CFML Summit 2015
Brian Ghidinelli

I'm at "Building Multi-tenant SaaS Apps" w/ @ghidinelli #CFSummit2015

Brian Ghidinelli

Entrepreneur, traveler, racecar driver

founder
MotorsportReg
RaceHero

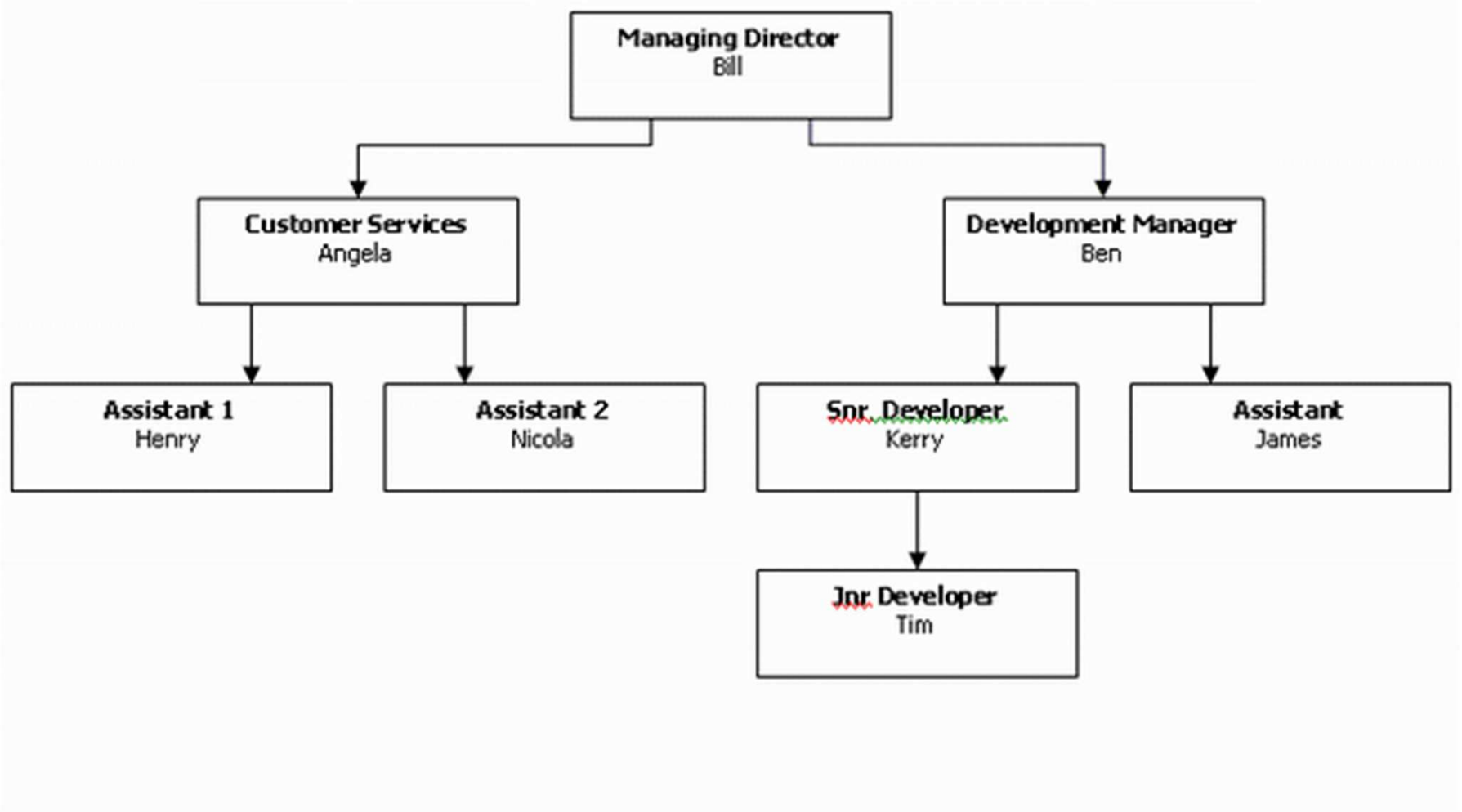
contact
fb.com/BrianGhidinelli
ghidinelli.com

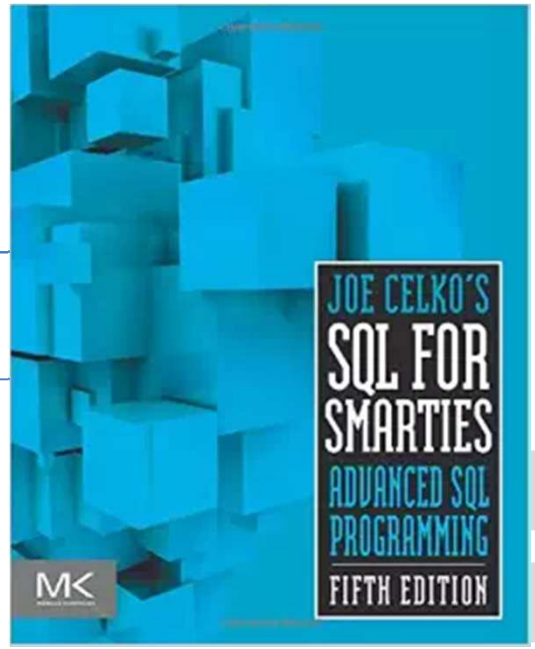
please tweet
@ghidinelli
#CFSummit2015

HIRING: Small-team Developer/VP/CTO

CF/JS with CI/CD/testing experience


bwin


Managing Director
Bill
Customer Services
Angela
Development Manager
Ben
Assistant 1
Henry
Assistant 2
Nicola
Snr. Developer
Kerry
Assistant
James
Jnr Developer
Tim



Classes

Natl 1, Natl 2, …

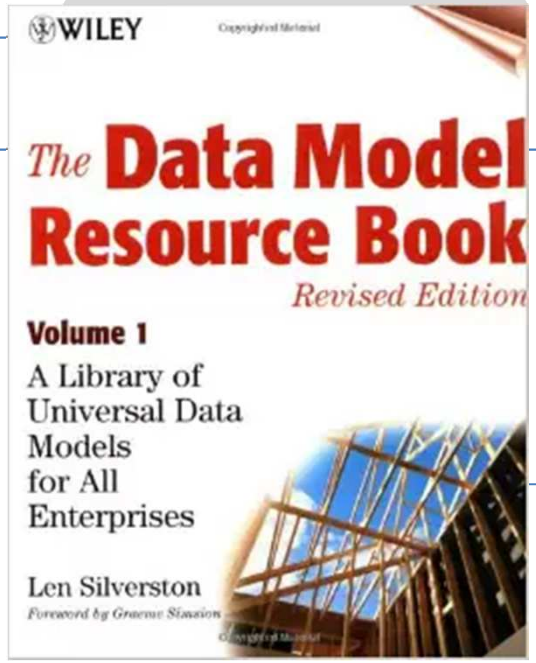
Natl 1, Natl 2 +
Local A, Local B, …

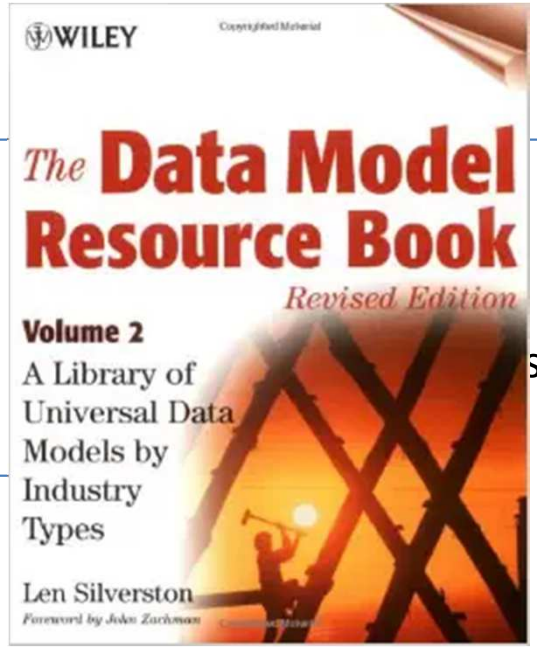
Participants

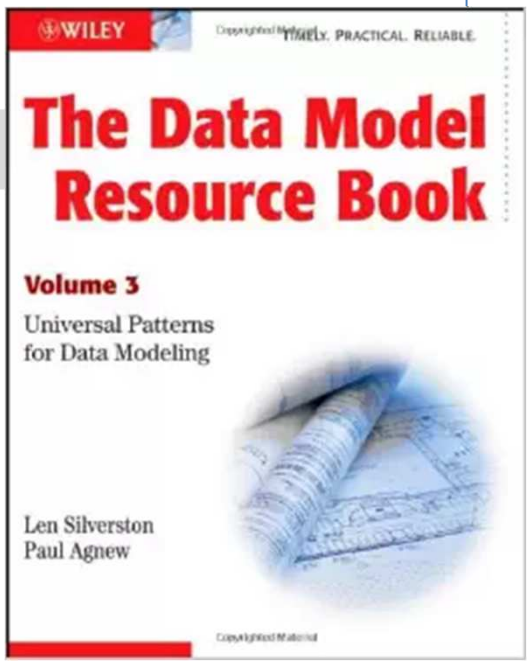
Joe, Mary, Frank, …

Joe

Frank







g11n = i18n + l10n

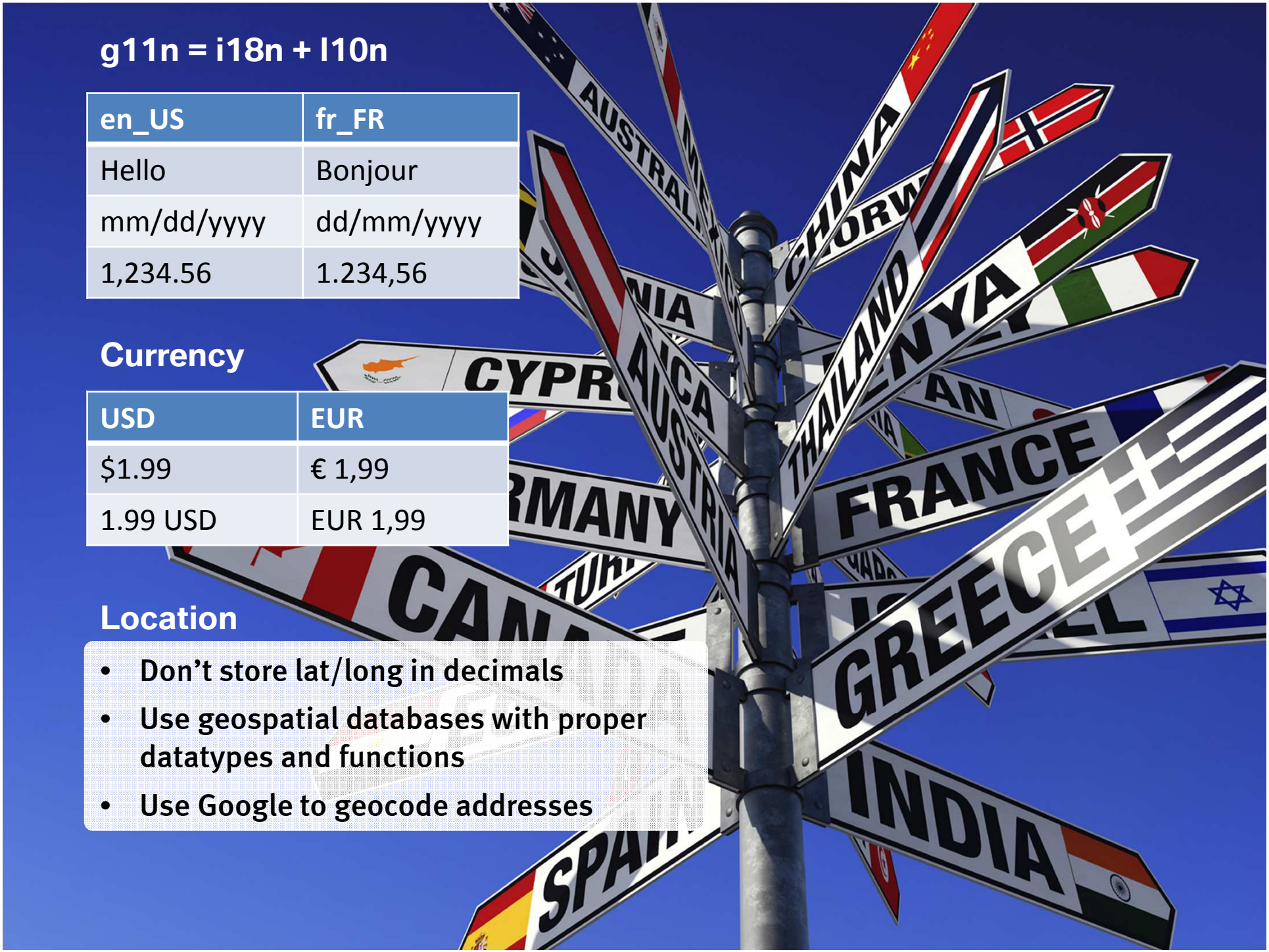
en_US	fr_FR
Hello	Bonjour
mm/dd/yyyy	dd/mm/yyyy
1,234.56	1.234,56

Currency

USD	EUR
$1.99	€ 1,99
1.99 USD	EUR 1,99

Location

- Don’t store lat/long in decimals
- Use geospatial databases with proper datatypes and functions
- Use Google to geocode addresses



```
# ResourceBundle
appointments = {
    en_US: { greeting: “Hello” },
    fr_FR: { greeting: “Bonjour” },
    ja_JP: { greeting: “もしもし” }
  };

# View
locale = (user selected, accept header, etc…)

#appointments[locale].greeting#!
```

“g11n is easy to add exactly once: at the beginning” @ghidinelli #CFSummit2015

```
# translation is not just for language
<cfscript>
  sports={};
  sports.en.cars.vehicles = “cars";
  sports.en.motorcycles.vehicles = “bikes”;
  sports.es.cars.vehicles = “coche”;
  sports.es.motorcycles.vehicles = “motos”;
</cfscript>

# View – language-specific terminology
vertical = (tenant, context, …)

#sports[locale][vertical].vehicles#
```

Track transaction *attempts* in database

```
insert(payid, tenantid, “pending”);
purchase(payid, amount);
if (purchase.success)
  update(payid, “success”)
else
  update(payid, “decline”)
```

“Dealing w/ PCI-DSS is horrible. Minimize it at all costs. Outsource to @braintree, @stripe, etc” @ghidinelli #CFSummit2015

Impact of Int'l Processing

- Cross-border fees to cardholders
- Foreign exchange fees to you
- Settling in foreign currencies requires local bank accounts or exotic multi-currency accounts
- Tax liabilities

Marketplace Tips

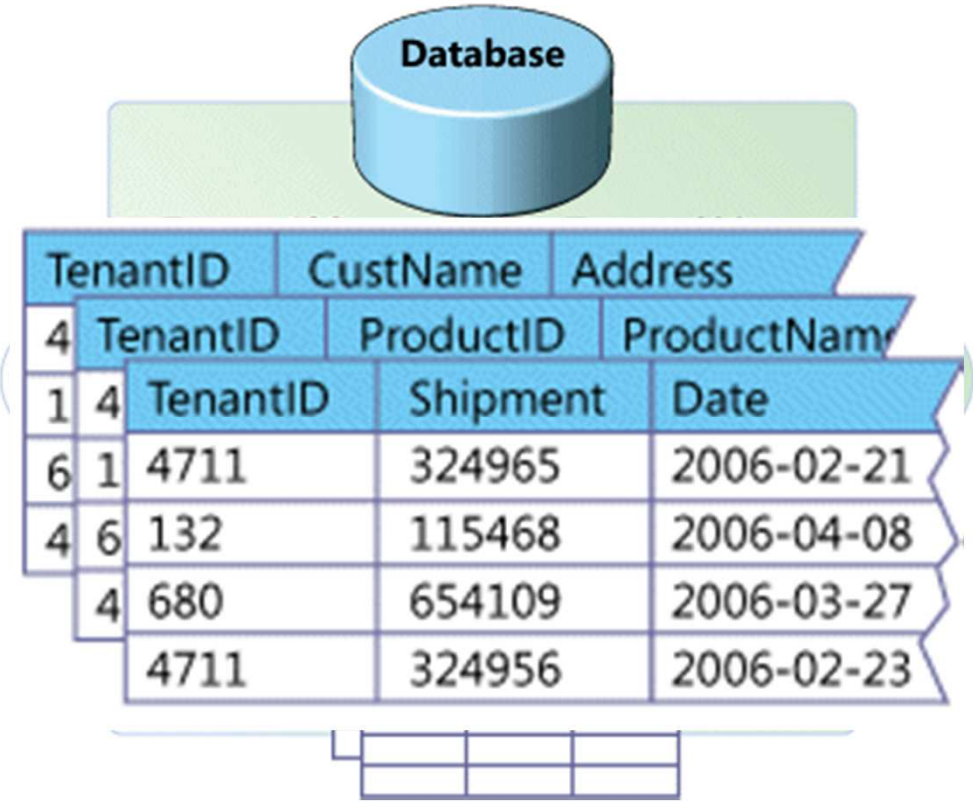
- Can each party choose their currency?
- Avoid FX roundtrip if you disburse in local currency
- New Stripe Marketplace is awesome

Marketplace Gotchas

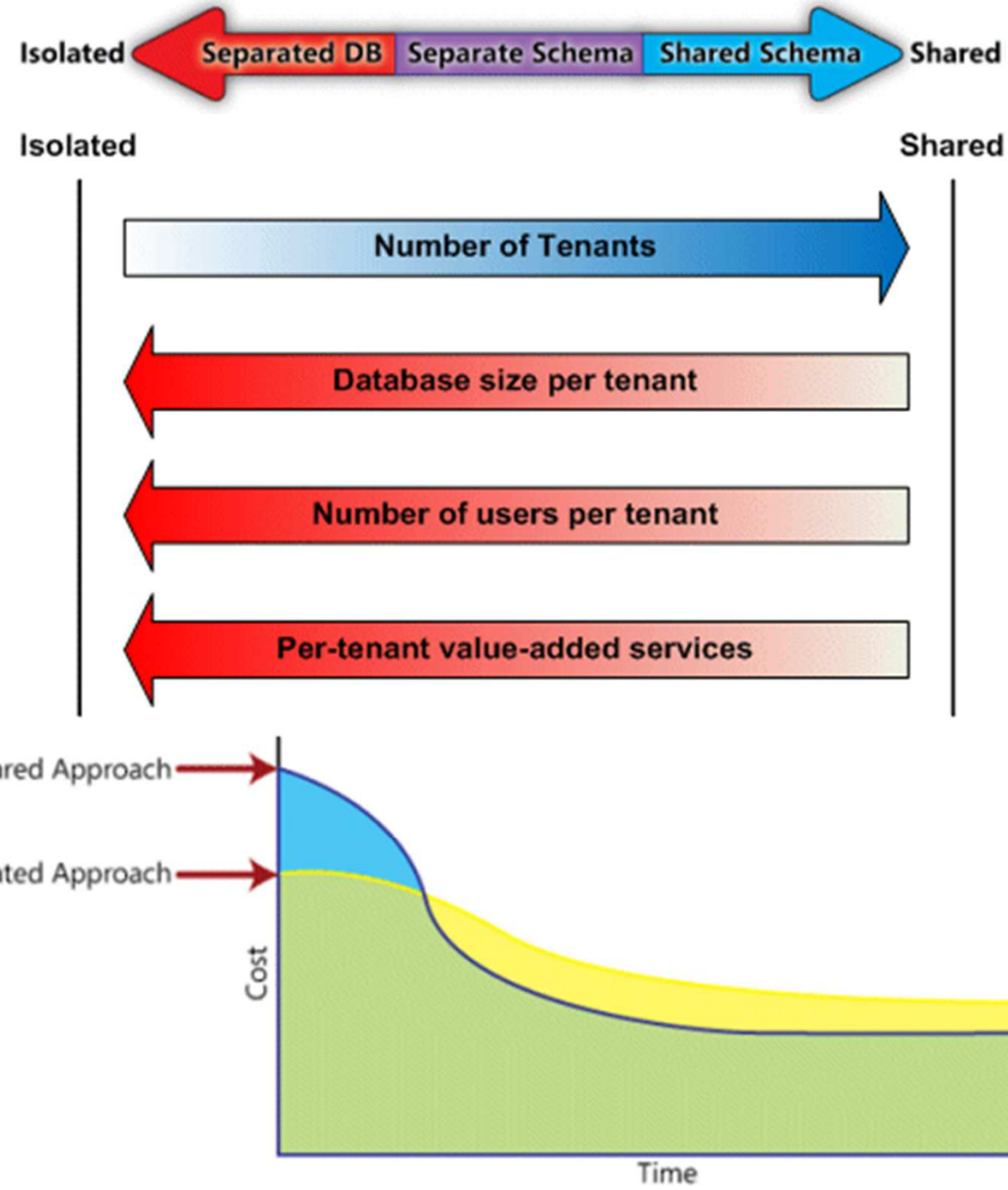
- Opening foreign bank accounts post-9/11
- Know Your Customer (KYC)
- IRS 1099K


PART 2: DATA STUFF


Isolated
Shared


Database
TenantID
CustName
Address
4
TenantID
ProductID
ProductName
1
4
TenantID
Shipment
Date
6
1
4711
324965
2006-02-21
4
6
132
115468
2006-04-08
4
680
654109
2006-03-27
4711
324956
2006-02-23


Isolated
Separated DB
Separate Schema
Shared Schema
Shared


Isolated
Separated DB
Separate Schema
Shared Schema
Shared
Isolated
Shared
Number of Tenants
Database size per tenant
Number of users per tenant
Per-tenant value-added services
Shared Approach
Isolated Approach
Cost
Time

“shared db gets easier over time; scaling isolated db gets hard” @ghidinelli #CFSummit2015

Physical Database Tips

XXXXXXXX-XXXX-XXXX-XXXXXXXXXXXXXXXX

~~Use UUIDs for keys~~

XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX

Use GUIDs for keys

All Entities

created_at, updated_at, deleted_at


WENGER
OFFICIAL
WORLD SCOUT KNIFE

Three Strategies To Mitigate Feature Creep

1. Say No.
2. Solve it via your API, not in your App
   - Expose building blocks and attack it there
3. Feature Flags
   - Great for per-tenant functionality
   - Also for soft or “dark” launching features

			cookie Uses cookies to apply only to your session.	database Database backed, applies to all users.	declaration The default status declared with the feature.	Default The system default when no strategies match.
`shiny_things`	Shiny things.	Off	Switch On Switch Off	Switch On Switch Off		Off
`world_domination`	World domination.	On	Switch On Switch Off	Switch On Switch Off	On	Off
`flakey`	Flakey.	On	Switch On Switch Off	Switch On Switch Off	Off	Off
`something`	Ability to purchase enrollments in courses	On	Switch On Switch Off	Switch On Switch Off	On	Off

```
# feature flag sql
CREATE TABLE tenants (tenant_id)
CREATE TABLE features (feature_name)
CREATE TABLE tenant_feature (tenant_id,
feature_name)

# code
if (tenant.hasFeature('coolFeature'))
{
  // do cool stuff
}
```

"How to avoid feature creep? Say no, solve using API, use feature flags." @ghidinelli

Tenant-Specific Data

- If you know what will be created
  - Regular o2m, m2o, m2m
  - Single table, Class table, Concrete table inheritance (Fowler)
- For when you don't know what will be created
  - Sparse Columns
  - Entity-Attribute-Value (EAV) / Key-Value
  - JSON / XML / NoSQL

```
CREATE TABLE articles (
   ArticleId uuid NOT NULL
   ,Name varchar NOT NULL
   ,Description varchar NOT NULL
   ,Tenant1Length float NULL
   ,Tenant2FavoriteColor varchar NULL
   ,Tenant3AirSpeed int NULL
   ,Tenant3SwallowVelocity int NULL
);
```

TenantID *Integer*	FirstName *String*	BirthDate *Date*	Custom1 *Integer*	Custom2 *String*	Custom3 *Untyped*
345	Ted	1970-07-02	null	Paid	null
777	Kay	1956-09-25	23	null	null
784	Mary	1962-12-21	null	null	null
345	Ned	1940-03-08	null	Paid	null
438	Pat	1952-11-04	null	San Francisco	Yes

EAV

Entity

Products
Category
Customer
Orders

Attribute

Name
Price

Value

Simple Values
Linked to attributes

Entity

ID
0001
0002
0003
...
1001
1002
1003
1004

Value

ID	EntID	AttrID	Value
1	0001	5	SQL
2	0001	6	300
3	0002	5	SQL
4	0002	6	400
5	0003	5	HTML
6	0003	6	200
7	1001	7	0001
8	1001	7	0003
9	1001	1	Rupert
10	1002	1	Hanz

Attribute

ID	Description
1	Name
2	Age
3	Email
4	Phone
5	Skillname
6	Skilllevel
7	Skill
8	...
9	
10	
11	

"EAV/NoSQL is a necessary evil in user-defined data models for SaaS apps" @ghidinelli

```
CREATE TABLE value (
        an
        en
        at                                    LL,
        boo
        dt
        vc                                    ying,
        in
        mn                                    ,
        tx
);
```

TenantID *Integer*	FirstName *String*	BirthDate *Date*	RecordID *Integer*
345	Ted	1970-07-02	893
777	Kay	1956-09-25	null
784	Mary	1962-12-21	564
345	Ned	1940-03-08	117
438	Pat	1952-11-04	301

1 ∞

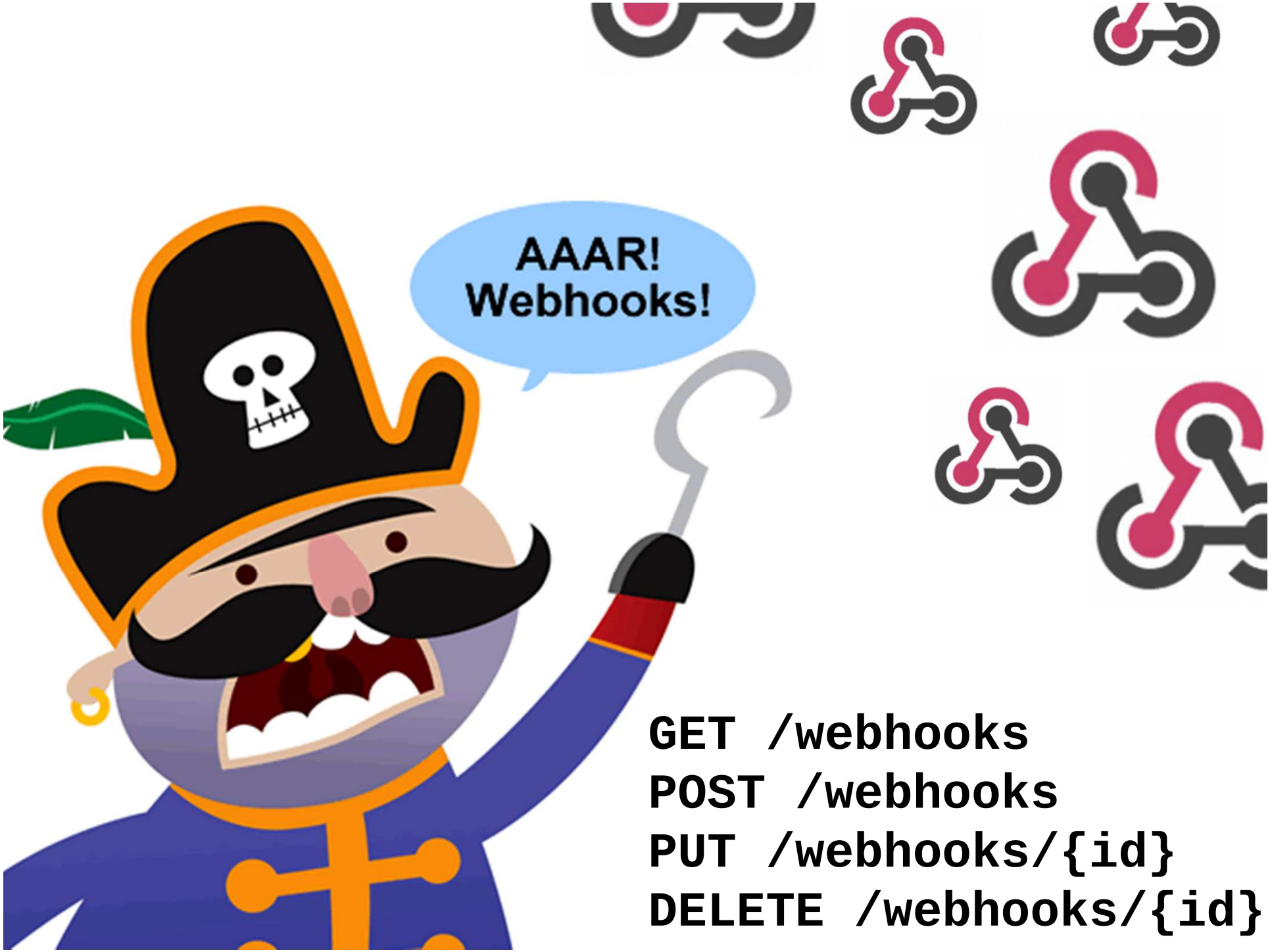
RecordID	Name	Value
893	Subscriber	Yes
null	Status	Gold
564	Expire	2008-07-29
117	Subscriber	No
301	Affiliation	Acme

JSON/XML/NoSQL

- Document-oriented DB like MongoDB
  - Big departure from relational model
- XML/JSON/HSTORE in database
  - Adds unstructured data directly to relational entity
  - Best of both worlds?

```
CREATE TABLE vehicle (
  vehicleid uuid NOT NULL,
  year int,
  make varchar,
  model varchar,
  color varchar,
  metadata jsonb = {
                     tenant1hp: ‘’,
                     tenant1torque: ‘’,
                     tenant2wheelbase: ‘’,
                     tenant3vin: ‘’,
                     tenant3logbook: ‘’
                   }
);
```

```
SELECT vehicleid
   ,year
   ,make
   ,model
   ,metadata ->> ‘tenant1hp’ AS hp
   ,metadata ->> ‘tenant1torque’ AS torque
   ,metadata ->> ‘noexist’ AS val
FROM vehicle
WHERE metadata ->> ‘tenant1torque’ > 250
```



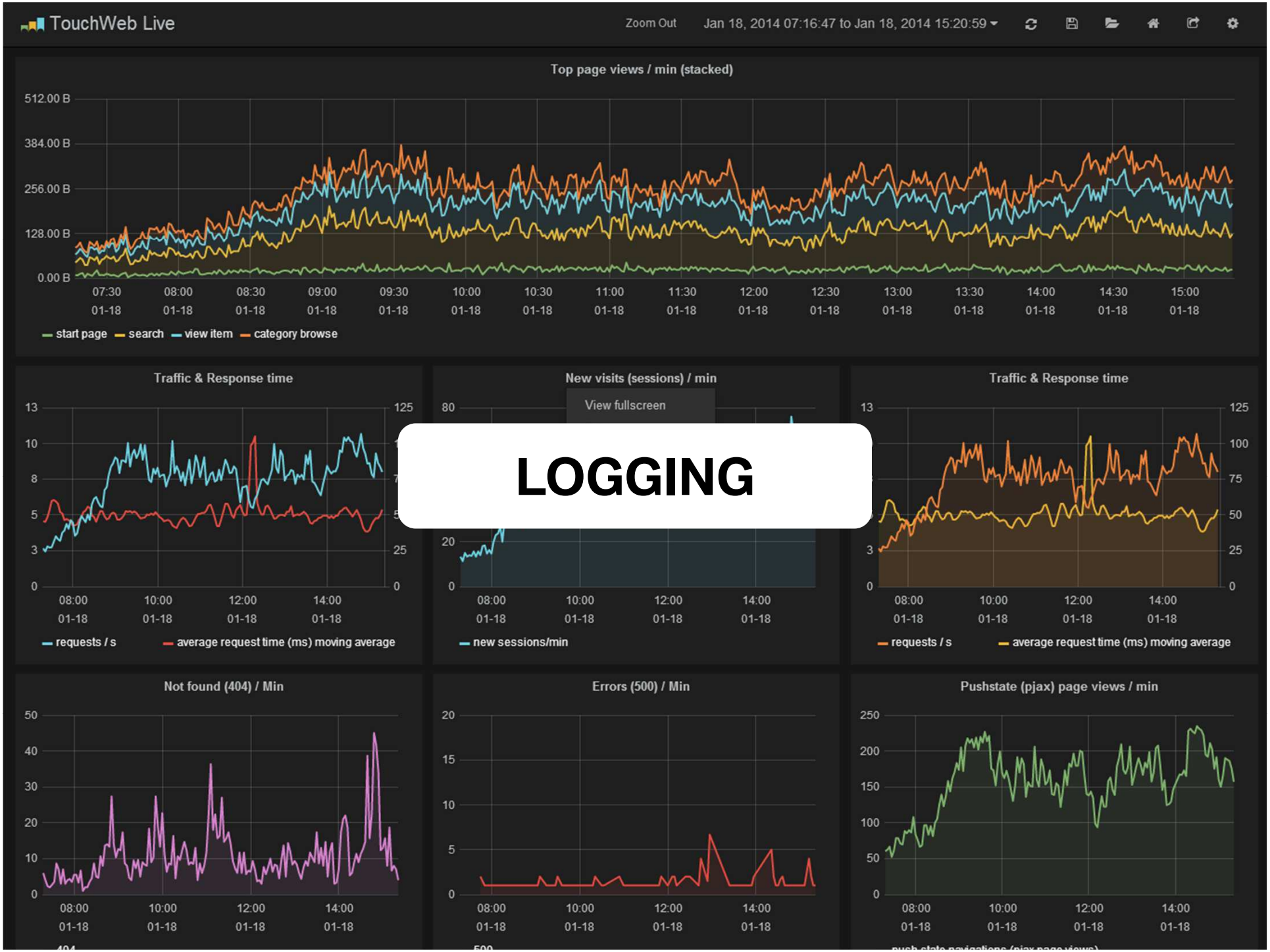

```
GET /webhooks
POST /webhooks
PUT /webhooks/{id}
DELETE /webhooks/{id}
```

PART 3: INFRASTRUCTURE

The Twelve-Factor App

- One code base, many deploys
  - CI, CD, “git push”
- Explicitly declare & isolate dependencies
  - Bundler (Ruby), Bower (JS), CommandBox (CF)
- Parity between dev, staging & production
  - Docker
- Store config in the environment
  - No config in your app repo!

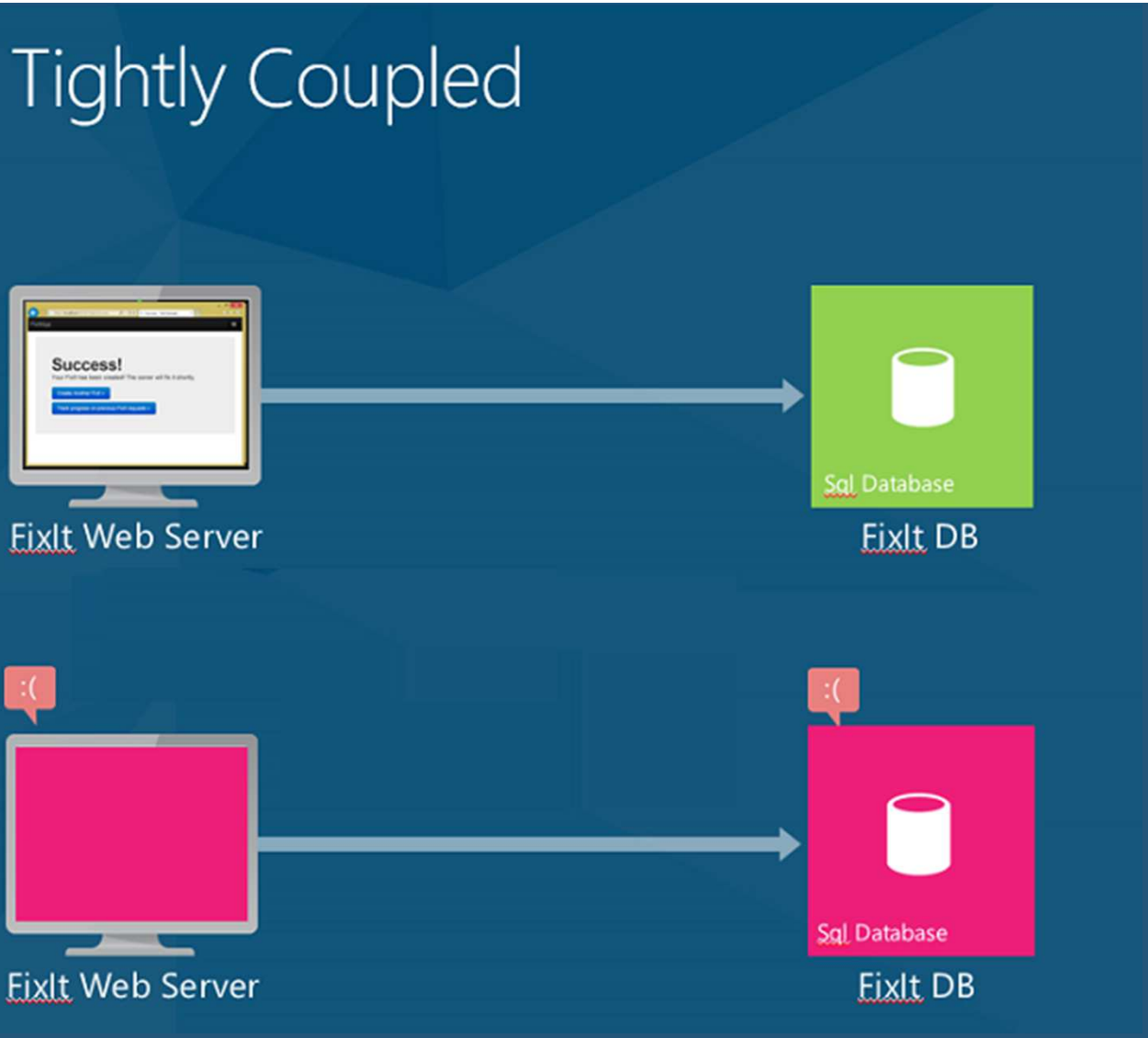
```
# cfstart.bat
SET TESTMODE=1
SET CUSTOM_TAGS_DIR=c:/cf10/tags
coldfusion.exe -start –console

# application.cfc
Local.env = createObject("java",
"java.lang.System")
this.customtagpaths =
env.getEnv("CUSTOM_TAGS_DIR") />
```

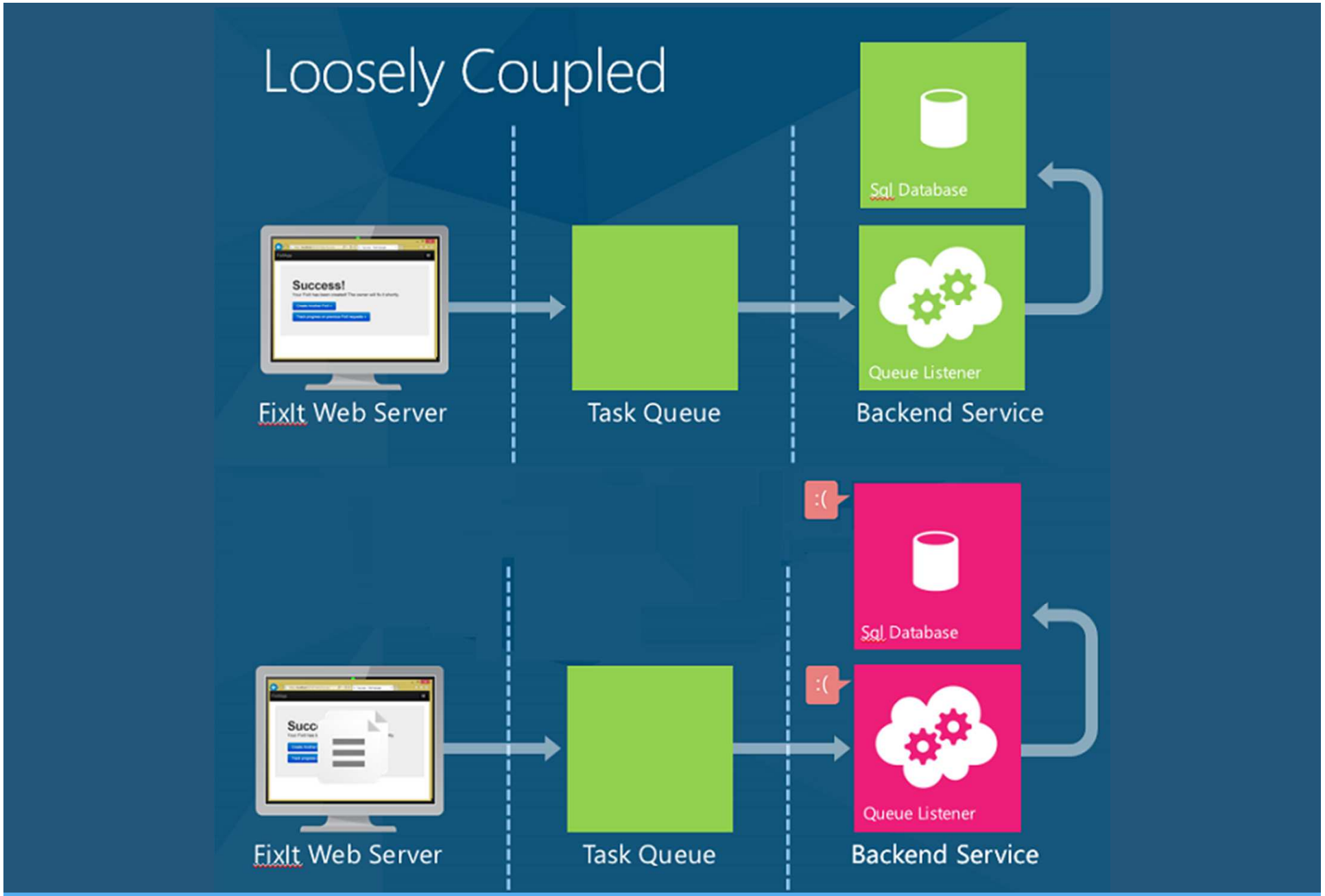

TouchWeb Live
Zoom Out
Jan 18, 2014 07:16:47 to Jan 18, 2014 15:20:59
Top page views / min (stacked)
512.00 B
384.00 B
256.00 B
128.00 B
0.00 B
07:30 01-18
08:00 01-18
08:30 01-18
09:00 01-18
09:30 01-18
10:00 01-18
10:30 01-18
11:00 01-18
11:30 01-18
12:00 01-18
12:30 01-18
13:00 01-18
13:30 01-18
14:00 01-18
14:30 01-18
15:00 01-18
start page
search
view item
category browse
Traffic & Response time
requests / s
average request time (ms) moving average
New visits (sessions) / min
View fullscreen
new sessions/min
Traffic & Response time
requests / s
average request time (ms) moving average
Not found (404) / Min
Errors (500) / Min
Pushstate (pjax) page views / min

LOGGING


“Using sticky sessions was a huge mistake” @ghidinelli #CFSummit2015

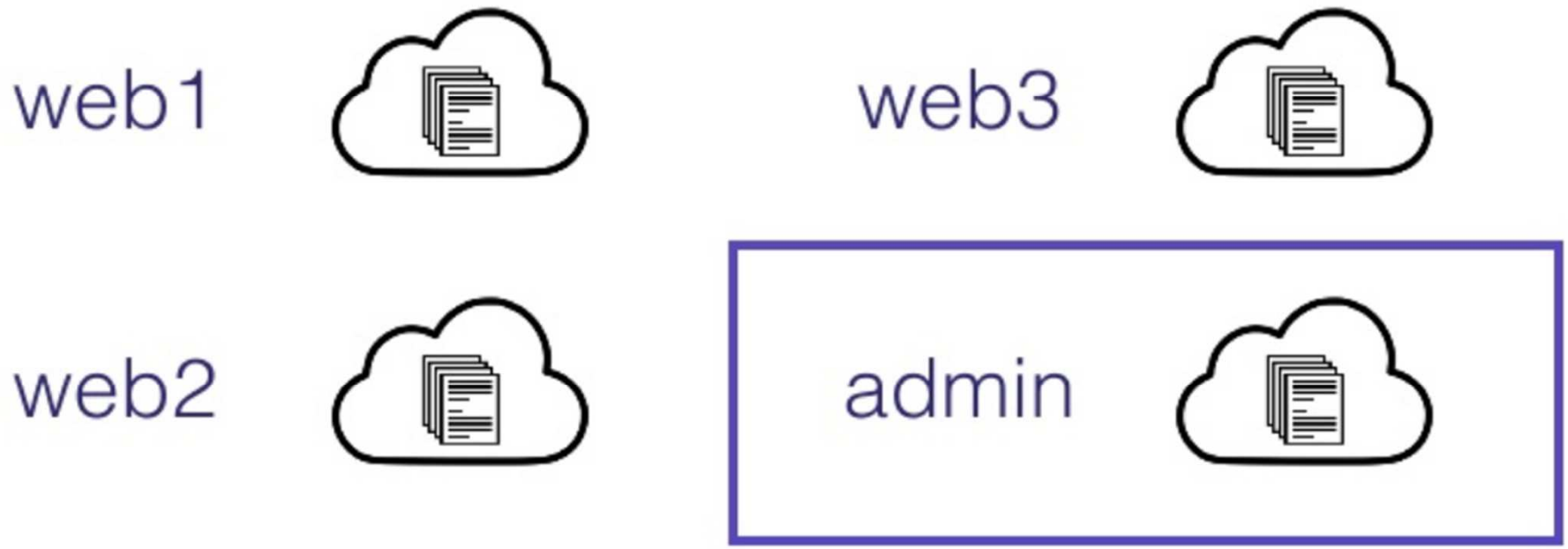
Tightly Coupled





"Using 3rd party services speeds dev but increases points of failure – design for things to fail regularly" @ghidinelli #CFSummit2015

Isolate Admin Tasks



Topic Recap

Big Picture

Hierarchies

Globalization

Currencies

Payments

Data Model

Multi-Tenancy

Custom Features

Custom Attributes

Sparse Columns

EAV

JSON/XML/NoSQL

APIs Live Forever

Webhooks

Infrastructure

12-Factor Apps

External Logging

Sticky Sessions

Async Processing

Admin Tasks

CF Summit 2015
@ghidinelli

Download slides: http://www.ghidinelli.com/cfsummit2015