

Chris Edwards

Zero Downtime Data Migrations



**Agile** ORLANDO  
JULY 24-28 **2023**

**Continuous  
Delivery**

**!=**

**Continuous  
Deployment**

**!=**

**Zero  
Downtime  
Deployment**

# Continuous Delivery



Continuous Delivery is the **ability** to get changes of all types—including new features, configuration changes, bug fixes and experiments—into production, or into the hands of users, safely and quickly in a sustainable way.

# Continuous Deployment



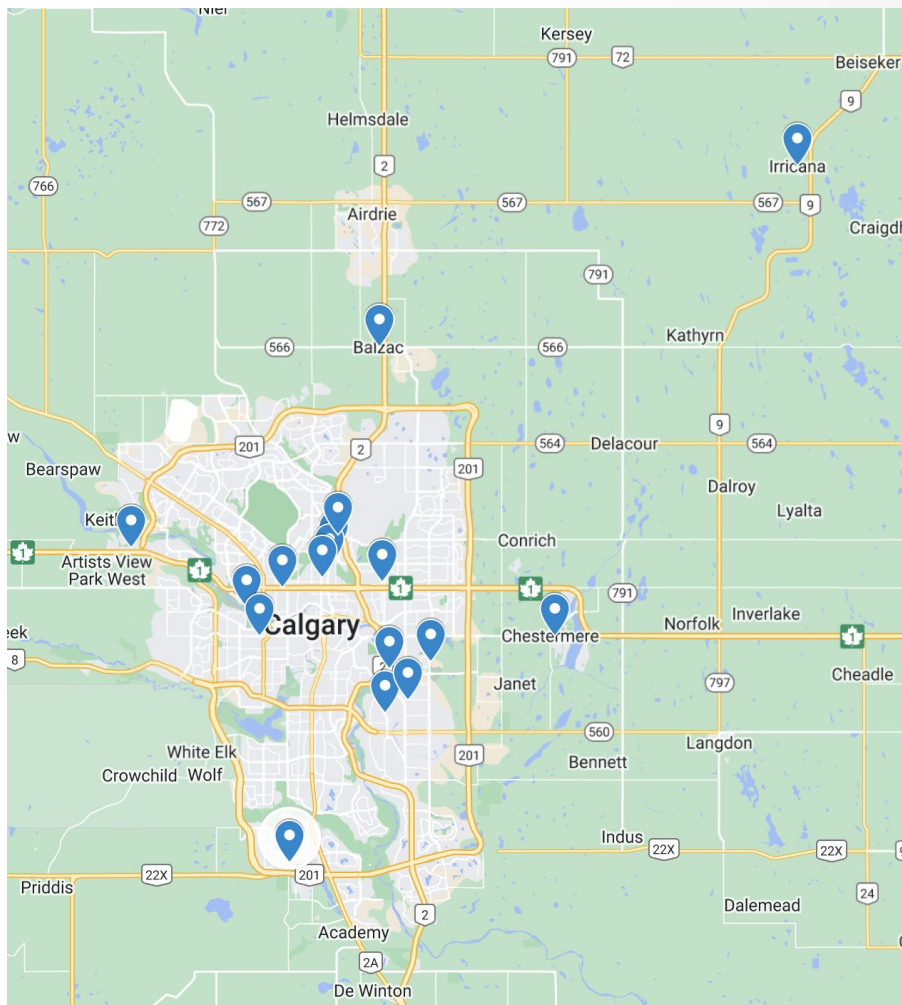
Involves the actual **release** of changes into a production environment using no manual steps

# Zero Downtime Deployment



The **release** of changes to a production environment **without disrupting users** (actual humans or external systems)





**15 Stations in  
and around  
Calgary**

**Stations hours:  
6am–11pm MT**

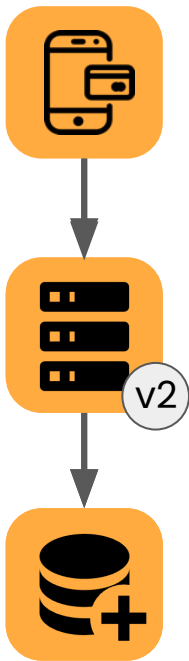


**11pm Release**

**~ Once every two  
weeks**



# Scheduled Maintenance

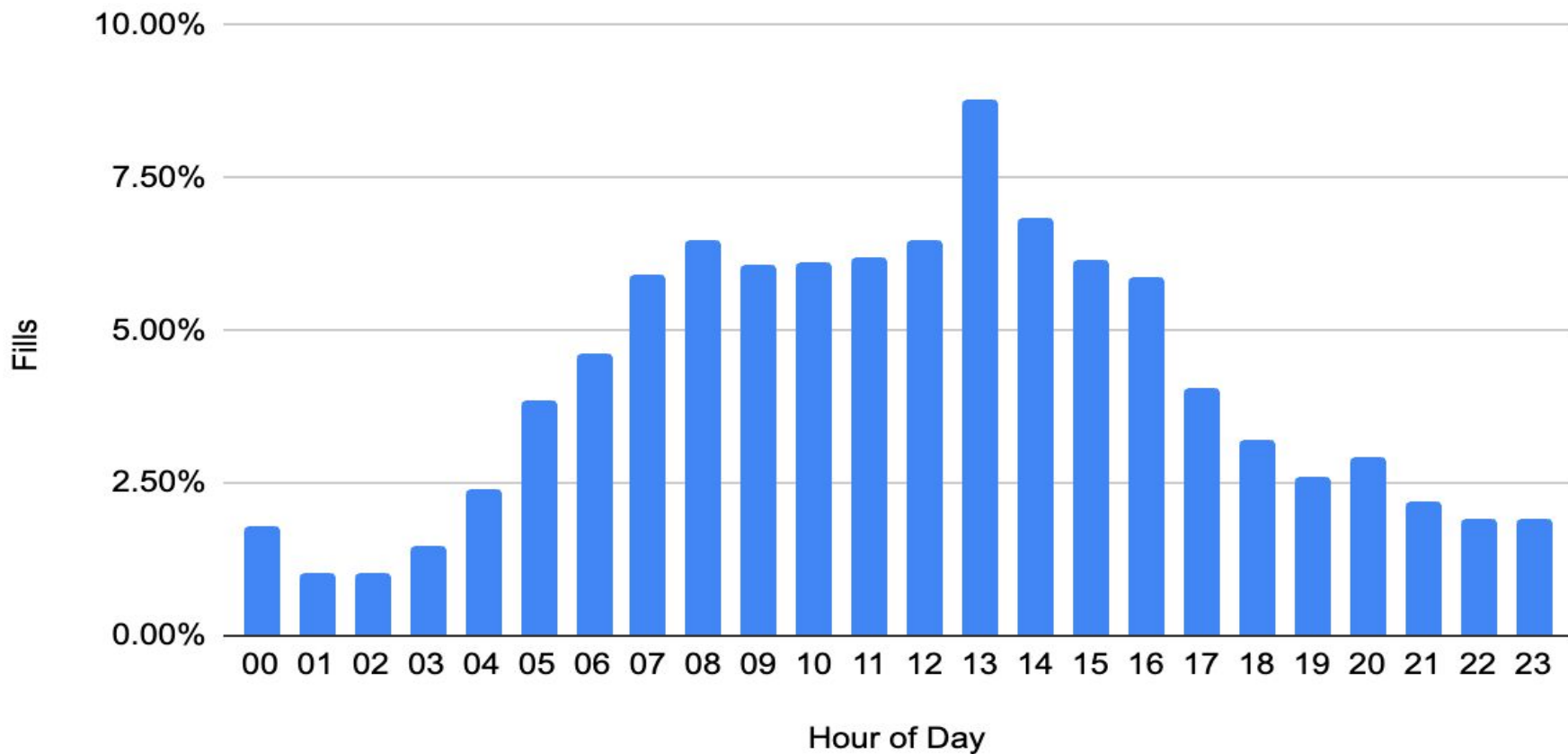


- ✓ Kill server
- ✓ Migrate data
- ✓ Spin up new server

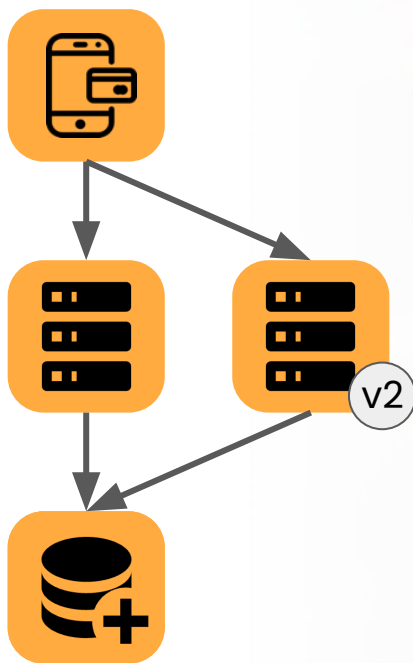
**12,000 stations  
across Canada**



## Fills vs. Hour of Day



# Zero Downtime Deployments



- ✓ Database migrated
- ✓ New server spun up
- ✓ Approx 5 minutes passes
- ✓ Traffic routed to new server
- ✓ Old server receiving no traffic
- ✓ Old server killed

# Two Minutes





**Demo Time!**



<https://demo.chrisedwardsyyc.com>



```
const counterSchema = new Schema({
  counter: {
    currentValue: Number
  }
})
```

```
class Counter {
  id;
  currentValue;

  constructor(id, count) {
    this.id = id;
    this.currentValue = count ?? 0;
  }

  static fromDoc(doc) {
    return new Counter(doc._id, doc.counter?.currentValue)
  }

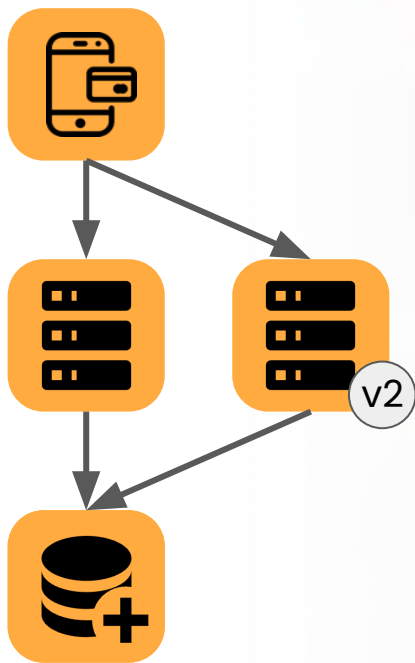
  toDoc() {
    return { counter: { currentValue: this.currentValue } };
  }
}
```



```
async up(db, client) {
  await db.collection('counters').update(
    {}, {
      $rename: {'counter.currentValue': 'counter.newValue'})
  })
```



# Zero Downtime Deployments

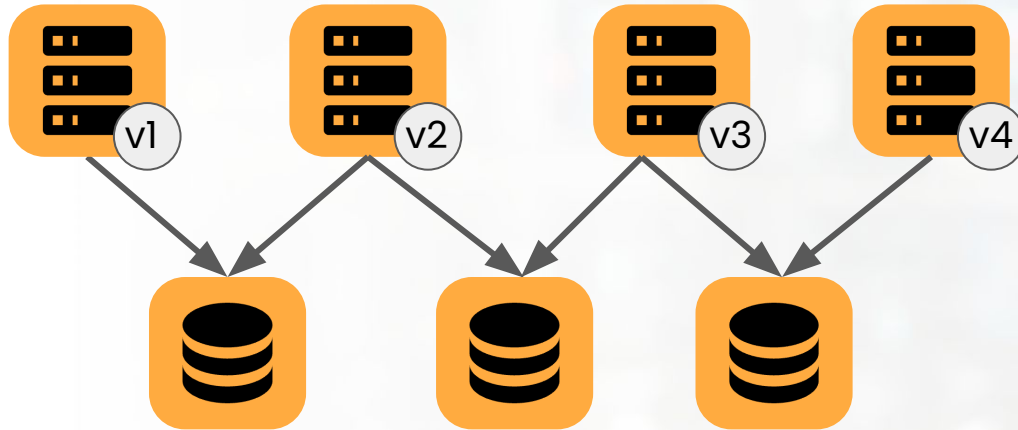


- ✓ Database migrated
- ✓ New server spun up
- ✓ Approx 5 minutes passes
- ✓ Traffic routed to new server
- ✓ Old server receiving no traffic
- ✓ Old server killed

# Multi-Step Migrations



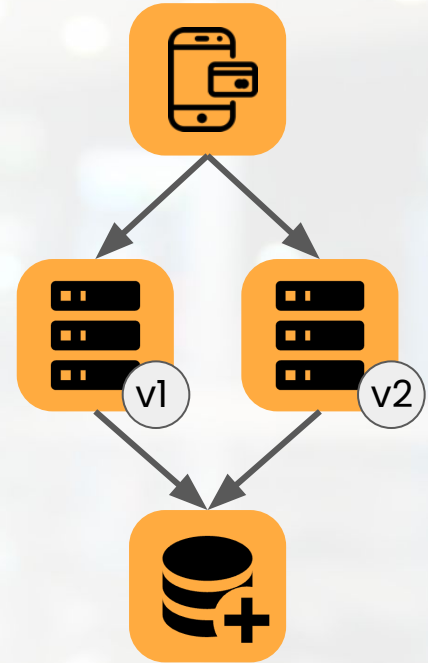
Load Balancer



# Rename - Step 1/3



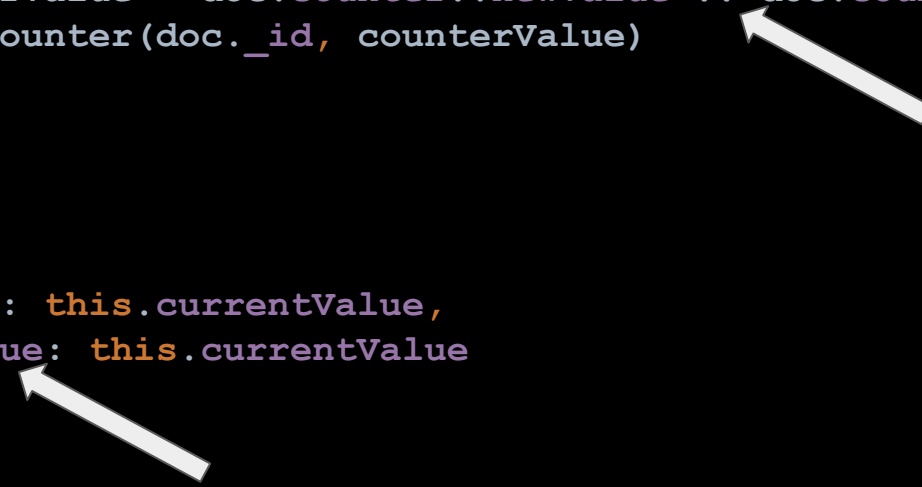
- Variable saved to new name AND old name
- Server can read from new name AND old name



```
const counterSchema = new Schema({
  counter: {
    newValue: Number,
    newerValue: Number
  }
});
```



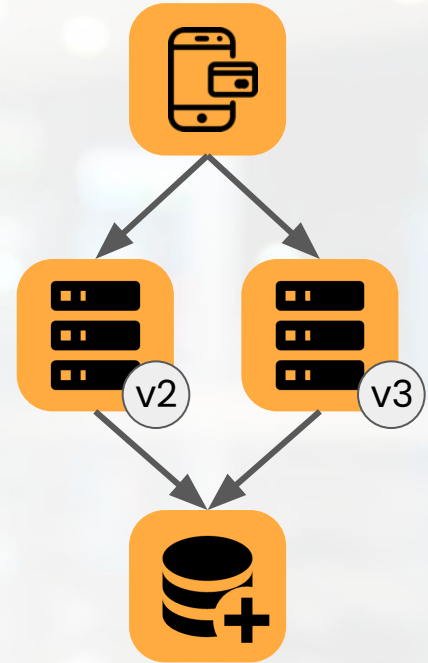
```
static fromDoc(doc) {  
    const counterValue = doc.counter?.newValue ?? doc.counter?.newerValue;  
    return new Counter(doc._id, counterValue)  
}  
  
toDoc() {  
    return {  
        counter: {  
            newValue: this.currentValue,  
            newerValue: this.currentValue  
        }  
    };  
}
```



# Rename - Step 2/3



- Database migration to rename variable
- Server no longer saves to old name
- Server no longer reads from the new name





```
const counterSchema = new Schema({
  counter: {
    newValue: Number
  }
});
```

```
static fromDoc(doc) {
  const counterValue = doc.counter?.newerValue;
  return new Counter(doc._id, counterValue)
}

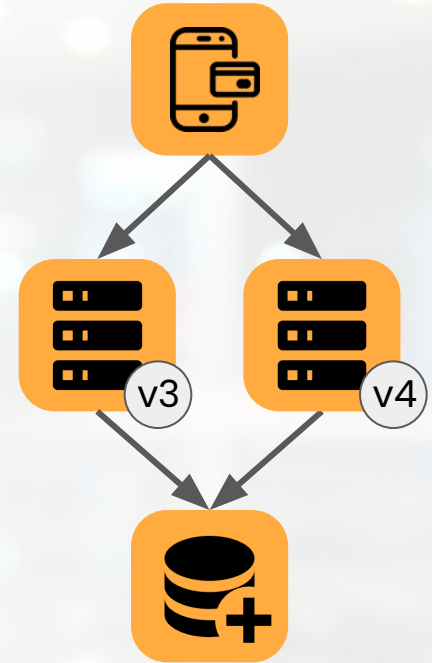
toDoc() {
  return {
    counter: {
      newerValue: this.currentValue
    }
  };
}
```

```
async up(db, client) {  
  await db.collection('counters').update(  
    {}, {  
      $rename: { 'counter.newValue': 'counter.newerValue' }})  
  })
```

# Rename - Step 3/3



- Migration to remove old schema



```
async up(db, client) {
  await db.collection('counters').update(
    {},
    [{ $unset: 'counter.newValue' }])
}
}
```

# Migration Assessment



- Can data loss occur?
- Is there customer impact of data loss?
- Can it be easily repaired?

# Is Zero-Downtime right for you?



- How many users do you have? Zero? 100? 10,000?
- How often do your users use your system?
- How critical is this system to their business (or yours?)
- What impact is the current deployment system having on your development team?

A solid teal vertical bar is positioned on the left side of the slide, extending from the top to the middle section.

# How we work

A solid orange circle is located in the bottom left corner of the slide.



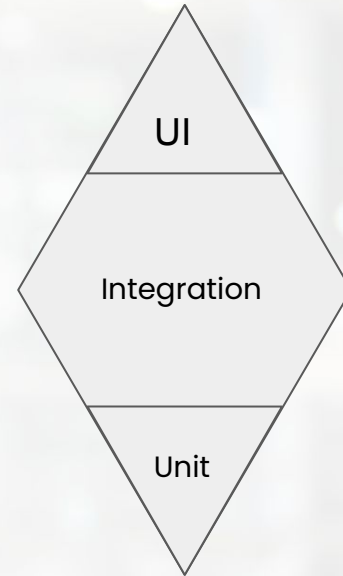
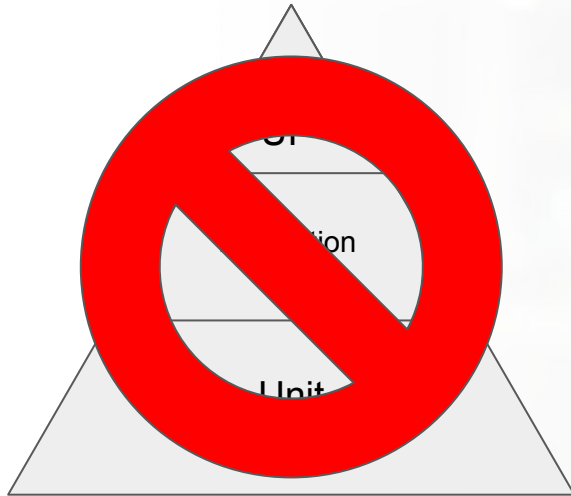
# Test-Driven Development

---



- Red-green refactor
- Nearly 100% test server-side coverage

# Test Pyramid Diamond



Test Pyramid - Mike Cohn, in his 2009 book Succeeding with Agile

# Data Pipeline Tests







A vertical teal bar is positioned on the left side of the slide, extending from the top to the middle of the page.A solid orange circle is located in the bottom left corner of the slide.

# **“Continuous” Deployment**

A vertical teal bar is positioned on the left side of the slide, extending from the top to the middle of the text area.

**You build it,  
You run it**

A solid orange circle is located in the bottom left corner of the slide.

**Questions?**







PRESENTS

# Agile ORLANDO 2023

JULY 24-28

**Join Agile Alliance today!**

Become an Agile Alliance member and help support our non-profit mission, while gaining access to valuable benefits like online events, in-person conference discounts, and event session videos.



#AGILE2023