PayPal Risk Data Access Platform

The Road to Achieve High Availability and Low Latency

Ruth Cao

Software Development Manager, PayPal

Ruth Cao

曹若沈

2016 – Present

Software Development Manager of PayPal Risk

2008 - 2016

Development Lead of Morgan Stanley Risk Management Technology

Agenda

The Road to Achieve High Availability & Low Latency

- Introduction to PayPal Risk Management

- PayPal Risk Data Access Platform

- Best Practices and Lessons Learnt
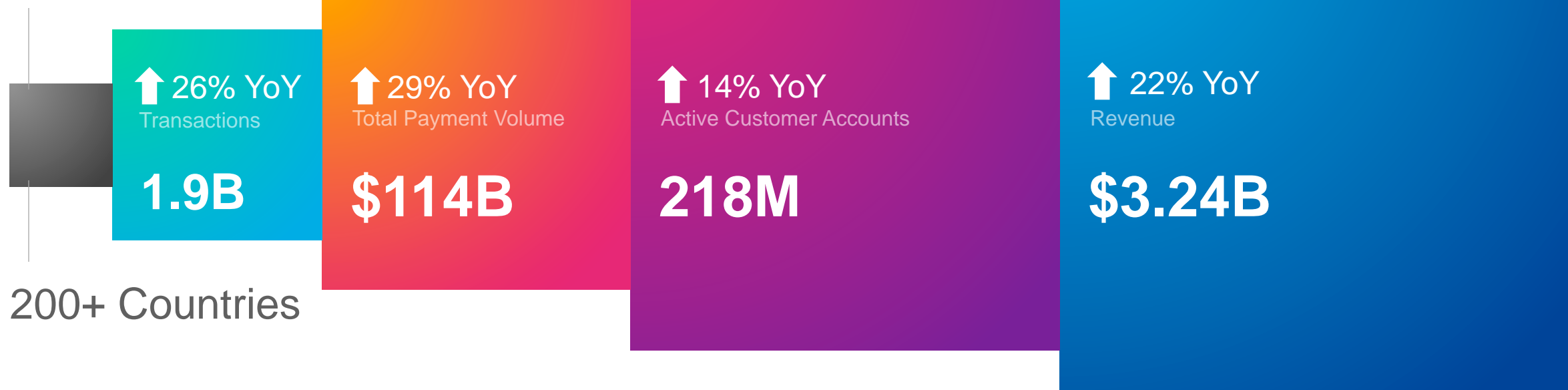
- Future Work

- Q & A

PayPal: a Leading Digital Payments Company

Q3 2017 Results

25 Currencies

** hold balances

⬆ 26% YoY
Transactions

1.9B

⬆ 29% YoY
Total Payment Volume

$114B

⬆ 14% YoY
Active Customer Accounts
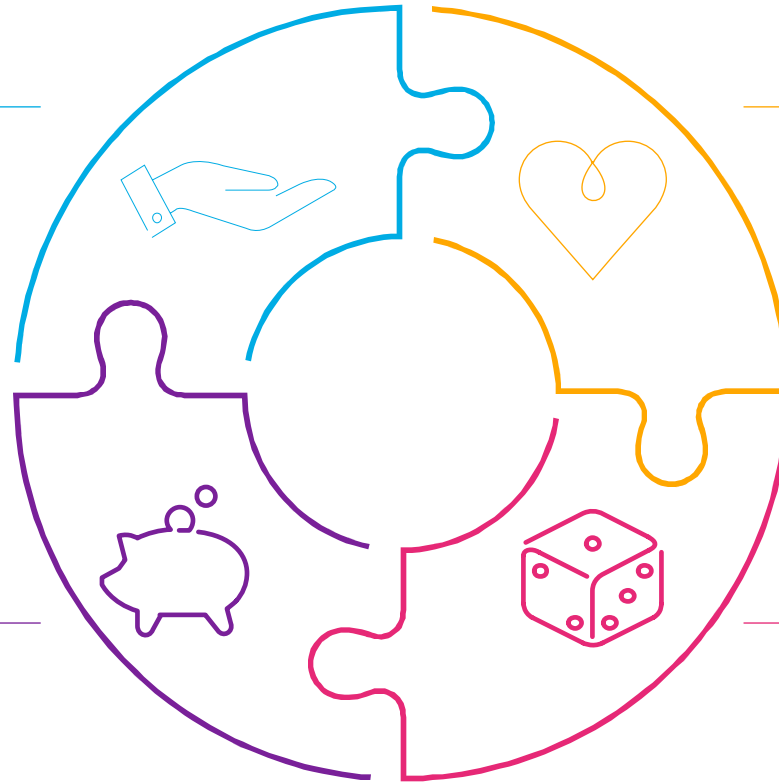
218M

⬆ 22% YoY
Revenue

$3.24B

200+ Countries

Risk Management is a Competitive Advantage for PayPal

Customer Trust

- Never share financial information with merchants
- The most trusted partner for mobile payments

Ease of Use

- Allow consumers to open a new account and transact immediately
- Real time risk decision making process

Financial Leverage

- Losses among lowest in the industry

Customer First

- Strong buyer and seller protection
- Lowest false positives in our history

# Risk Data Access Platform Business Requirements

Loss Reduction & Customer Protection

- 50+ Data-intensive Models

- 10,000+ Online Variables

- 1000+ Rules and Other Data Points

Great Customer Experience

- Real Time Risk Decision Making Process

- Transparent to Good Users

- Very Tight SLA
  - Light Weight Decisions need to be made within 50-100ms
  - Deeper Inspection are completed in 200-800ms
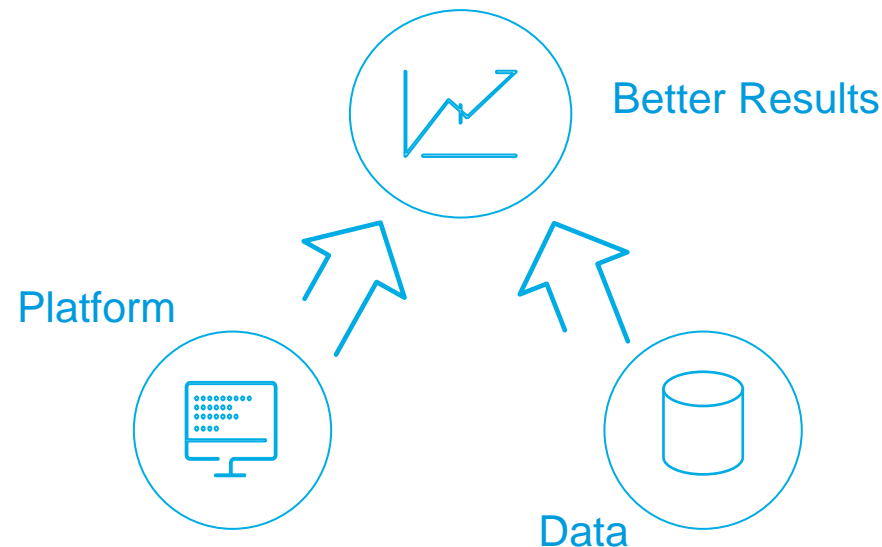
Better Results

Platform

Data

Agenda

The Road to Achieve High Availability & Low Latency

- Introduction to PayPal Risk Management

- **PayPal Risk Data Access Platform**

- Best Practices and Lessons Learnt

- Future Work

- Q & A

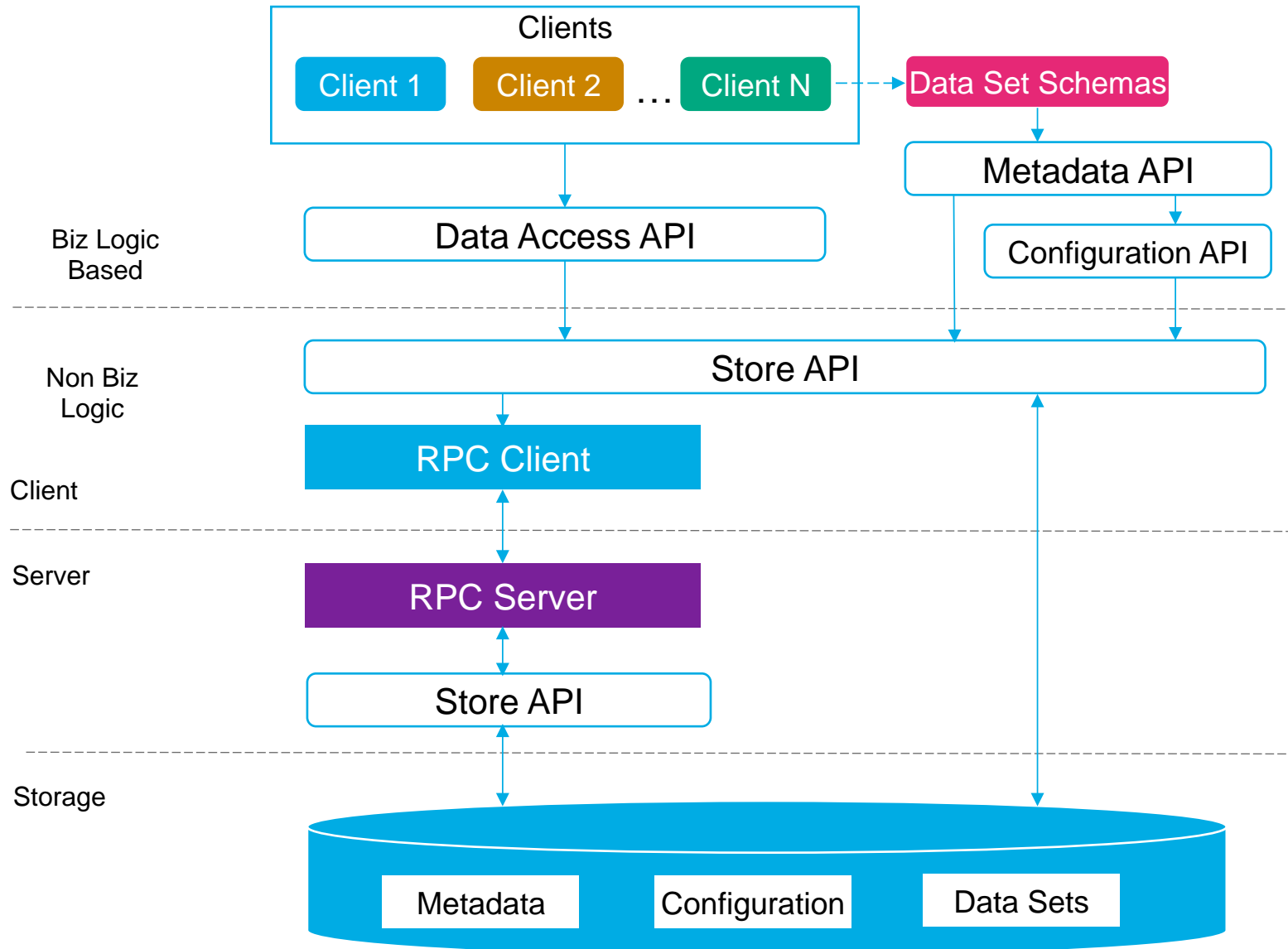# PayPal Risk Data Access Platform

Technical Asks

- Data Location Transparency

- Four 9's Availability

- High Performance

Design Principles

- ✓ Data Access Abstraction

- ✓ Metadata Driven

- ✓ Fully-asynchronous Design

Architecture

Data Access Abstraction Challenges
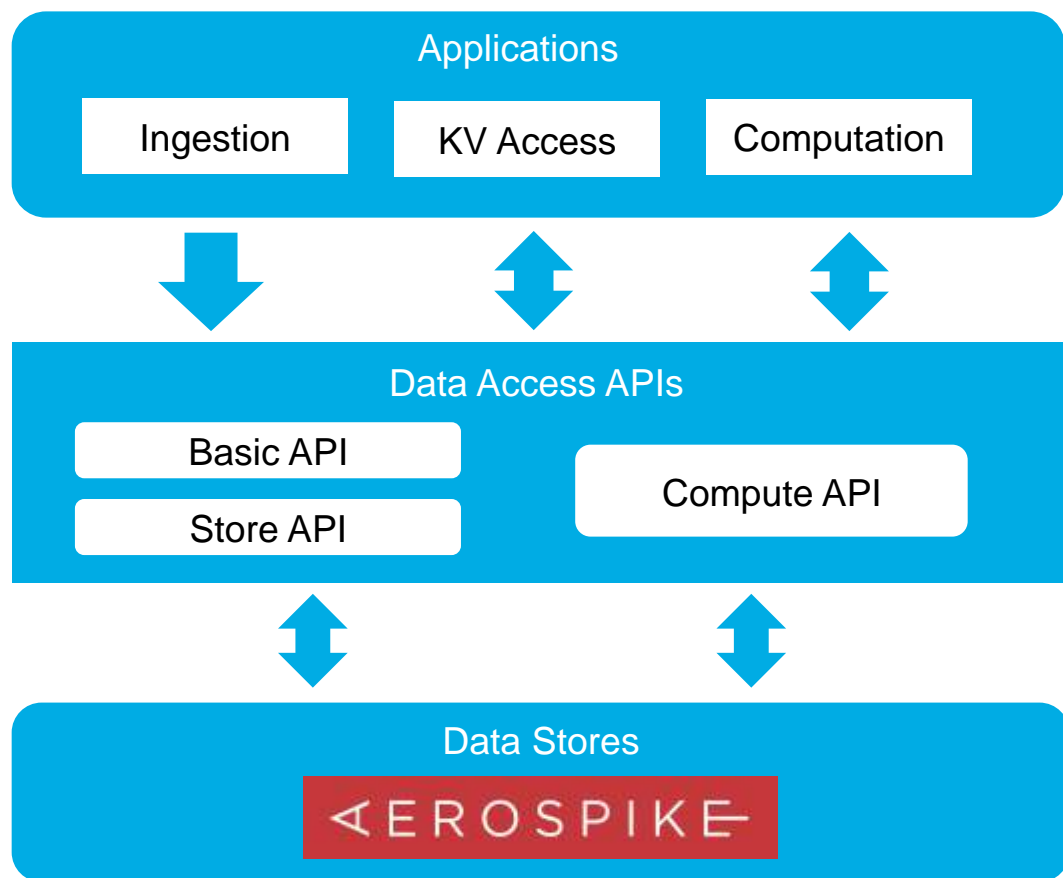
2

Stacks

15

Clusters

130

Client Components

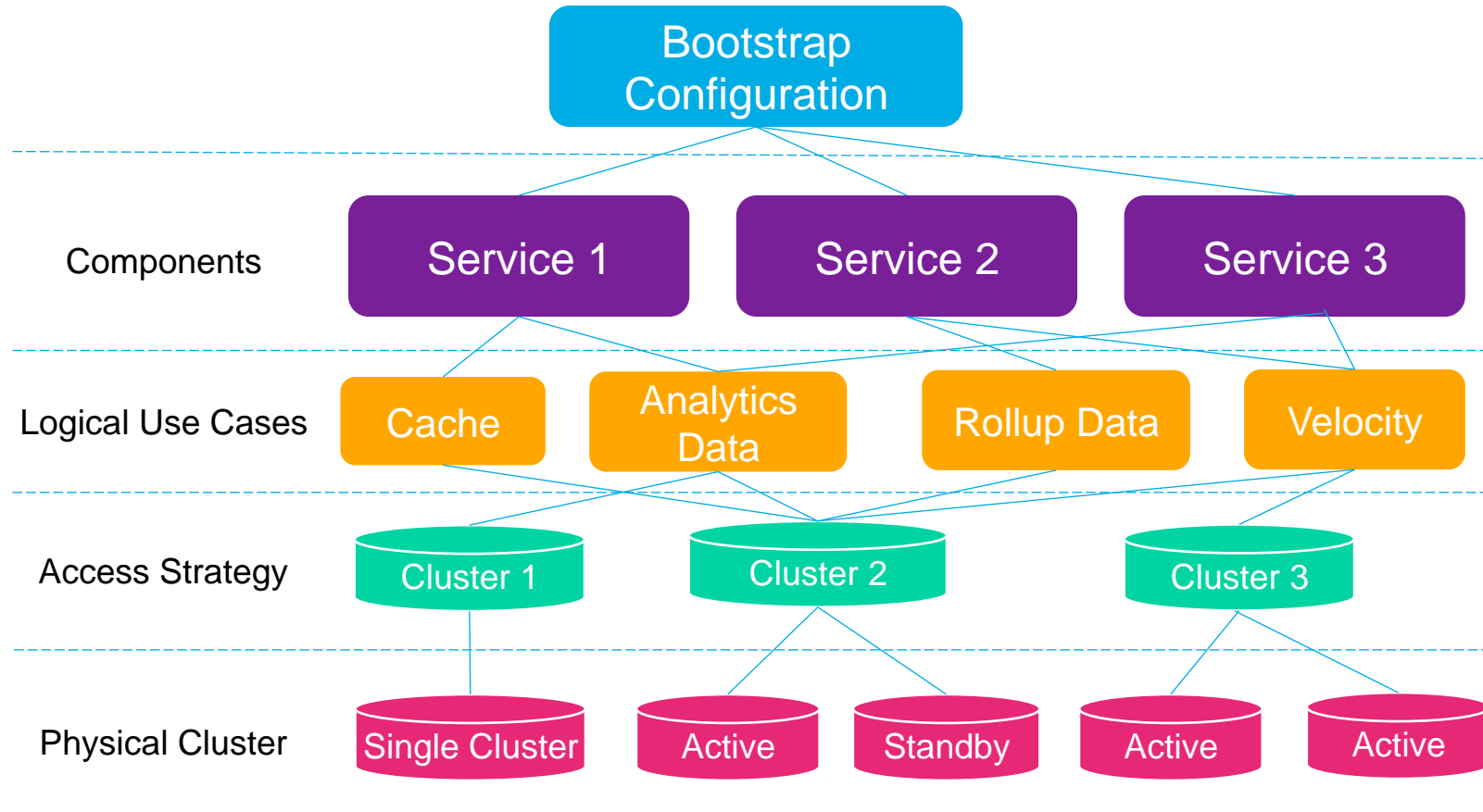# Data Access Abstraction Solution

- Data Store Agnostic
  - Support Multiple Key-Value Products
  - Migrate to the Newest Data Platforms Automatically

- Data Location Transparency
  - Integrate Once and Access Data Anywhere

- Intelligent Client
  - Simplify Client Integration and Offers Different Connection Modes

- Maximize Underlying Data Store Capability
  - User Defined Function in Aerospike

Why Metadata Driven Approach?

Pain Points	Solution
• Configuration Scattered Everywhere	✓ Single Source of Truth ✓ On-the-fly Refresh ✓ Multi-layer Backup
• Wrong Boundary	✓ Allow Client Side Override ✓ Physical/Logical Mapping ✓ Various Access Strategies

Unified Configuration System

Key Performance Metrics

High Availability

99.99% availability to business
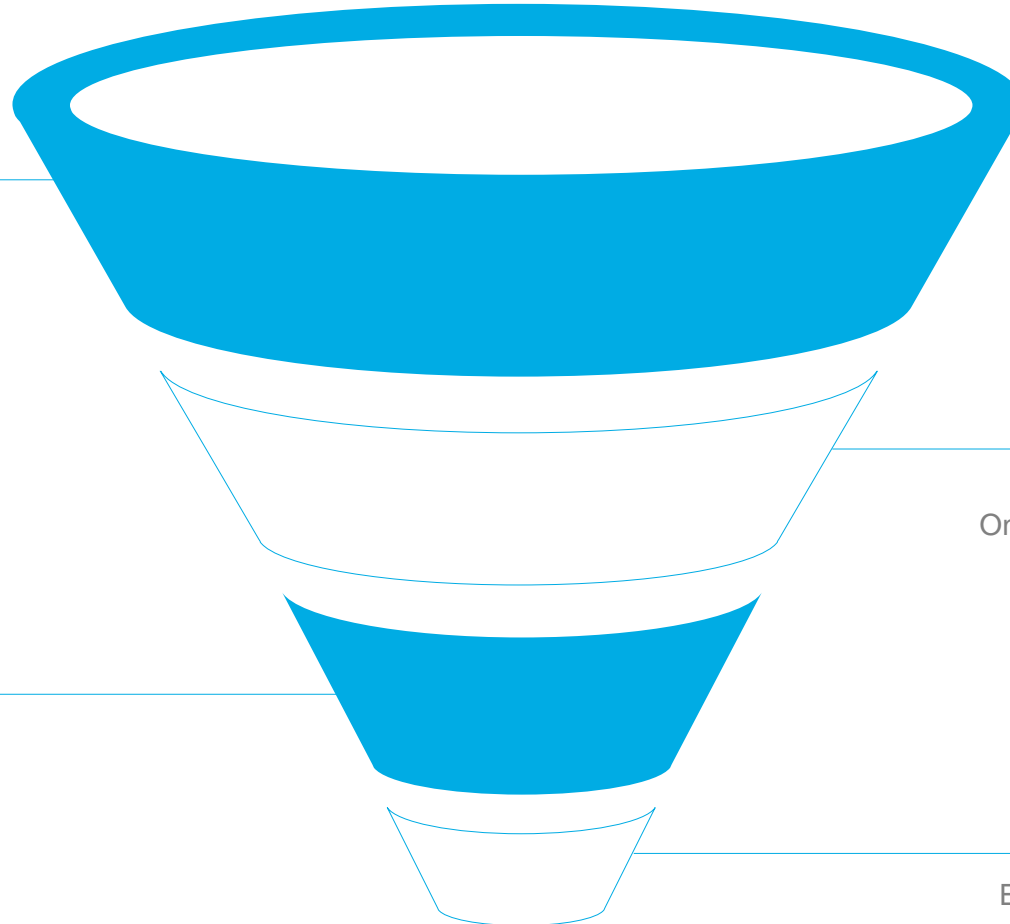Minimize GC footprint

High Throughput

Single Host TPS 20k
One Data Center 1.6MM Ops/second

Low Latency

99.99% latency < 50ms
Average latency < 2ms

Fault Tolerant

Enhance Fault Isolation Mechanism
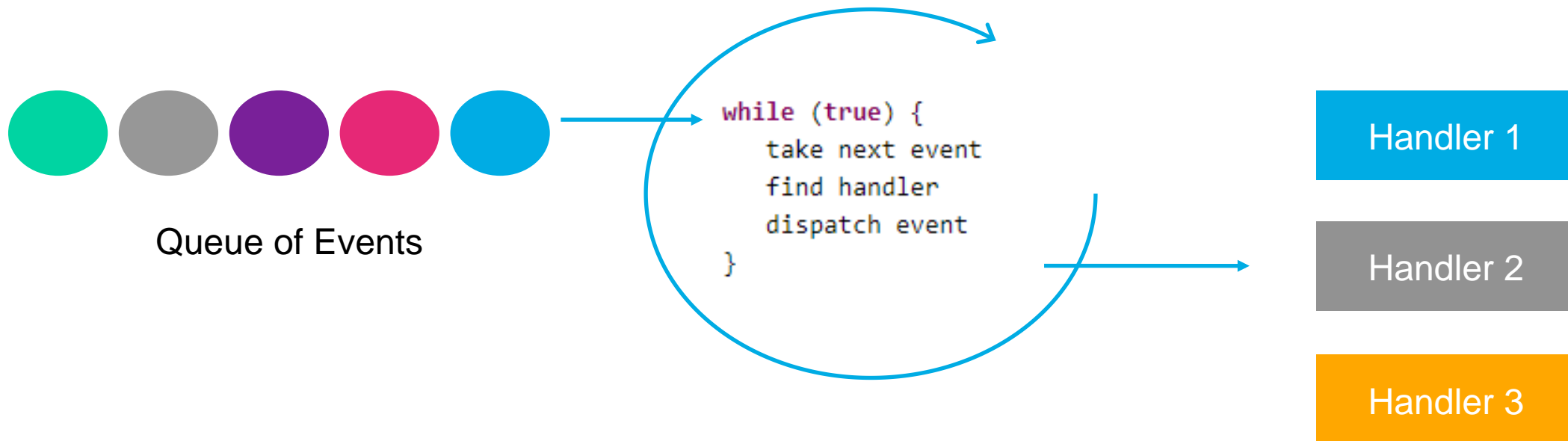Fewer Thread Context Switches

Asynchronous Development Model

```
// synchronous call
int res = compute (1, 2);
```
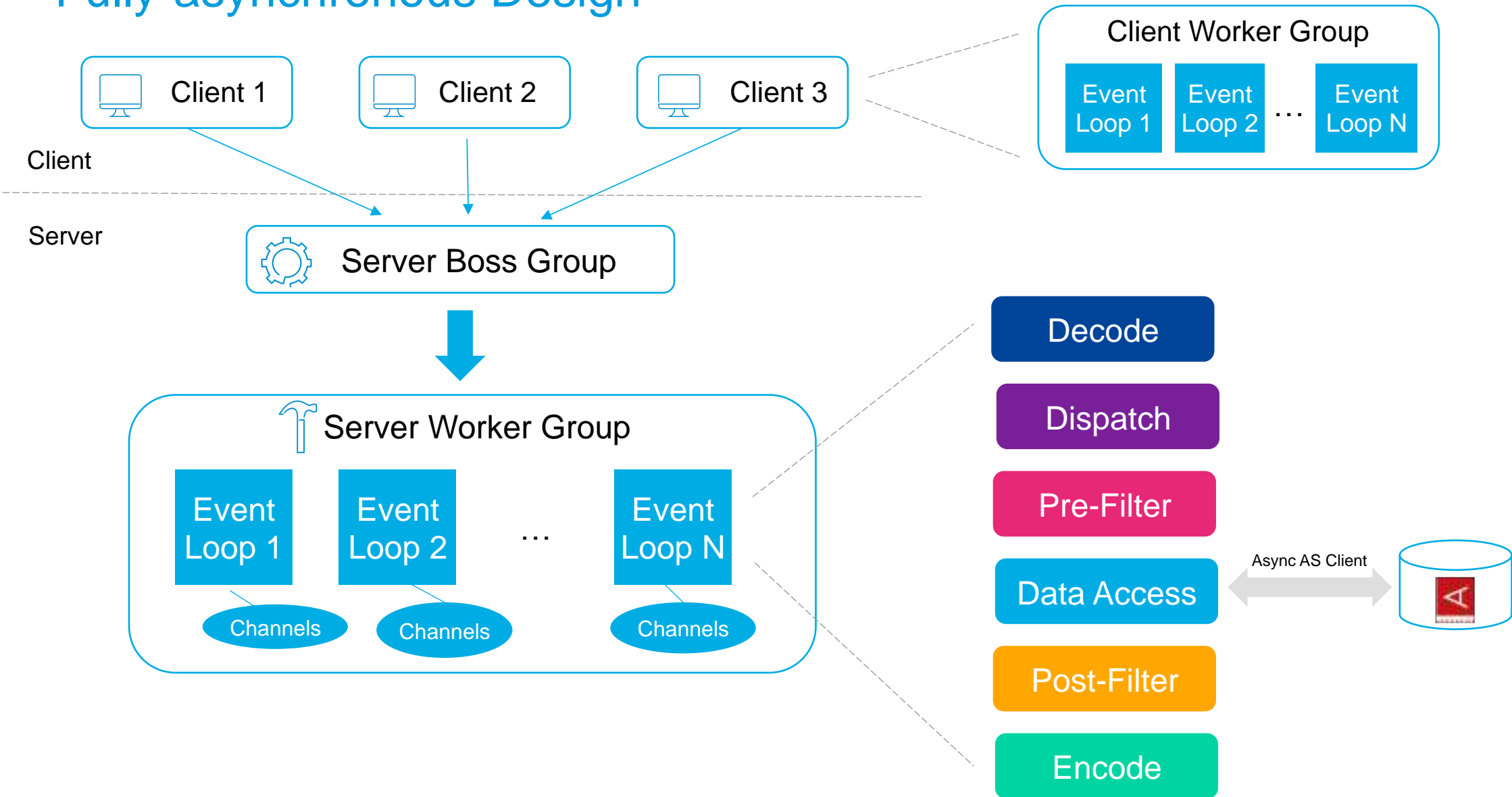
- ✓ Line-by-Line
- ✓ Straightforward

```
//asynchronous call
compute (1, 2, res -> {
    // Called with result
});
```

- ✓ Event-driven
- ✓ Non-blocking

Queue of Events

```
while (true) {
    take next event
    find handler
    dispatch event
}
```

Handler 1

Handler 2

Handler 3

Fully-asynchronous Design

Client

Client 1   Client 2   Client 3

Client Worker Group

| Event Loop 1 | Event Loop 2 | ... | Event Loop N |

Server

Server Boss Group

Server Worker Group

| Event Loop 1 | Event Loop 2 | ... | Event Loop N |

Channels   Channels   Channels

Decode

Dispatch

Pre-Filter

Data Access ←Async AS Client→

Post-Filter

Encode

P **PayPal**
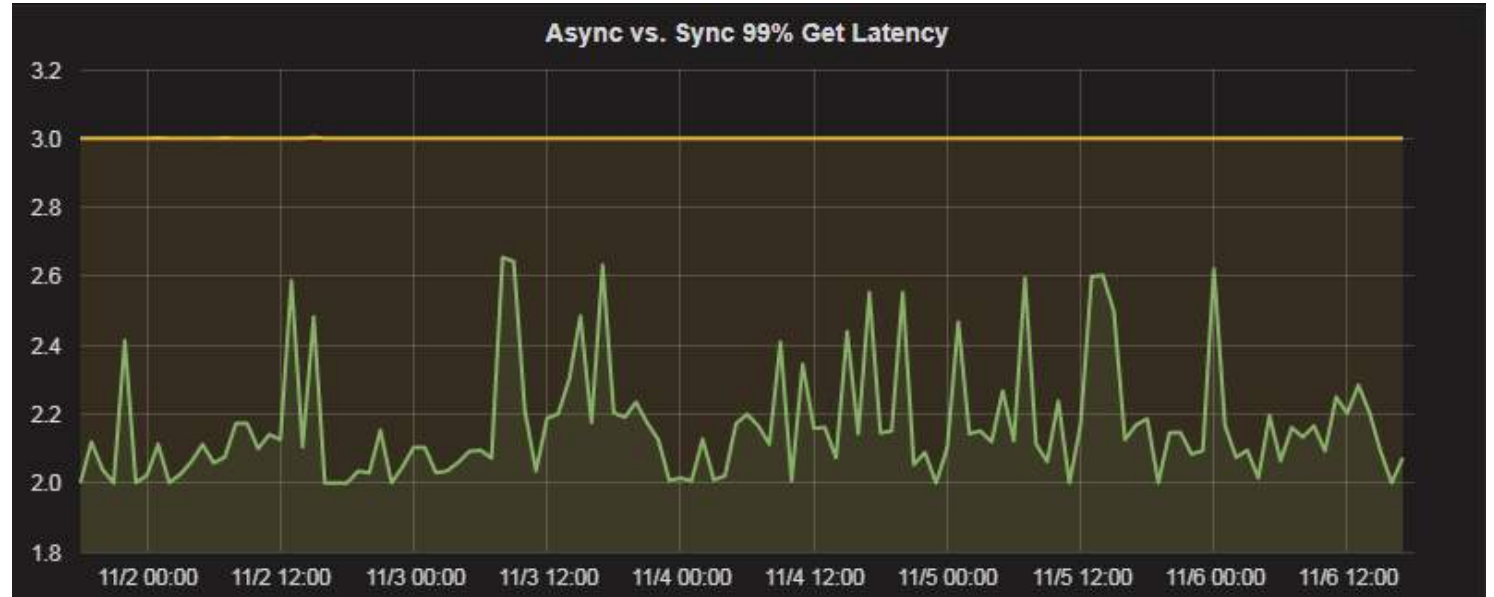
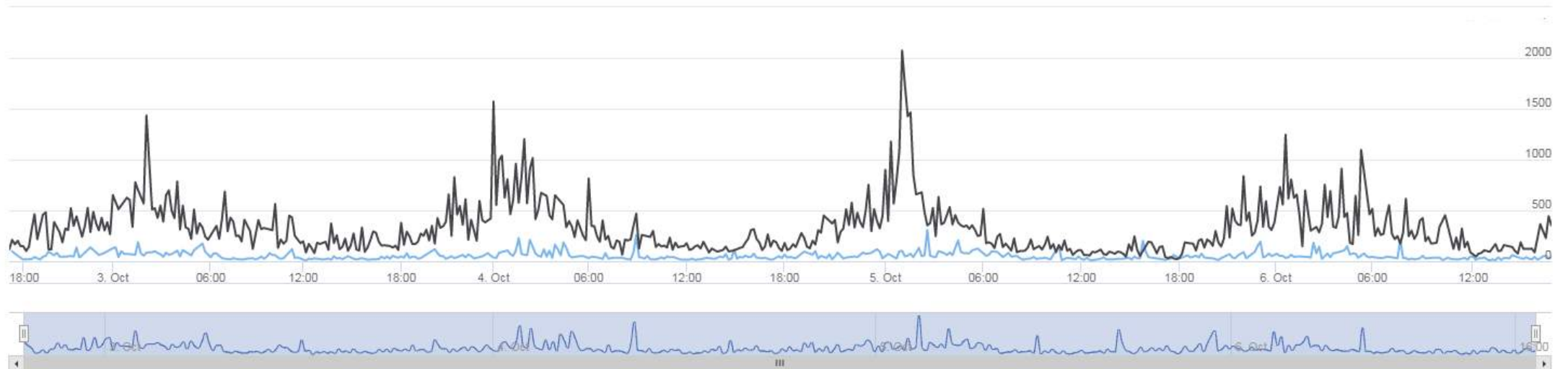# Performance Results (1)

- ## Lower Latency
  - 20% latency improvement

- ## Fewer Failure Count
  - 75-95% failure reduction

Performance Results (2)

✓ Higher Throughput

✓ 4-core VM Testing
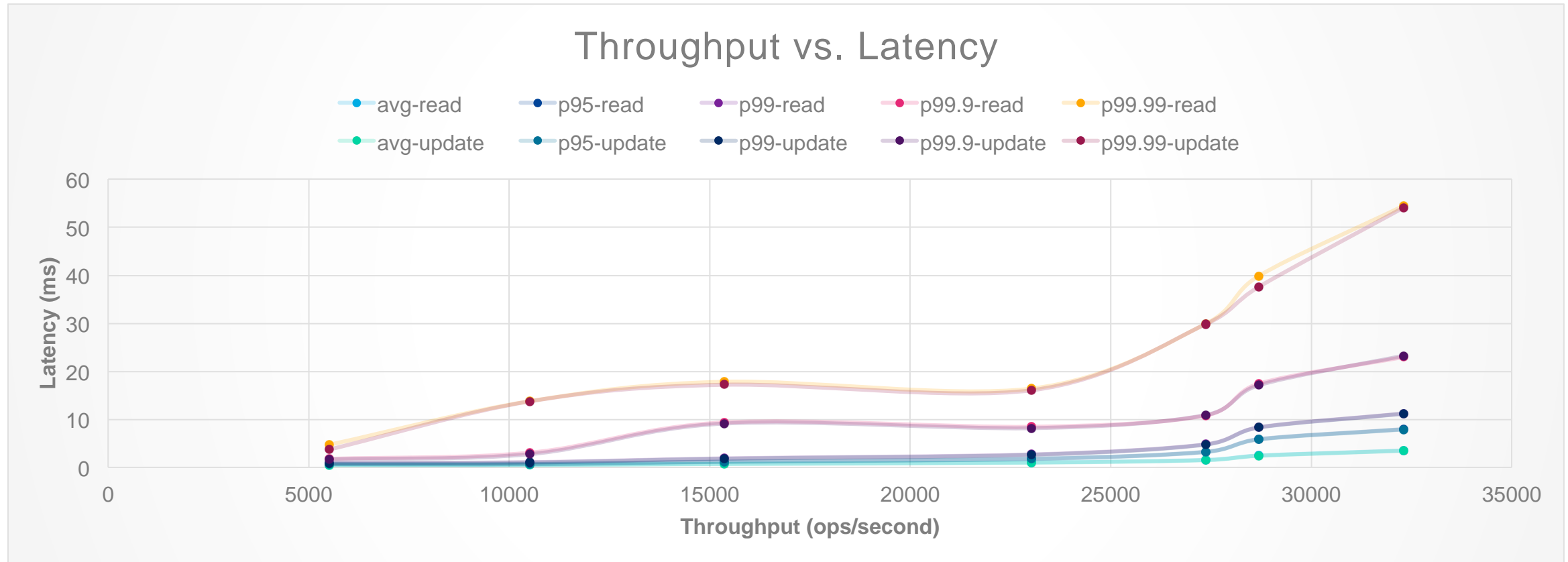
Throughput vs. Latency

Legend: avg-read, p95-read, p99-read, p99.9-read, p99.99-read, avg-update, p95-update, p99-update, p99.9-update, p99.99-update

X-axis: Throughput (ops/second)
Y-axis: Latency (ms)

Agenda

The Road to Achieve High Availability & Low Latency

- Introduction to PayPal Risk Management

- PayPal Risk Data Access Platform

- **Best Practices and Lessons Learnt**

- Future Work

- Q & A

NO Perfect Data Store Product

- Know What to Optimize for
  - Understanding Your Own Use Case Helps

- Reduce Operational Overhead
  - Monitoring Matters When You Scale
  - Define the Contract between Data Access and DB Carefully

- Pay Attention to Efficiency
 - Otherwise It's not worthwhile the Cost

Asynchronous Programming is Hard

- Require Mindset Shift from Synchronous Model
  - Have to Make Whole Stack Asynchronous
  - Require More Coding Discipline to Make Every Callback Correct

- More Complicated Error Handling Mechanism
  - Need More Tooling & Skills to Identify the Bug

- Tricky Memory Management
  - Pooled vs. UnPooled
  - Retain/Release ByteBuf at the Right Time

THERE ARE ONLY 2 HARD THINGS
IN COMPUTER SCIENCE:

0. Cache Invalidation
1. Naming Things
7. Asynchronous Callbacks
2. Off-by-one errors

LnP is Time-consuming

- Don't Expect Success at One Shot
  - Too Many Parameters to Tune

- Continuous Integration is NOT an Option
  - Sign-off after Each Code Commit

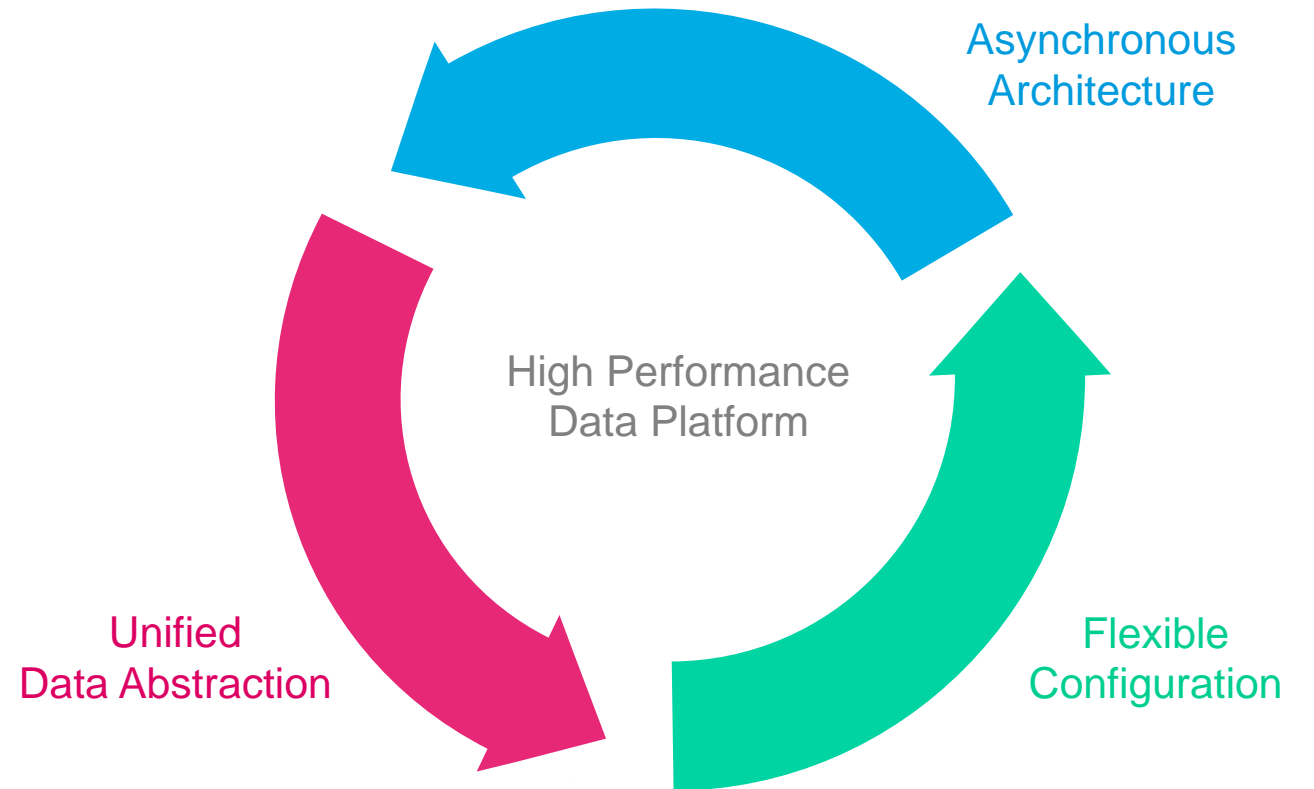- Dependency Matters
  - SSL, Logging, Environment, etc.
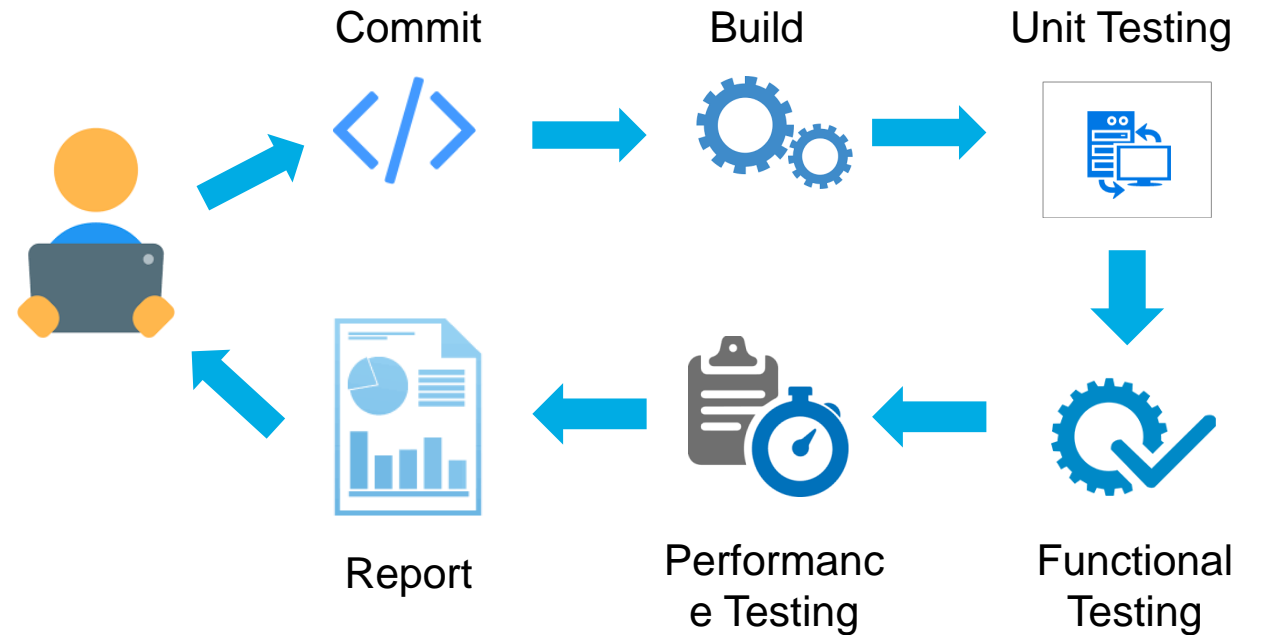
Commit

Build

Unit Testing

Report

Performance Testing

Functional Testing

Agenda

The Road to Achieve High Availability & Low Latency

- Introduction to PayPal Risk Management

- PayPal Risk Data Access Platform

- Best Practices and Lessons Learnt

- Future Work

- Q & A

Future Work

- Data Access Platform Buildout
  - Support More Data Stores
  - Data Abstraction Deep Dive
    - Domain-specific Abstraction
    - Environment Awareness
    - Better Governance
- Continuous Performance Tuning
  - NIO vs. Epoll SocketChannel
  - Proto Buffers vs. Msgpack
- Open Source

Asynchronous Architecture

High Performance Data Platform

Flexible Configuration

Unified Data Abstraction

Recap

- ✓ Risk Management is a Competitive Advantage for PayPal

- ✓ Data Access Abstraction

- ✓ Comprehensive Configuration System

- ✓ Asynchronous and Non-blocking Architecture

Q & A

Thank You!