Microservices and DevOps Journey at Wix.com

Aviran Mordo Head of WiXEngineering





@aviranm



http://www.aviransplace.com





WiX

Ð

A

Add Text

Themed Text

Site Title

Page Title

Large Heading

I'm a paragraph. Click here to add your own text and edit me. It's easy.

I'm a paragraph. Click here to add your own

Small Heading

text and edit me. It's easy.

Site Tools Help Upgrade

1

me

OUR LIFE

ved.

w Publis

Save Preview Home Book Now Company Fleet Services THE MOST RELIABLE LUXURY TRANSPORTATION SERVICE IN YOUR AREA! NO event is too small for us! Size We cater for Weddings, a Night on the town, Birthdays, Corporate Travel and many more. H: Position IIII :: vimeo Show on All Pages

Strip Shape Video Music Social Contact Menu List Blog Store

Big Title ALL CAPS TITLE

SMALL TITLE

Titles

Story Title



Wix In Numbers

- Over 86M users
- Static storage is >2Pb of data
- 3 data centers + 2 clouds (Google, Amazon)
- 2B HTTP requests/day
- 1200 people work at Wix





Over 200 Microservices on Production

WiXEngineering

Microservices - What Does it Take

Do / Use

Continuous Delivery DevOps Circuit Breaker Feature Flags Throttlers Monitoring Testing Message Bus RPC REST

Consider / Understand

SLA Distributed Transactions Backward / Forward Compatibility Clustering Conway's law Development / product lifecycle Boundary KISS YANGI LEAN



Microservices - What Does it Take RPC Message Flags /cle ward κιςς tions Monitoring Boundary DistributedCluste ODevC Feature **Delivery** Testing Circuit Breaker

WiXEngineering

How to Get There? (Wix's journey)

http://gpstrackit.com/wp-content/uploads/2013/11/VanishingPointwRoadSigns.jpg

In the Beginning of Time

About 6 years ago

http://p1.pichost.me/i/11/1339236.jpg

Initial Architecture

Tomcat, Hibernate, custom web framework

- One database
- Stateful login (Tomcat session), Ehcache, file uploads
- No consideration for performance, scalability and testing
- Intended for short-term use





The Monolithic Giant

- One monolithic server that handled everything
- Dependency between features
- Changes in unrelated areas caused deployment of the whole system
- Failure in unrelated areas will cause system wide downtime





Breaking the System Apart

https://upload.wikimedia.org/wikipedia/commons/6/67/Broken_glass.jpg



Concerns and SLA

Edit websites

- Many feature request
- Lower performance requirement
- Lower availability requirement
- Write intensive

View sites, created by Wix editor

- Not many product changes
- High performance
- High availability
- Read intensive



Phase 1

Mono-Wix



Extract Public Service

Editor service (Mono-Wix)

Public service

WiXEngineering





Why 2 Monoliths? Baby Steps

Editor need fast development; (microservices => decoupling) Public needs stability; microservices => scalability /resilient



Separation by Product Lifecycle

Oecouple architecture => Decouple teams

Deployment independence

Areas with frequent changes



Separation by Service Level

- Scale independently
- Use different data store
- Optimize data per use case (Read vs Write)
- Run on different datacenters / clouds / zones
- System resiliency (degradation of service vs. downtime)
- Faster recovery time





@aviranm



WIXEngineering



WiXEngineering

http://blogs.adobe.com/captivate/2011/03/training-adding-interactivity-to-elearning-courses-with-adobe-captivate-5.html/time-to-learn-clock

Service Boundary

WiXEngineering

Separation of Databases

Copy data between segments

Optimize data per use case (read vs. write intensive)

Different data stores





Serialization

WiXEngineering

Serialization / Protocol

Binary?
JSON / XML / Text?
HTTP?





Serialization / Protocol - Tradeoffs

- Readability?
- Performance?
- Debug?
- Tools?
- Monitoring?
- Dependency?





API Transport/Protocol



How to Expose an API











API Versioning

WiXEngineering





A-Synchronous

WiXEngineering

Which Queuing System to Use





Service Discovery

WiXEngineering







Resilience

WiXEngineering

What does the Arrow Mean?





Failure Points = Network I/O

Retry policy
 Retry only on idempotent operations
 Circuit breaker
 Throttlers
 Retry only on idempotent operations
 Be careful – you may cause downtime



Degradation of Service

- Feature killer (Killer feature)
- Fallbacks
- Self healing













WiXEngineering

Build visibility into service

Application Inform	nation	Usage Summary (Incoming Calls Only)			
Title	App Integration Bus - WebApp		11		
Artifact	app-integration-bus-web	Total Calls	15		
Version	1.179.0-SNAPSHOT	Total Successfull calls	15		
Build	cd45eff4f2b3bddda09ffc1d7bf30597782458e8	Throughput	0.		
Build Timestamp	20140608-1049	Error Rate	0.		
Server Name		System Errors	0/		
Uptime	3 days, 20 hours and 21 minutes.	Business Errors	0/		
Server Time Zone	America/Chicago				
Server Startup	08/06/2014 07:05:30.421	Prev. Startup Information			
Server Current Time	12/06/2014 03:27:20.653	Version 1.178.0-SNAPSHOT	01/06/20		
F		Version 1.176.0-SNAPSHOT	29/05/20		

	1h	48h
Total Calls	15	713
Total Successfull calls	15	708
Throughput	0.3 rpm	0.2 rpm
Error Rate	0.00 %	0.70 %
System Errors	0/0/0/0	0/0/0/5
Business Errors	0/0/0/0	0/0/0/0

01/06/2014 09:30:42.651 29/05/2014 03:15:30.798 Version 1.175.0-SNAPSHOT 29/05/2014 01:39:05.230

Usage Statis	tics										
Method						Calls	Throughput (rpm)	Average (mSec)	Max (mSec)	System Errors	Business Errors
(METER) c.w.a.d	.CachedReadOnlyLo	calAppsRepository.	getAppByIdOpt		48h 1h	277,611 1,037	97.46 17.78	0.0 0.0	77.8 0.0	0/0/0/0 0/0/0/0	0/0/0/0 0/0/0/0
(RPC_CLIENT) r.p	.c.c.w.m.s.ReadOnly	MetaSiteManager.g	etMetaSiteByIns	tanceld (http://a	48h 1h	76,499 370	26.86 6.34	8.3 10.7	6,505.0 1,088.0	1/0/0/0 0/0/0/0	0/0/0/0 0/0/0/0
r.p.c.c.w.m.s.R	eadOnlyMetaSiteMai	nager.getMetaSiteB	yInstanceId (http	o://api.aus.wixpre	ss.co	om/meta-	-site-manager	ReadOnlyMe	taSiteMana	ager)	
100.00%		Total Calls	mm hr.								
10.00%		Max Time	_M								
		Avg Time	- human								
1.00%		System Fatal									
0.10%		System Error									
0.10%		System Warning									
0.01%	┛╷┚╷┚╷╴╷╸╸	System Recover	λ								
100 - 316	< 1	Business Fatal									
316 - 100	1 - 3	Business Error									
3160 - 10000	10 - 32	Business									
> 10000	32 - 100	Warning	·•								
In Call Times	(msec.)	Business Recover									
time	Exception				Mess	age					
11/06/2014 01:38:51.808	com.wixpress.framework.rpc.client.exceptions.RpcTransportException			c.w.f.r.c.e.RpcTransportException – Server connection timed out [endpoint= url=\'http://apl.aus.wixpress.com/meta-site-manager/ReadOnlyMetaSiteMa at com.wixpress.framework.rpc.client.RpcOverHttpClientSclass.renderFault(at com.wixpress.framework.rpc.client.BlockingRpcOverHttpClient.renderFault at com.wixpress.framework.rpc.client.BlockingRpcOverHttpClientSsanonfunSes							



Ownership

WiXEngineering

Team Work



- Microservice is owned by a team
- You build it you run it
- No microservice is left without a clear owner
- Microservice is NOT a library it is a live production system



What is the Right Size of a Microservice?





The Size of a Microservice is the Size of the Team That is Building it.

"Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations" **Conway, Melvin**



What did you Learn from Just 2 Services

- Service boundary
- Monitoring infrastructure
- Serialization format
- Synchronous communication protocol (HTTP/Binary)
- Asynchronous (queuing infra)
- Service SLA
- API definition (REST/ RPC / Versioning)
- Data separation
- Deployment strategy
- Testing infrastructure (integration test, e2e test)
- Compatibility (backwards / forward)

Continue to Extract More Microservices

WiXEngineering



When to Extract a New Microservice



Microservice or Library?

I need time zone from an IP address

O I create deployment dependency?

- What is DevOps overhead (managing middleware) ?
- Who owns it?

Ooes it have its own development lifecycle?

Ooes it fit the scalability / availability concerns?

_{@avi}Can a different team develop it?







QULE O

Which Technology Stack to Use



Free to Chose?

S kafka

Microservices gives the freedom to use a different technology stacks.
 Enables innovation

ka

mongoDB

MyS

 cassandra

WiXEngineering

Couchbase





Default to the Stack You Know how to Operate.









QULE O



Polyglotic System?



RULEOR

Limit your Stack

- Code reuse
- Cross cutting concerns (session, security, auditing, testing, logging...)
- Faster system evolution
- Development velocity





http://wallpaperbeta.com/dogs_kiss_noses_animals_hd-wallpaper-242054/

What else will you learn

- Distributed transactions
- System monitoring
- Distributed traces
- Tradeoff of a new microservice vs. extending an existing one
- Deployment strategy and dependency
- Handling cascading failures
- Team building/splitting

Summary



wavnann

Why Microservices

Scale engineering

Development Velocity

Scale system





wavnamm



Microservices is the First Post DevOps Architecture



Every Microservice is a **dev** ps Overhead





Microservices Guidelines & Tradeoffs



- Each service has its own DB schema (if one is needed)
 - Gain Easy to scale microservices based on service level concerns
 - Tradeoff system complexity, performance
- Only one service should write to a specific DB table(s)
 - Gain Decoupling architecture faster development
 - Tradeoff system complexity / performance
- May have additional read-only services that accesses the DB (not recommended)
 - Gain Performance gain
 - Tradeoff coupling
- Services are stateless
 - Gain Easy to scale out (just add more servers)
 - Tradeoff performance / consistency



MONOLITH



MICRO SERVICES



ngineering

Thank You



wavnannn



Aviran Mordo Head of Engineering

http://engineering.wix.com

@WixEng





