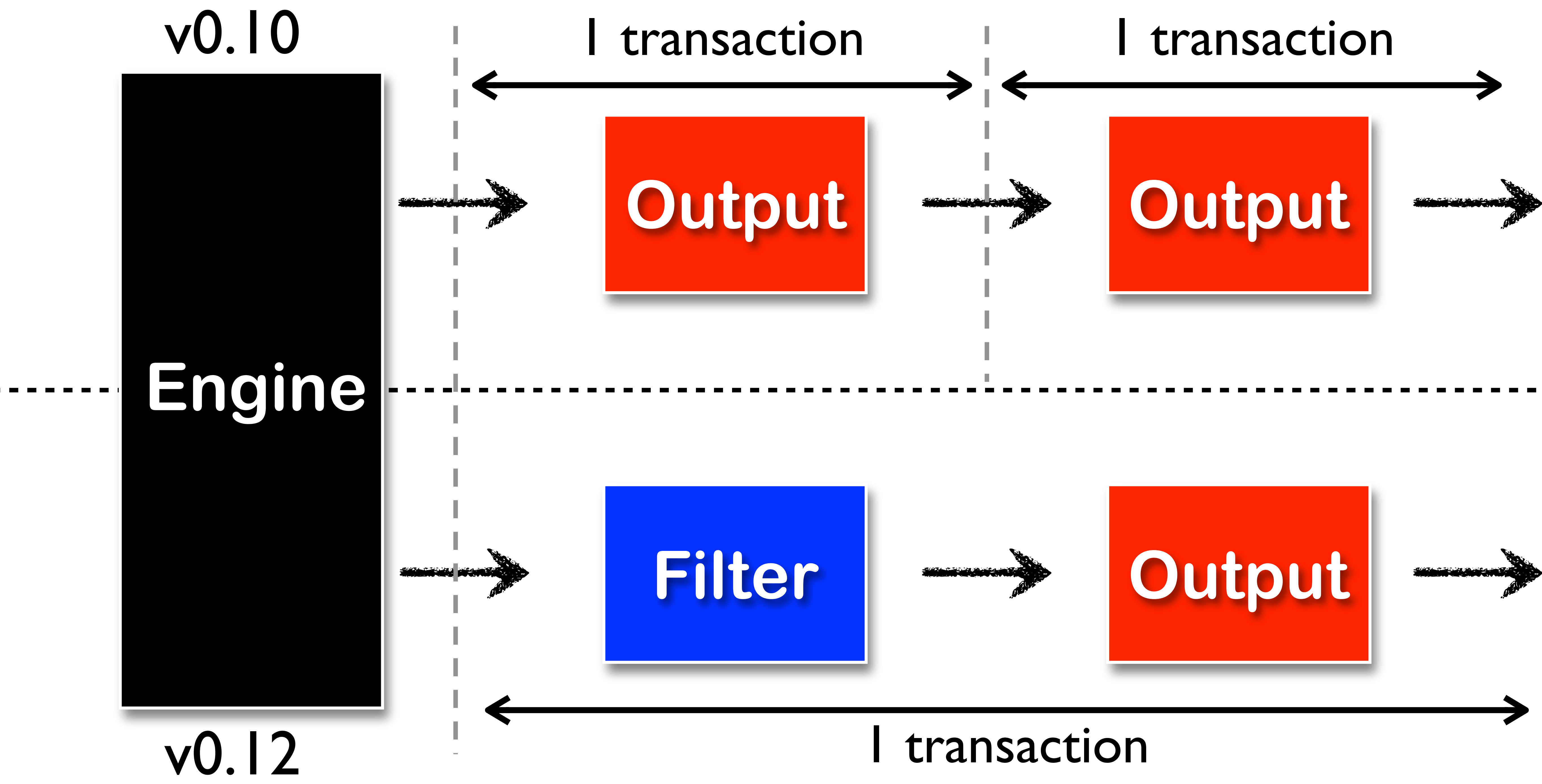
v0.10 (old stable)

- **First stable release**
  - Treasure Data provides td-agent 1
- **Mainly for log forwarding**
  - Only Input and Output
  - No complex event handling support
- **Only At-most-once semantics in forwarding**

v0.12 (current stable)

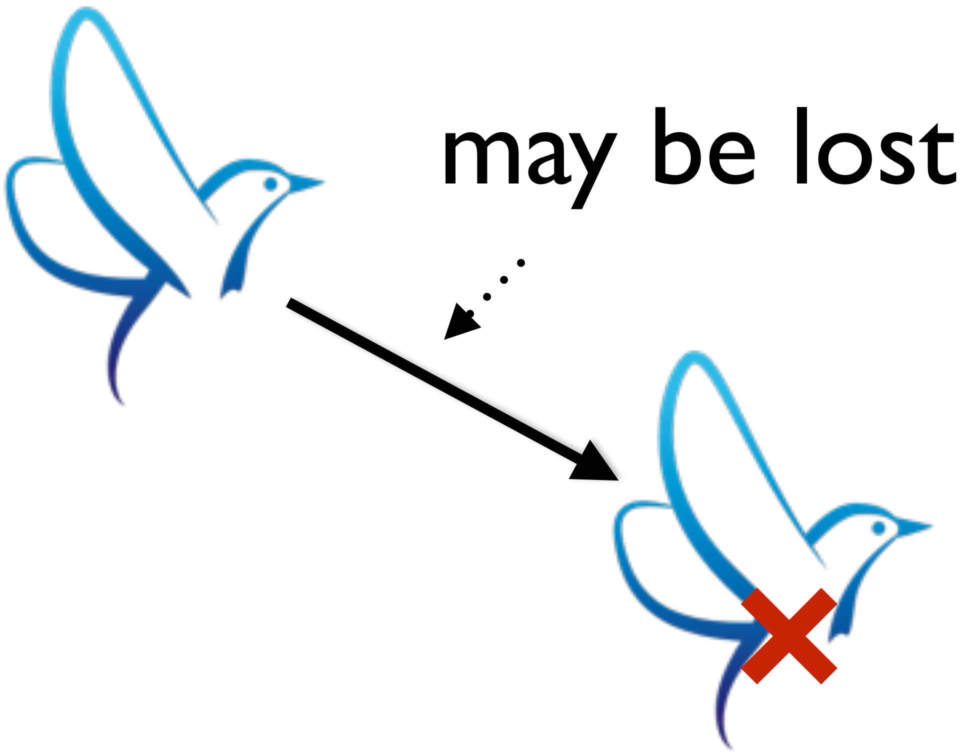
- **Event handling improvement**
  - Filter, Label, Error Stream
- **New configuration format**
- **Add at-least-once semantics in forwarding**
  - require_ack_response parameter
  - http://ogibayashi.github.io/blog/2014/12/16/try-fluentd-v0-dot-12-at-least-once/
- **HTTP RPC based process management**

Processing pipeline comparison



At-most-once and At-least-once

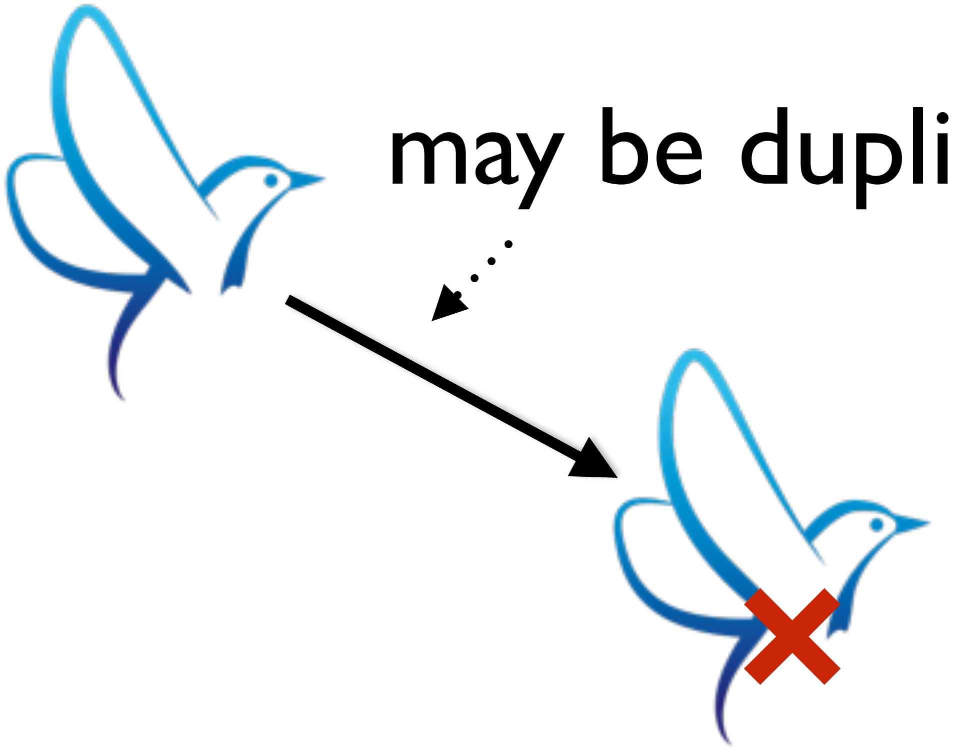
```
<match app.**>
  @type forward
</match>
```



may be lost

Error!

```
<match app.**>
  @type forward
  require_ack_response
</match>
```



may be duplicated

Error!

v0.14

- New Plugin APIs, Plugin Helpers & Plugin Storage
- Time with Nanosecond resolution
- Supervisor using ServerEngine
- Windows support

New Plugin APIs

- **Input/Output plugin APIs w/ well-controlled lifecycle**
  - stop, shutdown, close, terminate
- **New Buffer API for delayed commit of chunks**
  - parallel/async "commit" operation for chunks
- **100% Compatible w/ v0.12 plugins**
  - compatibility layer for traditional APIs
  - it will be supported between v1.x versions

Plugin Storage & Helpers

- **Plugin Storage: new plugin type for plugins**
  - provides key-value storage for plugins
  - to persistent intermediate status of plugins
  - built-in plugins (in plan): in-memory, local file
  - pluggable: 3rd party plugin to store data to Redis?
- **Plugin Helpers:**
  - collections of utility methods for plugins
  - making threads, sockets, network servers, ...
  - fully integrated with test drivers to run test codes after setup phase of helpers (e.g., after created threads started)

Time with nanosecond

- **Fluent::EventTime**
  - behaves as Integer (used as time in v0.12)
  - has methods to get sub-second resolution
  - be serialized into msgpack using Ext type
- **Fluentd core can handle both of Integer and EventTime as time**
  - compatible with older versions and software in eco-system (e.g., fluent-logger, Docker logging driver)

ServerEngine based Supervisor

- **Replacing supervisor process with ServerEngine**
  - it has SocketManager to share listening sockets between 2 or more worker processes
- **Replacing Fluentd's processing model from fork to spawn**
  - to support Windows environment

Windows support

- **Fluentd and core plugin work on Windows**
  - several companies have already used v0.14.0.pre version on production
  - We will send a patch to popular plugins if it doesn't work on Windows
- **Use HTTP RPC instead of signals**
- **Treasure Data will provide td-agent msi package**

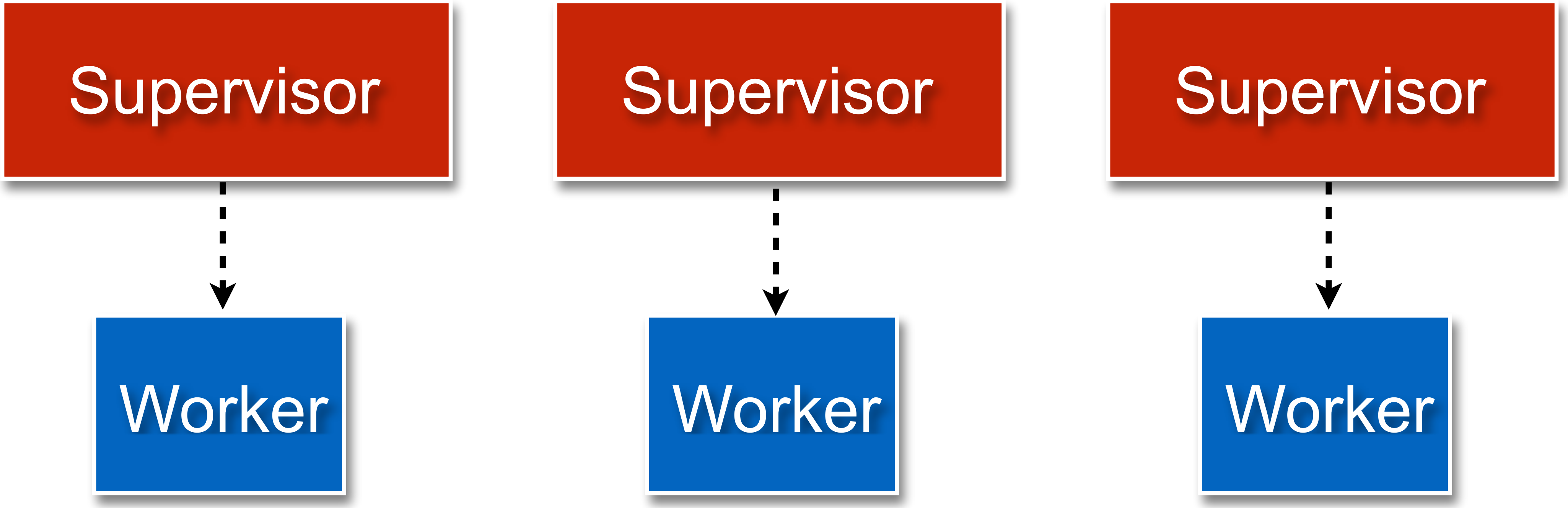
v0.14.x - v1

- **v0.14.x (some versions in 2016)**
  - Symmetric multi-core processing model
  - Counter API
  - TLS/authentication/authorization support (merging secure forward)
- **v1 (3Q or 4Q in 2016)**
  - Stable version for new APIs/features
  - fully compatible with v0.12.x

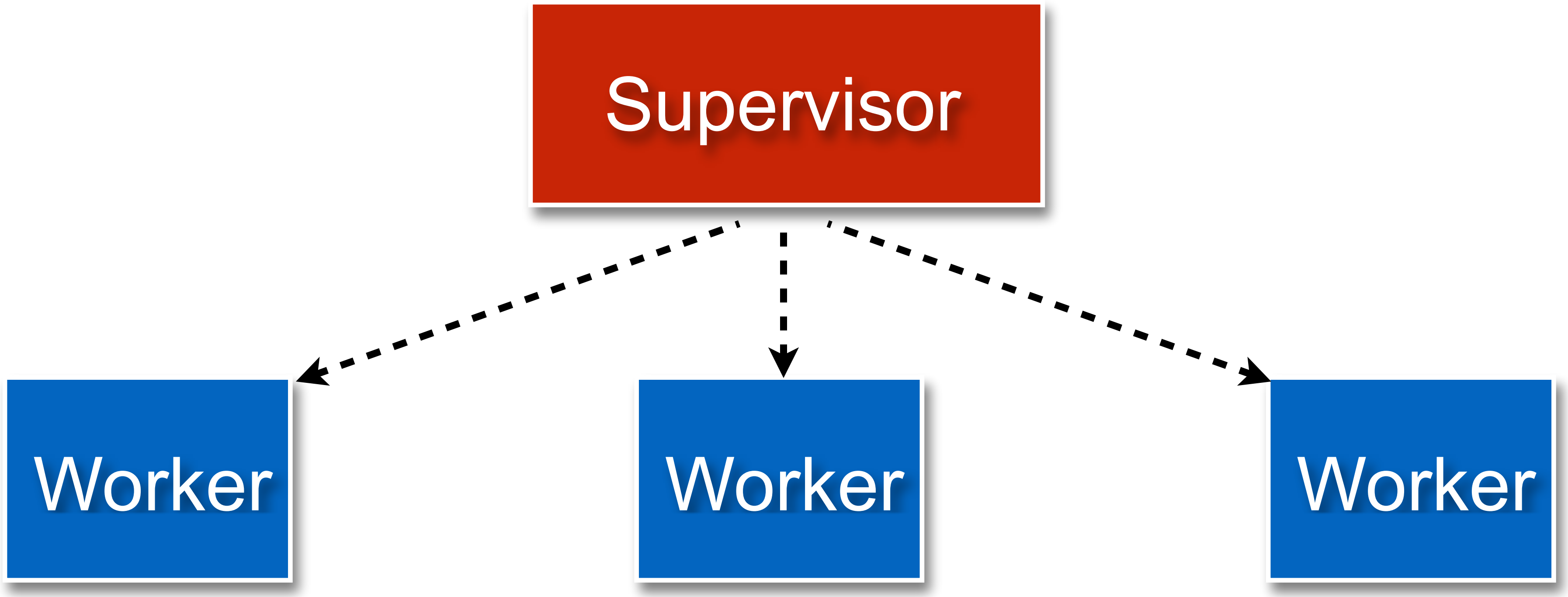
Symmetric multi core processing model

- **2 or more workers share a configuration file**
  - and share listening sockets via PluginHelper
  - under a supervisor process (ServerEngine)
- **Multi core scalability for huge traffic**
  - one input plugin for a tcp port, some filters and one (or some) output plugin
  - buffer paths are managed automatically by Fluentd core

Using fluent-plugin-multiprocess

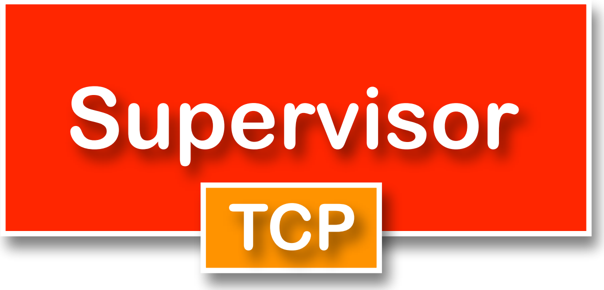


v0.14



Zero downtime restart

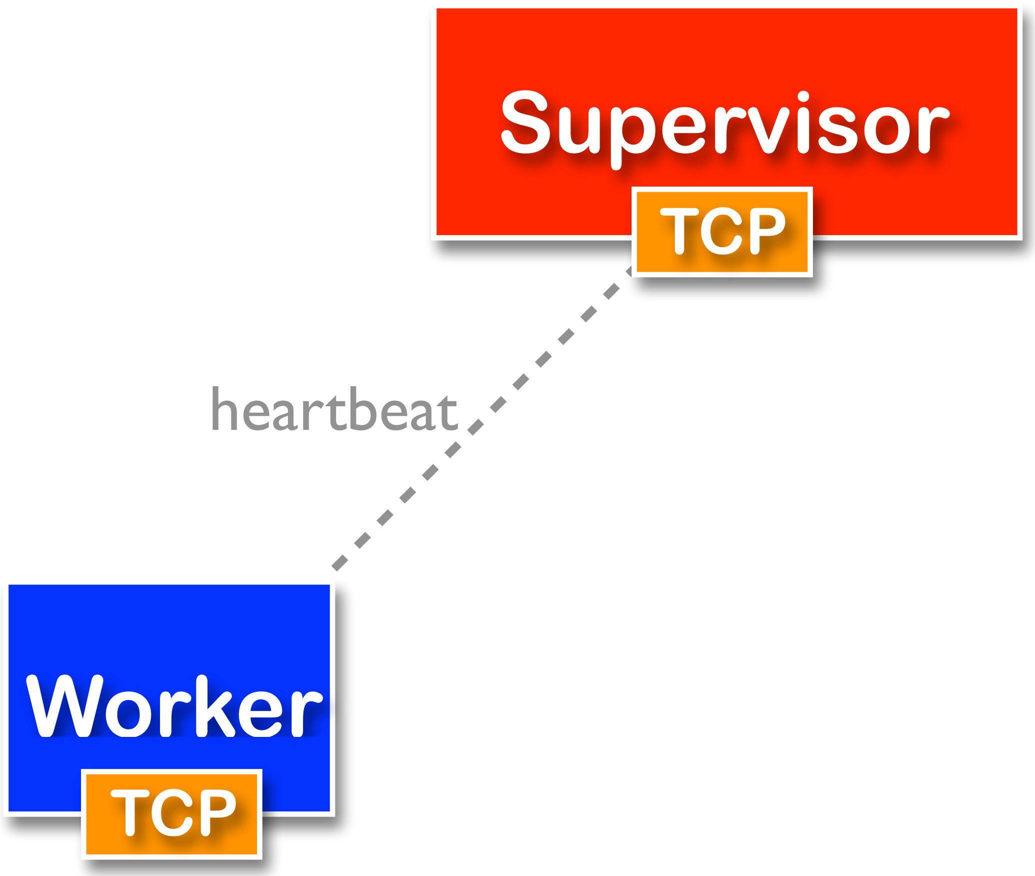
- SocketManager shares resources with workers



1. Listen to TCP socket

Zero downtime restart

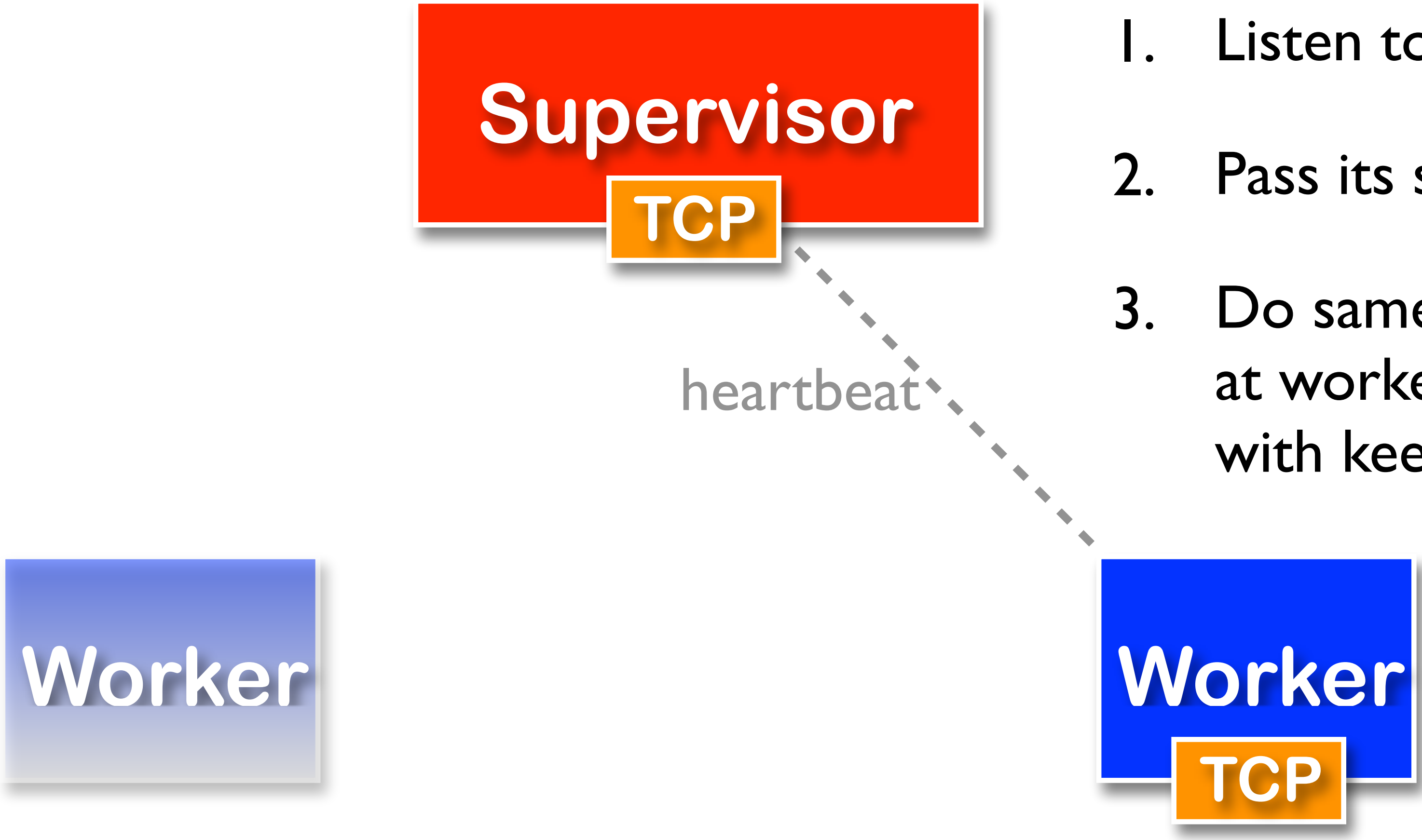
- SocketManager shares resources with workers



1. Listen to TCP socket
2. Pass its socket to worker

Zero downtime restart

- SocketManager shares resources with workers



1. Listen to TCP socket
2. Pass its socket to worker
3. Do same action
   at worker restarting
   with keeping TCP socket

Counter API

- **APIs to increment/decrement values**
  - shared by some processes
  - persisted on disk backed by Storage API
- **Useful for collecting metrics or stats filters**

TLS/Authn/Authz support for forward plugin

- **secure-forward will be merged into built-in forward**
  - TLS w/ at-least-one semantics
  - Simple authentication/authorization w/ non-SSL forwarding
- **Authentication and Authorization providers**
  - Who can connect to input plugins?
    What tags are permitted for clients?
  - New plugin types (3rd party authors can write it)
  - Mainly for in/out forward, but available from others