

# Best Practices for Availability

David Owczarek | Sr. Engineering Manager, Document Cloud

MAKE IT AN  
EXPERIENCE

# Adobe is one of the largest and most diversified software companies in the world



17,000+

Employees in  
37 countries



34

Years of  
revolutionizing industries



\$5.85B

FY2016  
Revenue

78%

Of employees work in  
LEED certified  
workspaces

~3,900

Patents\*

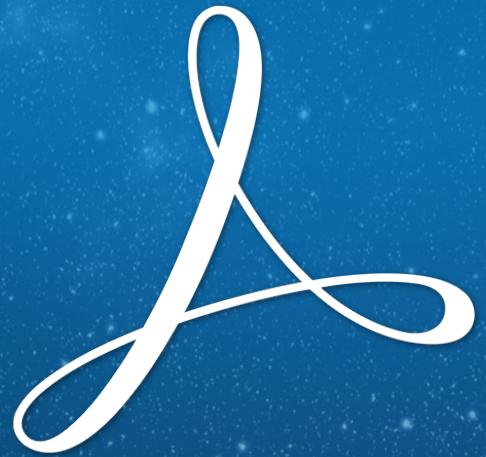
\$37.8M

Given to the  
community in 2016

# Adobe's Solutions



Adobe Creative Cloud

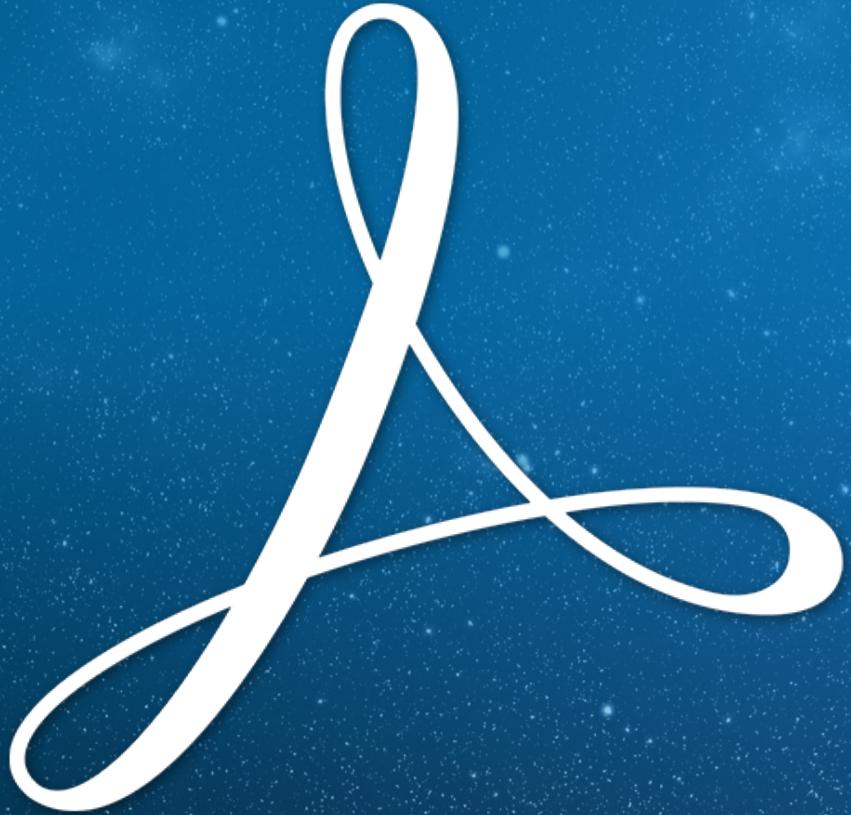


Adobe Document Cloud



Adobe Experience Cloud

# Adobe Document Cloud



Acrobat DC



Document Cloud PDF Services



Adobe Sign – Digital Signatures



Mobile Applications



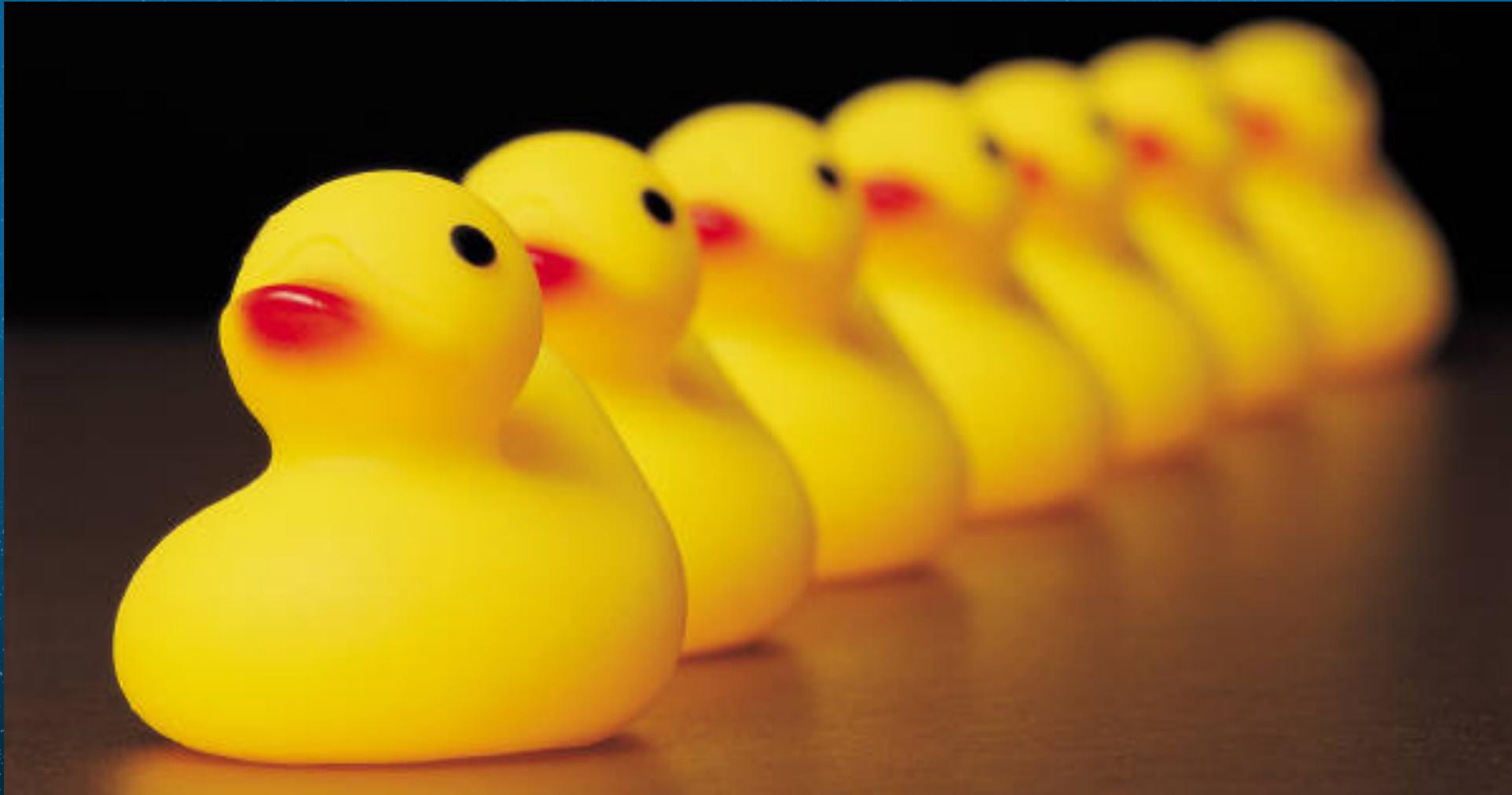
Integrations

Why am I here?

Provide *always-on* 99.99%  
services that power  
Adobe's customer  
*experience*  
-Abhay Parasnis,  
CTO



# Working across the Enterprise



# Overview of project spin-up

Recruit  
steering  
committee

Interview  
teams

Review  
results with  
teams

Recruit  
brainstorming  
teams

Collate  
results

# Enterprise Skepticism

- We don't work for you
- We don't even work in your business unit
- We have higher priority work
- We don't believe in your process
- It's a feel-good exercise for management
- The information we provide will be used against us
- We have a competing approach
- Our service doesn't fit the model

# How to get 100 engineering teams to do what you want

- \*VP, or C-level support with an explicit call to action
- Use your personal reputation to assuage concerns
- Co-opt your audience into solving the problem with you
- Walk through the alternatives
- Use emotional arguments to request concrete actions
- Develop your tactics collaboratively with key service teams
- Lead by example

# Results: Best Practices



# Best Practices for Availability

Culture and  
Ownership

Mature  
Service  
Management

Availability-  
Driven  
Architecture

# Culture & Ownership



Engineers support the service directly and are on-call



Quality objectives get resource priority



Defensive dependency risk management

# Mature Service Management

## Process

- Incident management
- Change management
- Problem management

## Monitoring

- Endpoint checks
- Synthetic transactions
- Log analysis

Secure development lifecycle

# Results: Data



# Failure Mode Analysis (FMA)

Term	Value
Frequency (f)	How many times per year?
Impact (i)	How many minutes of downtime?
Risk	( $f * i$ , in minutes per year)

Failure Mode	Frequency Requirements
Database	$f > 0$
Region	N/A
Application code	N/A
Dependent service	N/A
DNS	N/A
3 <sup>rd</sup> Party Vendor	$f \geq 2$
Instance	$f \geq 4$
DDOS	N/A
Cache	N/A
ELB	N/A
Zone	$f \geq 1$
Burst event	N/A

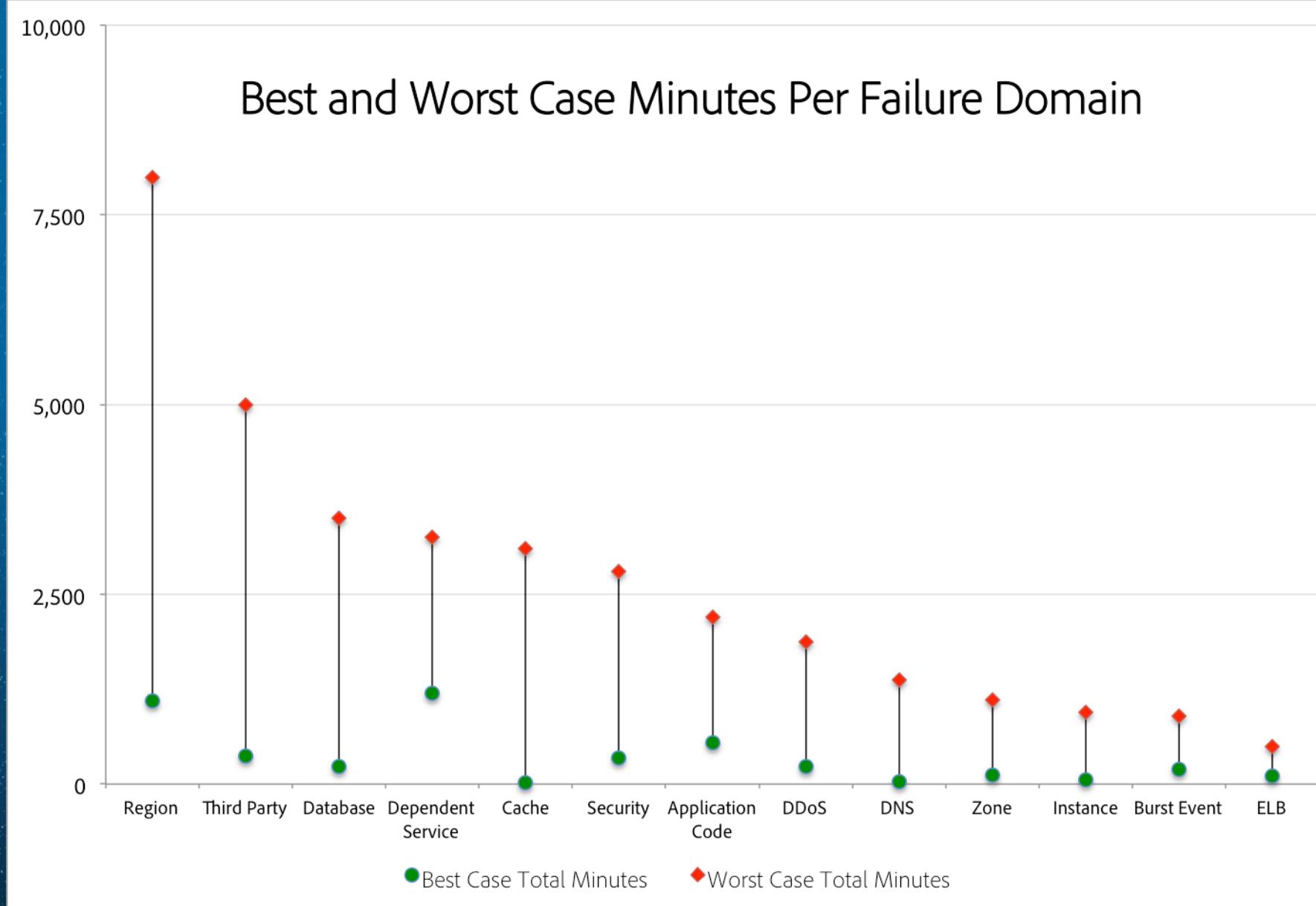
# Example failure mode analysis

Failure Mode	Frequency (incidents per year)	Impact (SLA minutes)	Risk ( $f * i$ , in minutes per year)
Database	.5	30*	120
Region	.5	30*	96
Application code	24	3	72
Dependent service	6	12	72
DNS	0	0	0
3 <sup>rd</sup> Party Vendor	2	5	10
Instance	36	0	0

\*30 minutes = disaster recovery time objective  
minutes/year

99.99% is 52

# Results



# Future work

FMA scores versus historical SLA performance – tying the two together

Better visualization of FMA data

Require all services to declare an SLA (99.99% or 99.9%)

Use the FMA as part of the software lifecycle (SDLC)



MAKE IT AN  
EXPERIENCE