



Microservices + DevOps + Oracle Cloud **= A Bright Future**

Sai Janakiram Penumuru

Chief Technologist, Oracle ACE Director

Sai Janakiram Penumuru

- Chief Technologist – Apps Transformations - DXC Technology
- Co-Founder, Vice President - All India Oracle Users Group (AIOUG)
- Oracle ACE Director, Oracle Developer Champion
- Oracle VM SIG Leader www.oraclevmSIG.org
- Blog: www.oadba.com; www.oracle12c.info
- Contacts – ps.janakiram@gmail.com ; twitter - @sai_penumuru



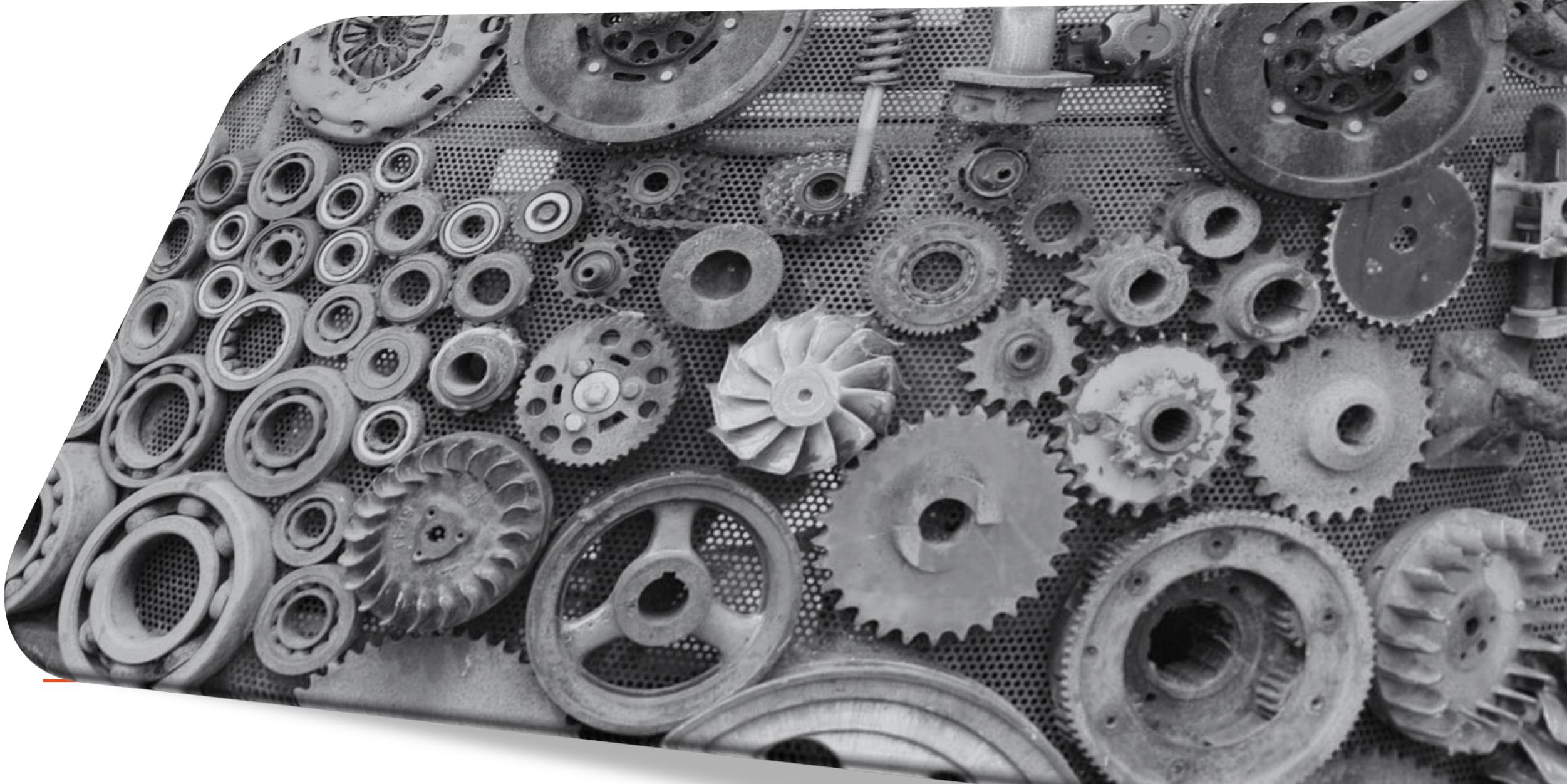
Agenda

For 45 Minutes

- Microservices
- DevOps
- Oracle Cloud



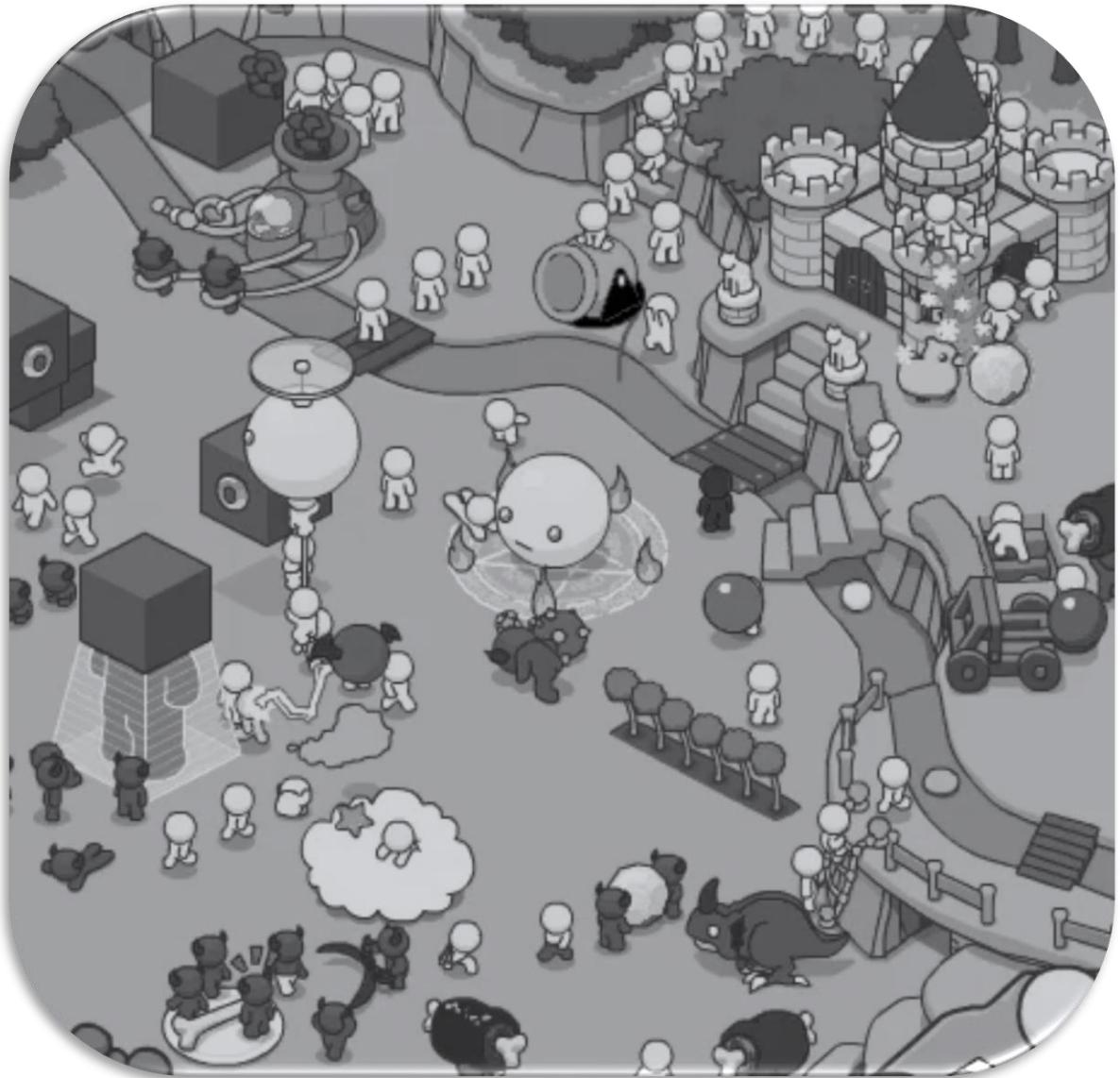
Microservices



What is Microservices

- The first one, **Microservices** is a software architecture style in which complex applications are composed of **small, independent processes** communicating with each other using **language-agnostic APIs**.
- The second one describes Microservices Architecture as a style of architecture that promotes **business alignment** by developing **applications** as set of **small independent** and **self-contained** services that directly caters to an **atomic business activity**.

Martin Fowler: Microservices



Microservices are Analogous to Unix Utilities

Same Concept, Different Decade

"Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface."



Doug McIlroy
Inventor of the Unix Pipe

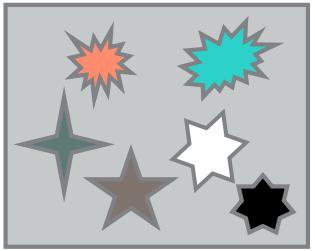
- Unix Executable: Does one thing and does it well
- Runs independent of other commands
- Produces text-based response

```
curl -v -H "Accept: application/json" -H "Content-type: application/json" -X POST  
-d '{"productId":645887,"quantity":"1"}'  
"http://localhost:8840/rest/ShoppingCart/"
```

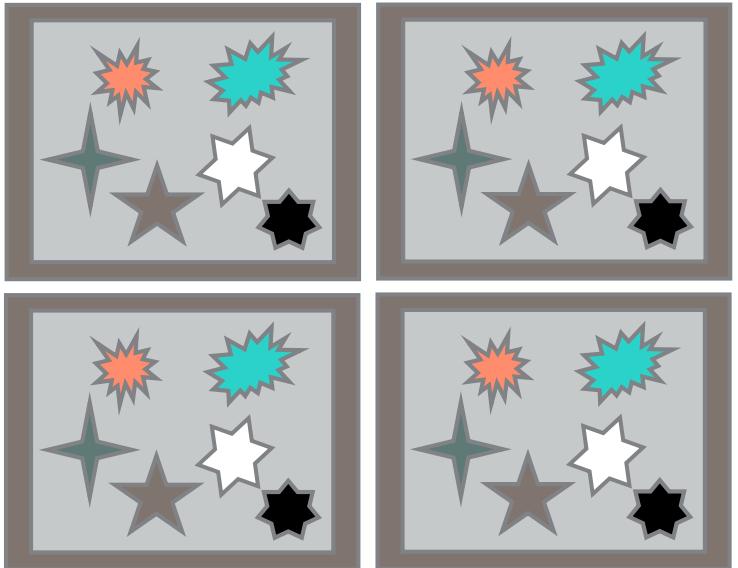
- Microservice: Does one thing and does it well
- Runs independent of other microservices
- Produces text-based response to clients

Monolithic versus Microservices

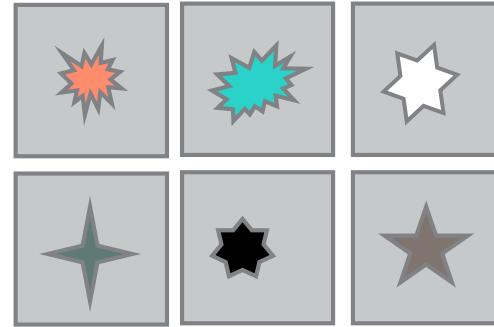
A monolithic application puts all its functionality into a single module



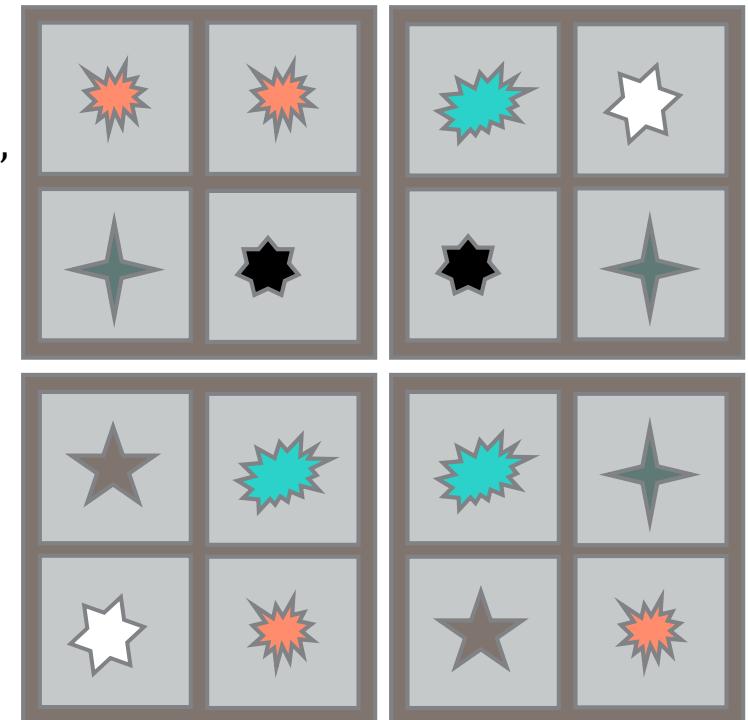
... and scales by replicating the module on multiple servers



A microservices architecture puts each element of business functionality into a separate service



... and scales by distributing these services, in parallel, across servers, replicating as needed.



Monolith vs

Microservices



- Single, Monolithic App
- Must Deploy Entire App
- One Database for Entire App
- Organized around Technology Layers
- One Technology Stack for Entire App

- Many, smaller minimal function Microservices
- Can deploy Each Microservices independently
- Each Microservices often has its own Data store
- Organized around Business capabilities
- Choice of Technology for each Microservices

Fundamentally

Do you want...

Traditional App Development

Easier Deployment/Ops

- One big block of code, sometimes broken into semi-porous modules
- Complexity handled inside the big block of code
- Each big block is hard to develop but easy to deploy

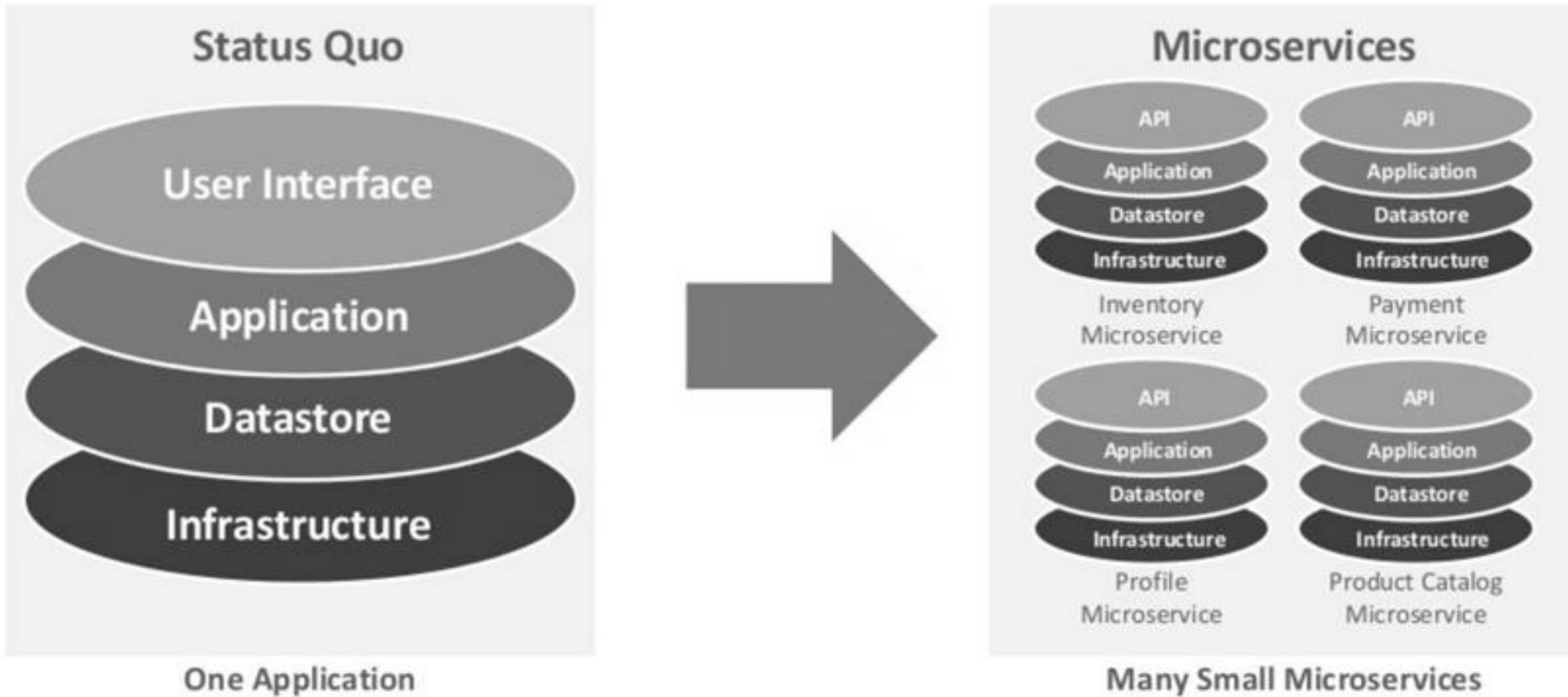
Microservices

Easier Development

- Many small blocks of code, each developed and deployed independently
- Complexity encapsulated in each microservice
- Each microservice is easy to develop but hard to deploy

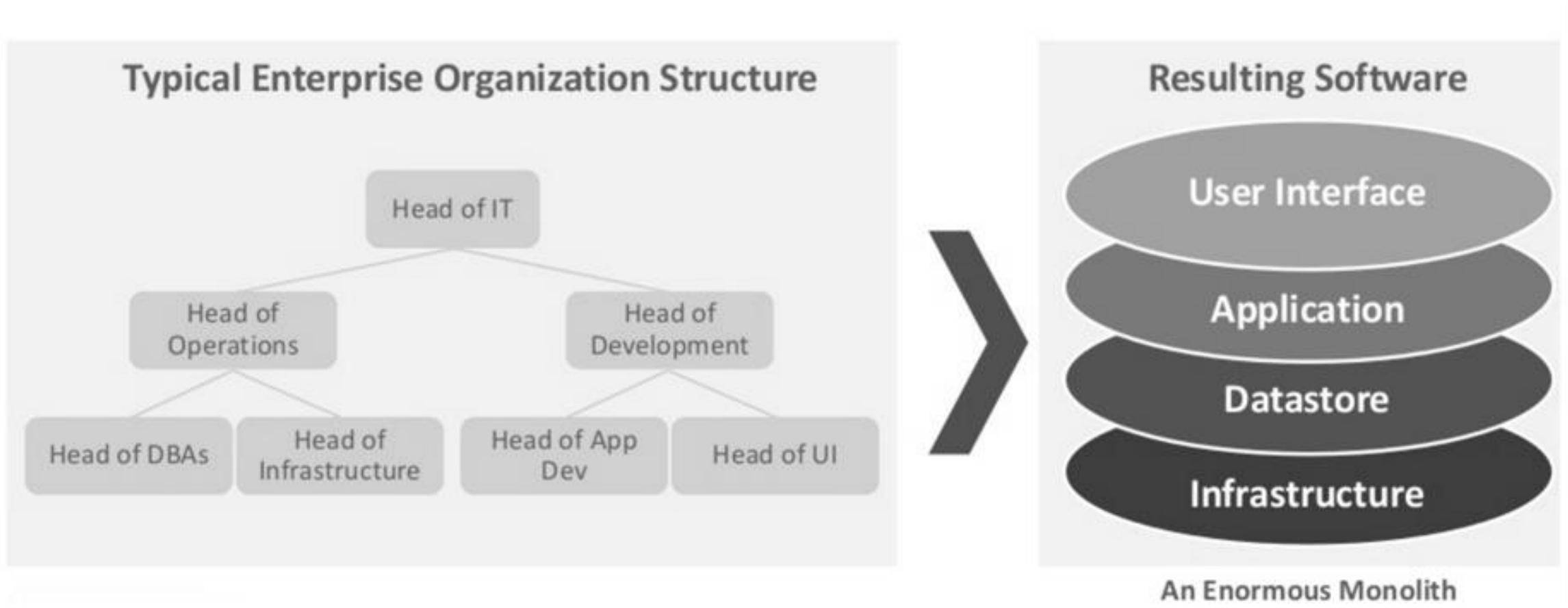


Microservices Apps Are Developed/Deployed Independently



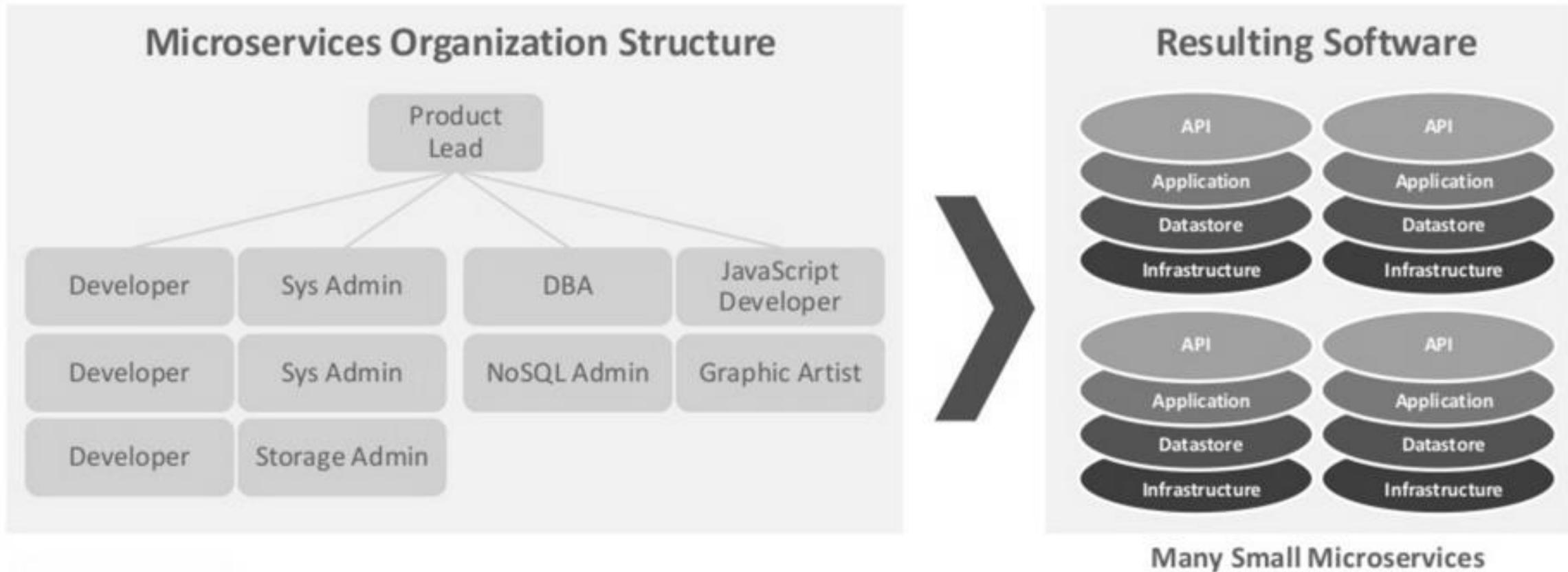
Conway's Law in Action

Any piece of software reflects the organizational structure that produce it



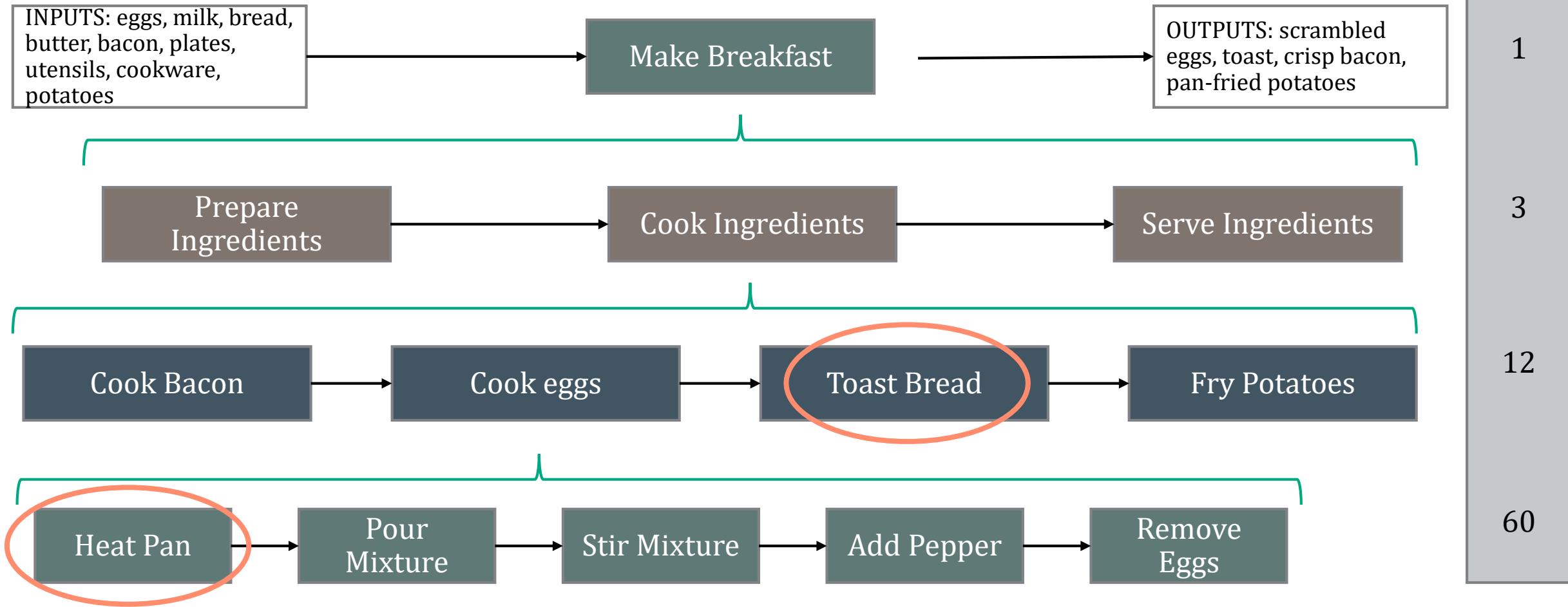
Successful Teams Structure their teams around Products

Build Small vertical teams



What is the right level

Let's make breakfast - objective of the use of Microservices is to **increase re-use**

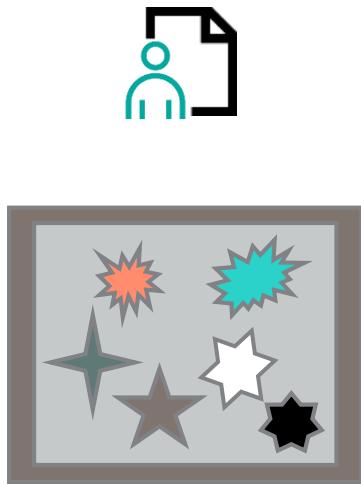


The right level of decomposition is Business and Organization dependent.
What that entity executes as an **atomic business function**.

Characteristics of Microservices

Organized around business capabilities

Traditional Approach



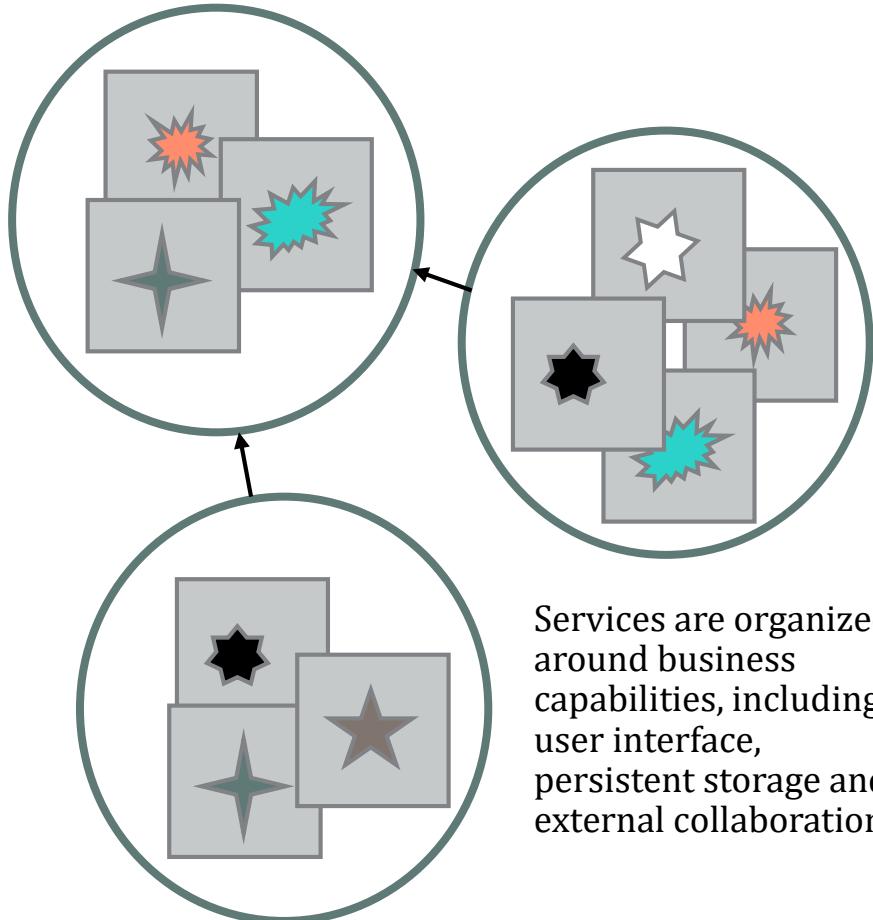
Presentation Layer

Application Layer

Database Layer

Siloed functional Teams

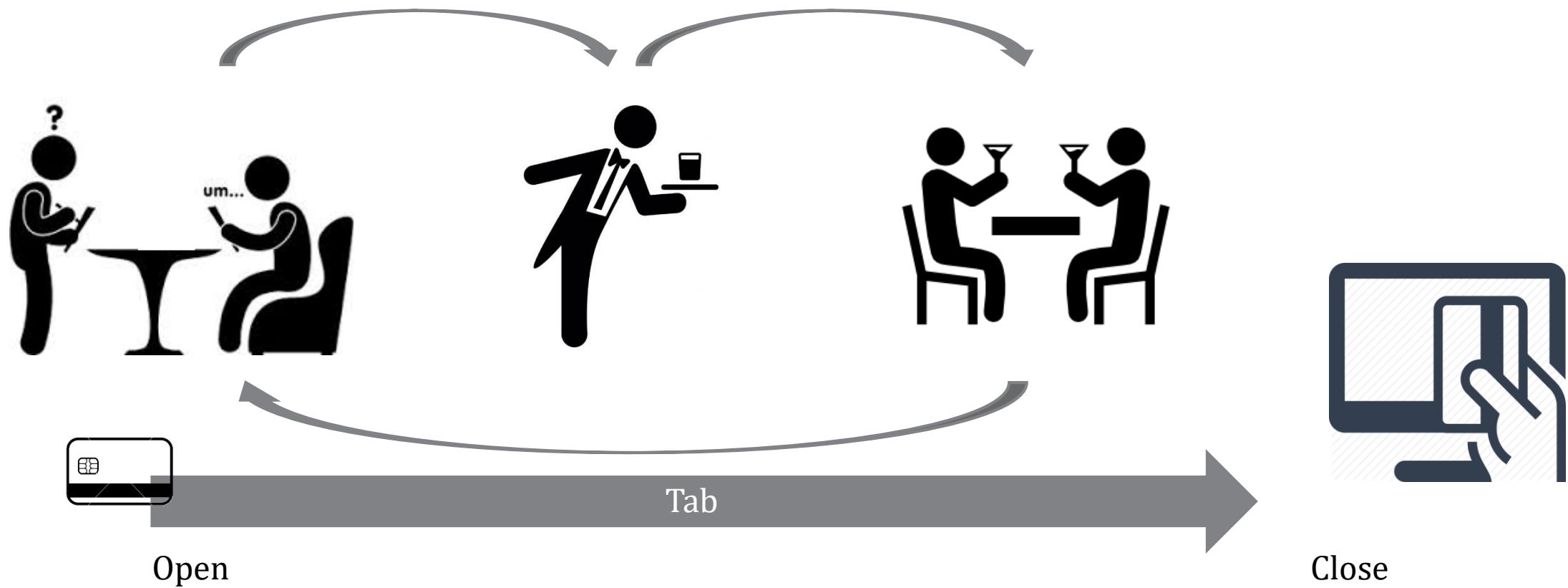
Microservices Approach



Cross-functional Teams

Traditional Transaction

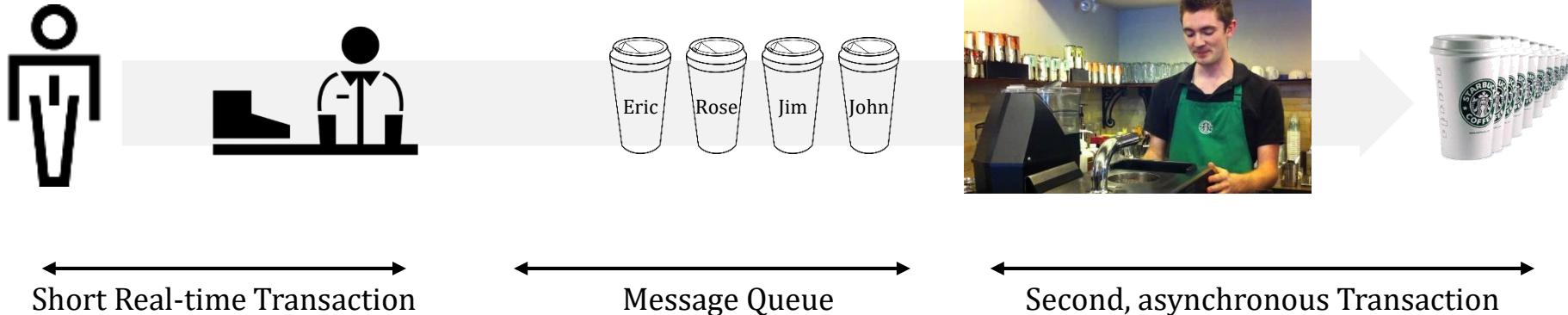
Let's go to the pub



A session remains open during your presence at the pub.

Transaction less coordination between services

The Starbucks model



If something goes wrong compensating operations are used to address the issue. Failure does not result in loss of state as Microservices are stateless.

Failure is imminent, graceful recovery is paramount.

By making the link between Microservices asynchronous, having one service going down does not affect the operation of the other. Synchronous calls are considered harmful. If one service no longer responds, the other keeps hanging for a response and becomes unavailable.

Characteristics of Microservices

Design for Failure

Traditional Approach

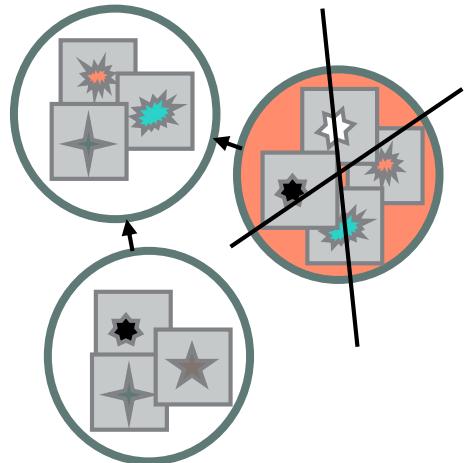
500 Internal Server Error

Microservices Approach

Graceful recovery and process resilience are mandatory in the architecture and design of Microservices based solutions.

Unfortunately the XYZ service is currently down, we apologize for the inconvenience. However our remaining services are operating normally.

1. Service down, recognize the failure, spin up a new parallel instance or do graceful degradation.
2. If the service is slow, timeout, either act (spawn a new parallel instance to compensate for extra workload) or report to the user the unavailability and potentially allow him to retry.
3. Service responds with unexpected content, is your service capable of recognizing and handling this? Microservices must have a stable and published contract.
4. When-ever possible, use caching mechanisms so the service can still respond if the back-end is down.

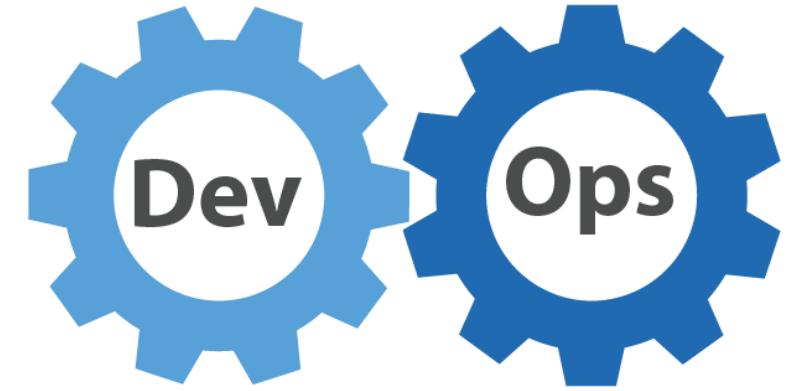


What is required for Microservices?

Rapid Provisioning
Monitoring
Rapid Application Deployment
DevOps Culture



DevOps



Familiar?

DevOps seeks to solve this

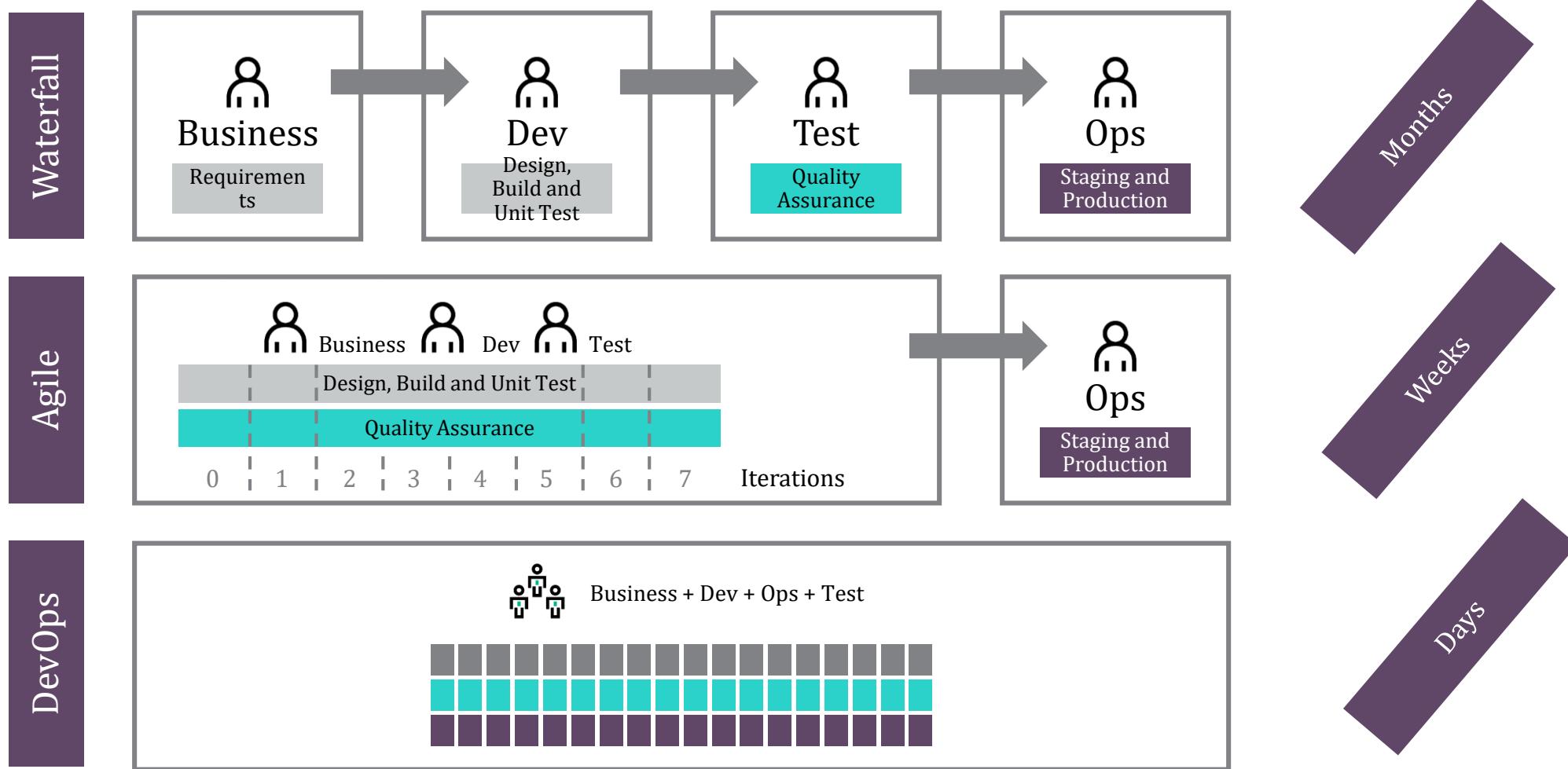


Dev and Ops Constantly Argue

Code is written... it's your problem now



Shift in priorities is demanding DevOps

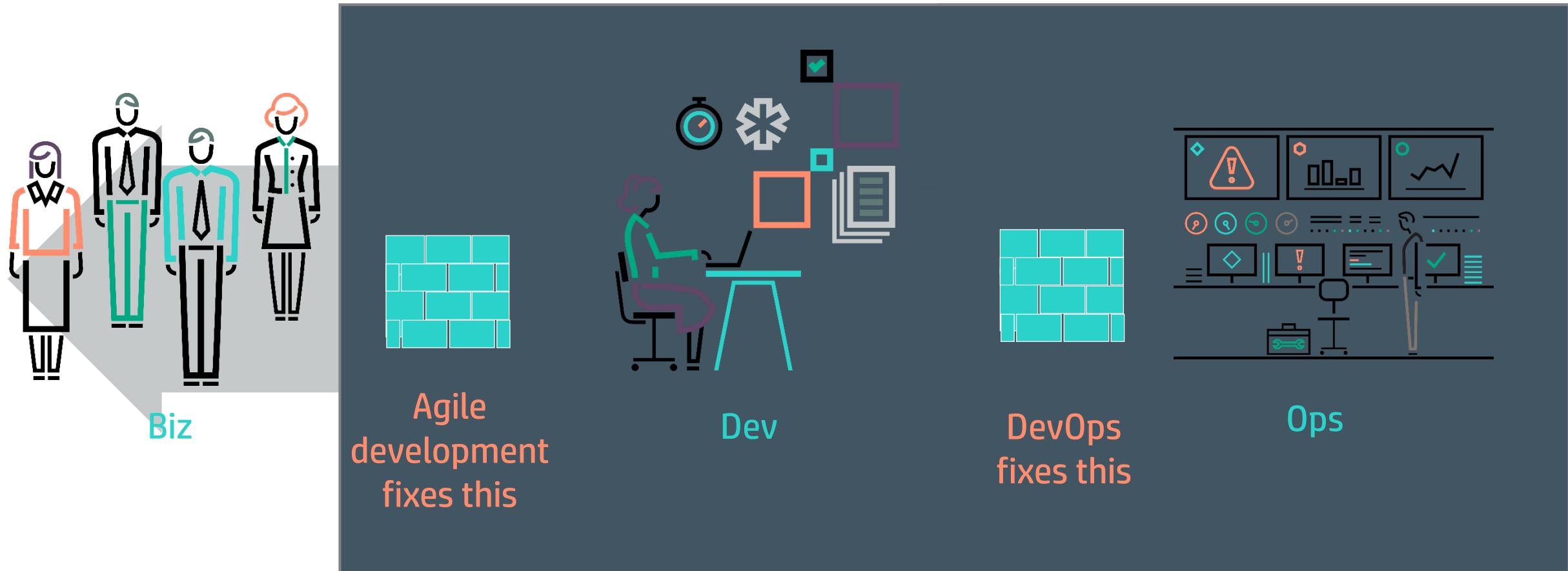


One product team!

- Shared objectives, Shared customer-oriented goals, Shared accountability

Building a continuous application development supply chain

Create “BizDevOps”

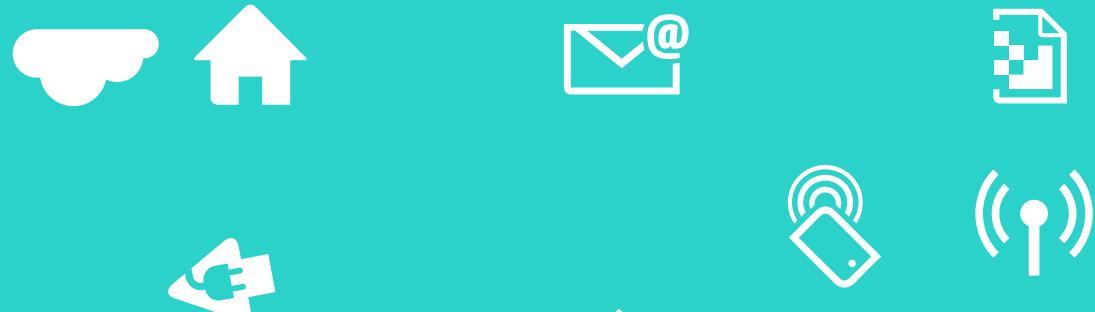


Per "[What is Devops](#)" by dev2ops.org

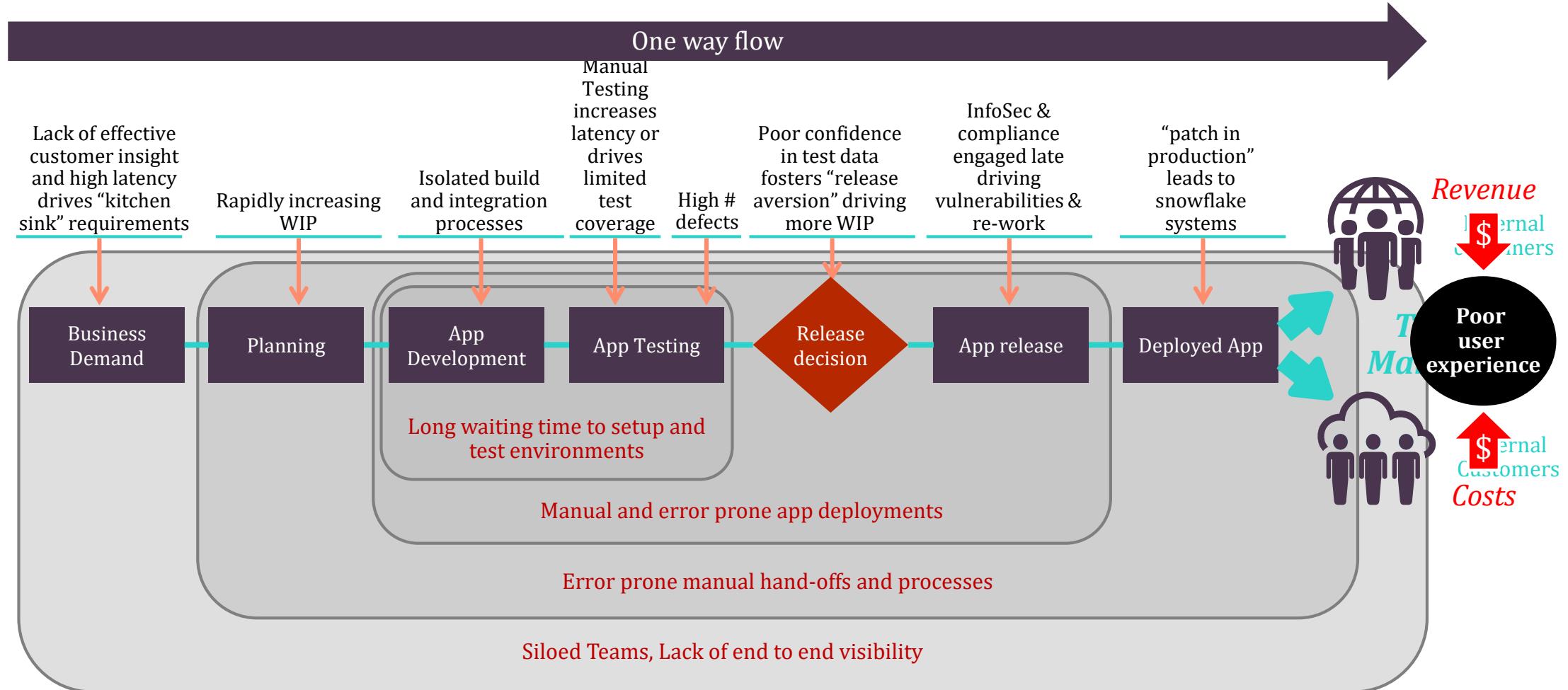


It worked
fine in dev,
it's ops' problem now

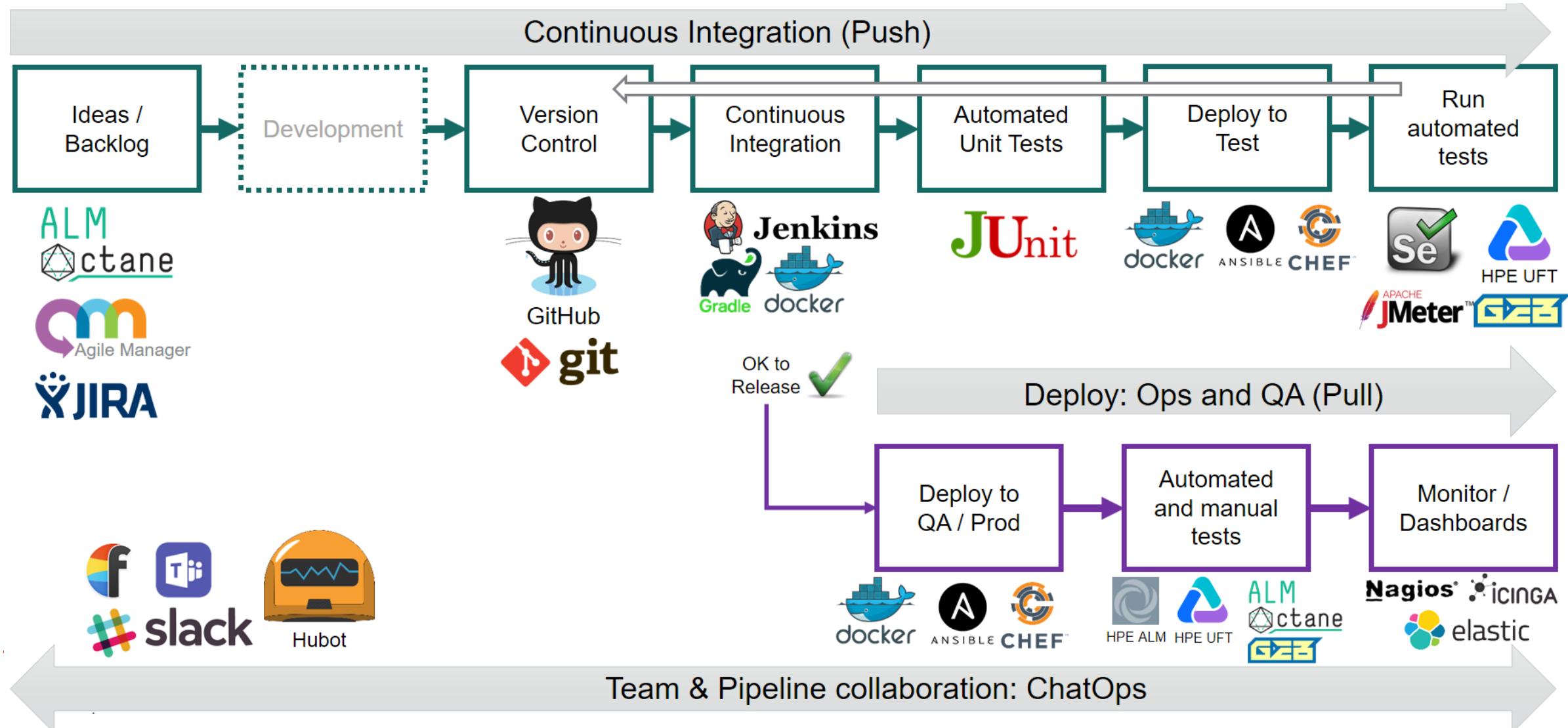
Culture Clash



Today's application delivery is full of obstacles and challenges



Technology Stack - Tools



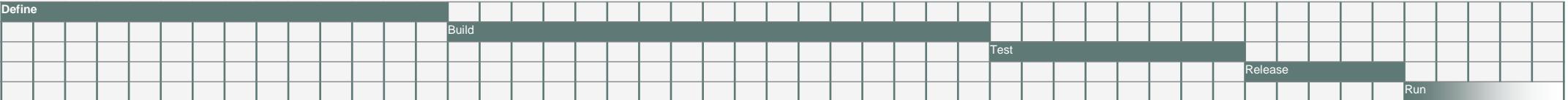
YOU FOOL!

It's not about the tools, it's about the **CULTURE!**

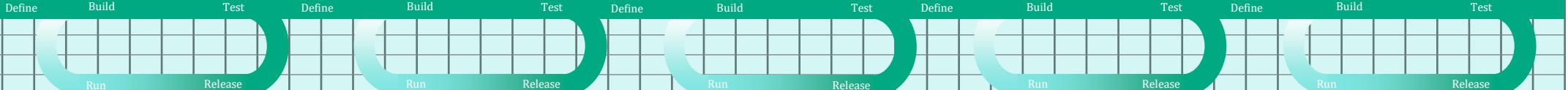


Change in Culture

Traditional IT Development Life Cycle	Directive	Task Focused	Dedicated Roles	Resistance to Change	Manual	Project Focused
	Long and rigid planning process related to current and historical needs	Step-by-step delivery in a waterfall or production line approach	Departmentalized roles working in silos and throwing jobs over the wall	Changes are problematic as they need re-defining, re-scheduling and re-working, leading to delays and increased costs	Slow and costly manual deployments and implementations	Working in silos leads to a narrow view aligned with limited or initial requirements



Agile/DevOps Development Life Cycle	Adaptive	Goal Focused	Teamwork	Responsive to Change	Automated	Business Focused
	React and responds to changing needs constantly	A business oriented approach and attitude which sees "the bigger picture"	Collaborative and integrated methods to empower teams across the business and the client	Agile methods quickly identify areas for change or improvements	Automated deployment tools and virtualized environments to accelerate releases	Working in teams leads to a holistic view in line with today's business needs



Why Do DevOps?

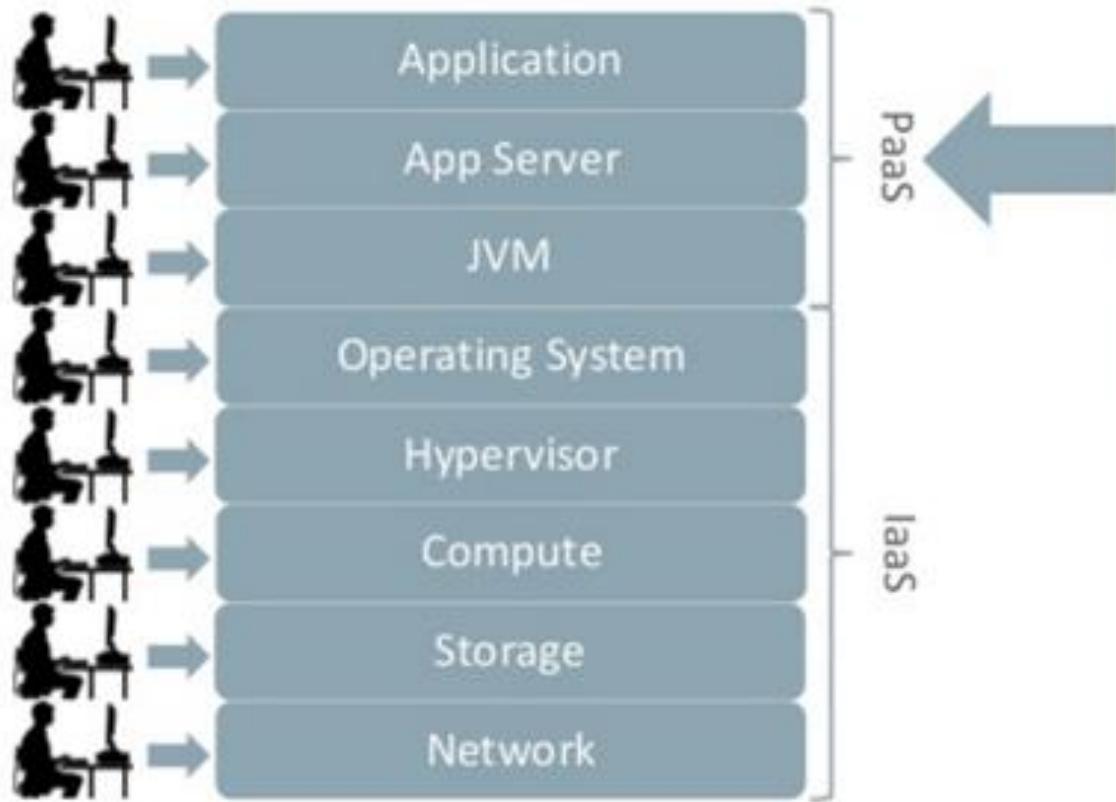
- enabling faster feature time to market
- increased customer satisfaction & market share
- employee productivity and happiness
- organizations to win in the marketplace

In contrast, organizations that require weeks or months to deploy software are at a significant disadvantage in the marketplace.

COMPANY	DEPLOY FREQUENCY	DEPLOY LEAD TIME	RELIABILITY	CUSTOMER RESPONSIVENESS
Amazon	23,000/day	minutes	high	high
Google	5,500/day	minutes	high	high
Netflix	500/day	minutes	high	high
Facebook	1/day	minutes	high	high
Twitter	3/week	minutes	high	high
Typical enterprise	once every 9 months	months or quarters	low/medium	low/medium

PaaS/IaaS Now Allows Resources to be Easily Provisioned

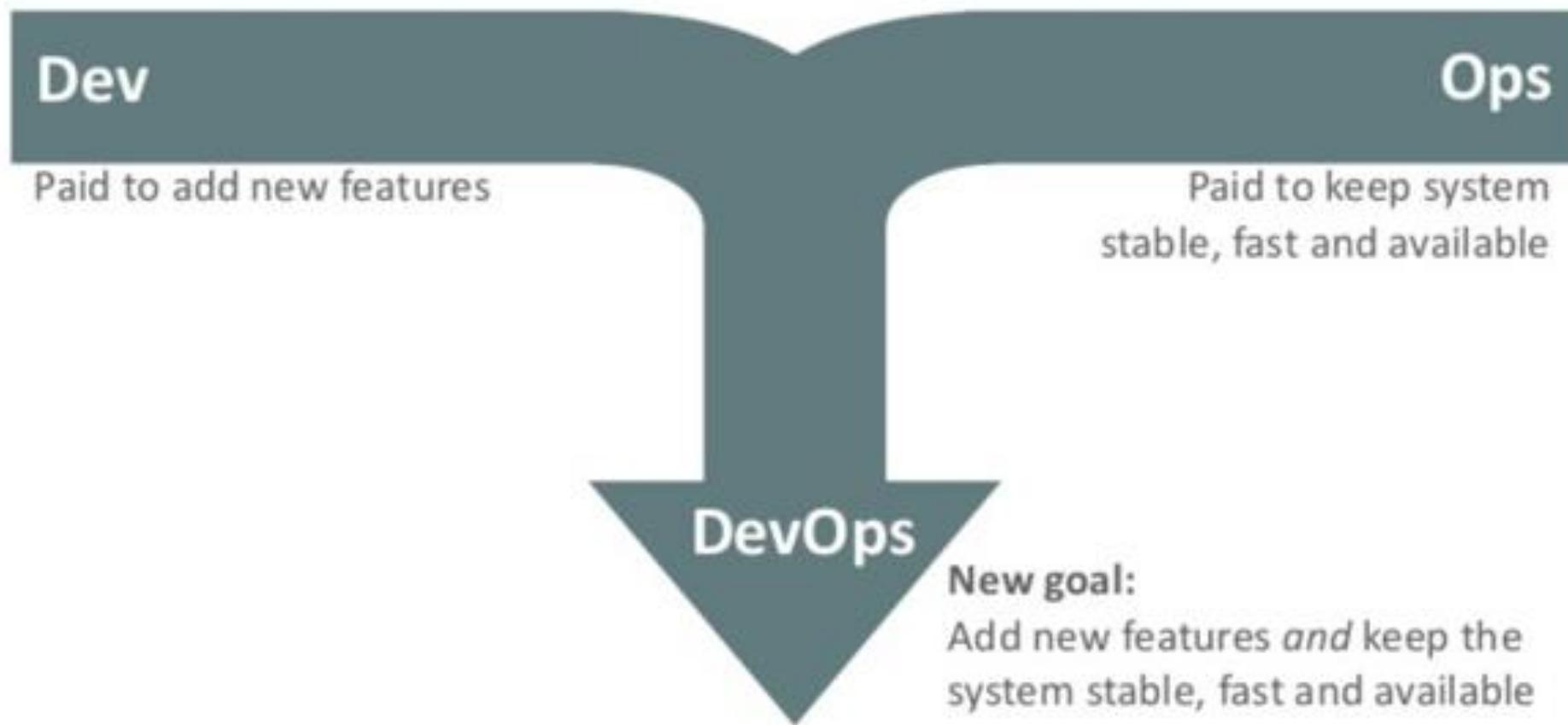
Past: Ops manually provisioned each layer



Today: Developers can provision entire stacks of hardware + software through REST API

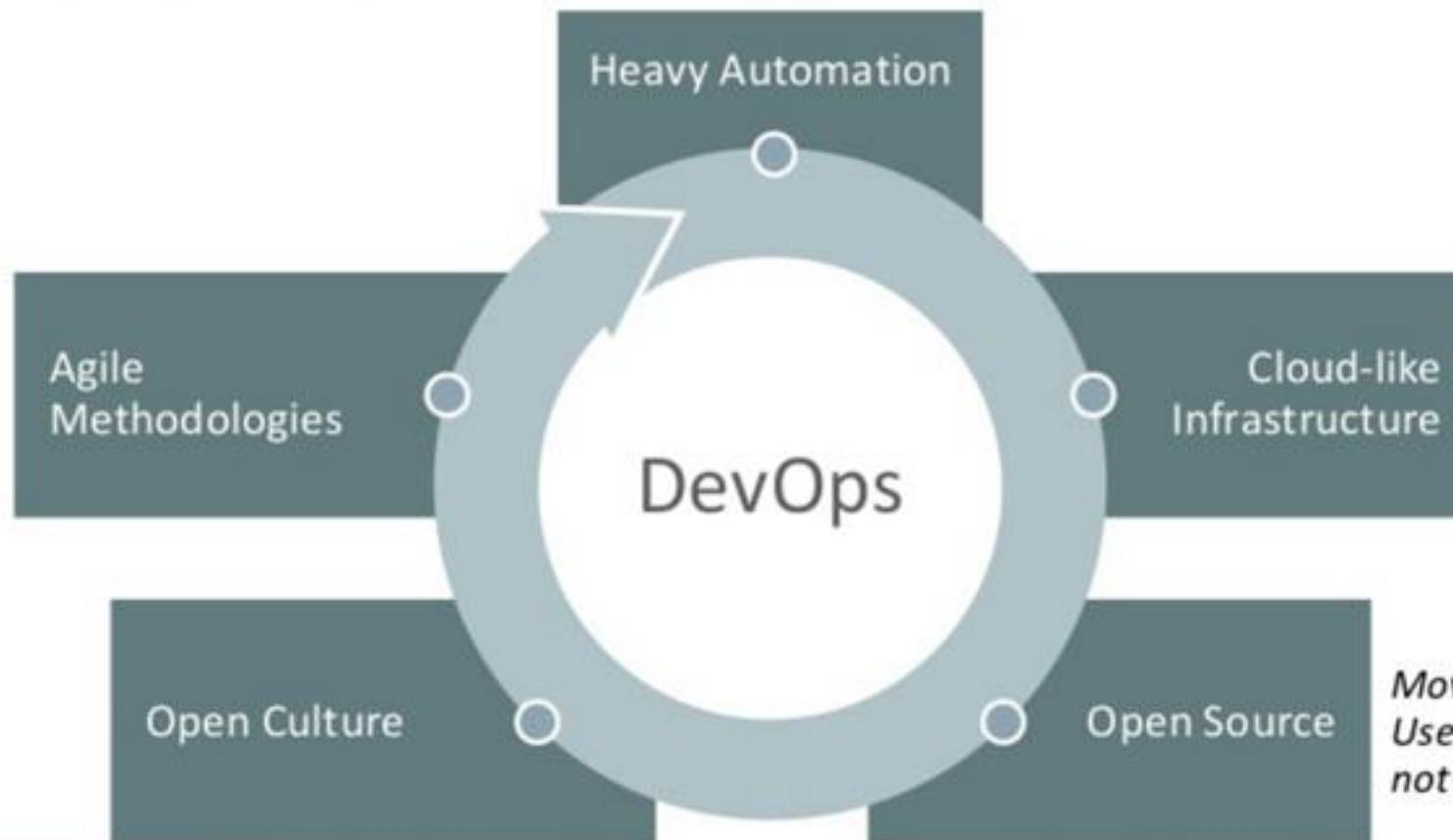
DevOps Principles

Cultural movement enabled by technology



Characteristics of DevOps Movement

Principles have been around for decades

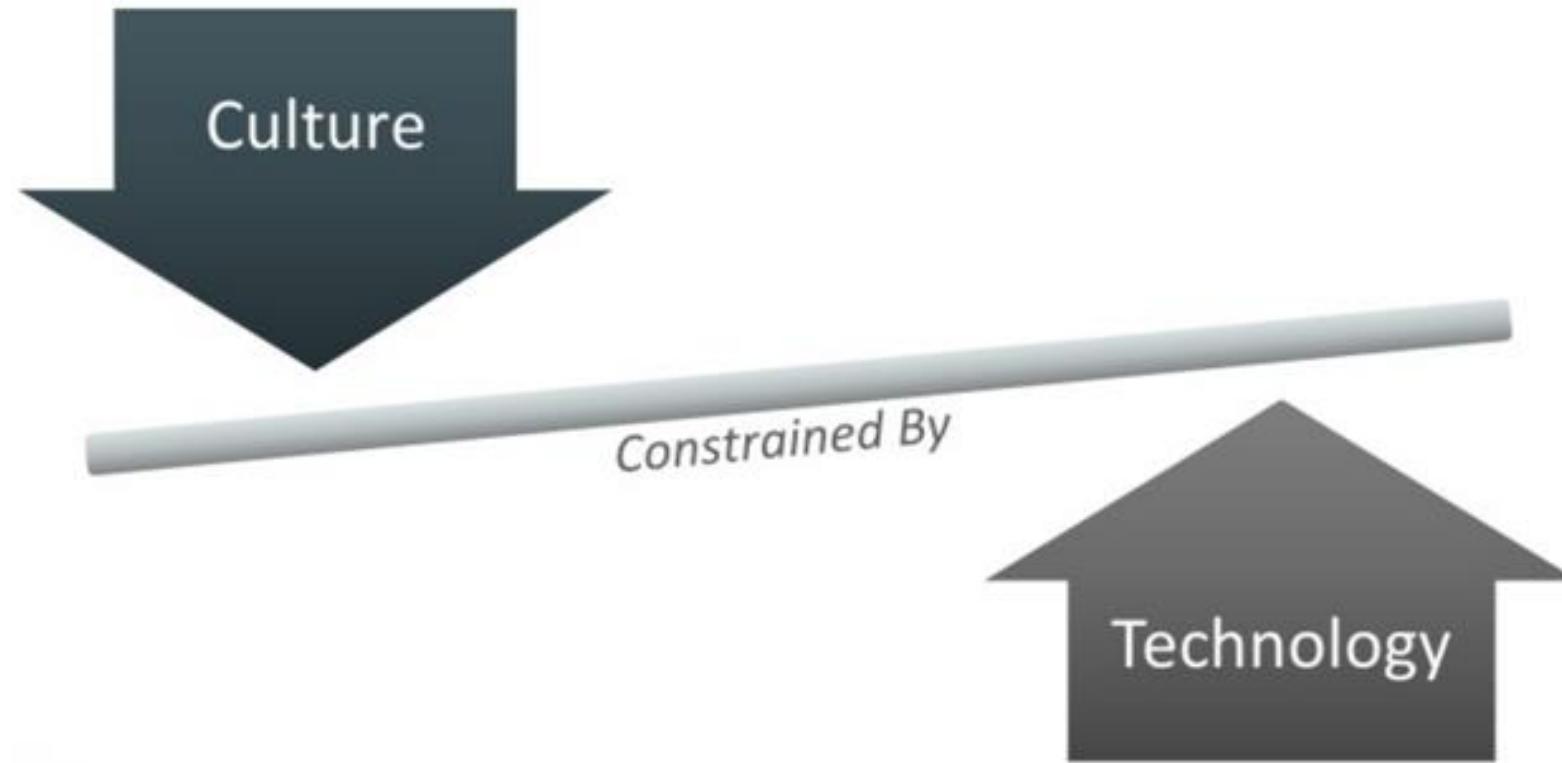


*Movement began in startup community.
Use of open source seen as integral but
not technically necessary*



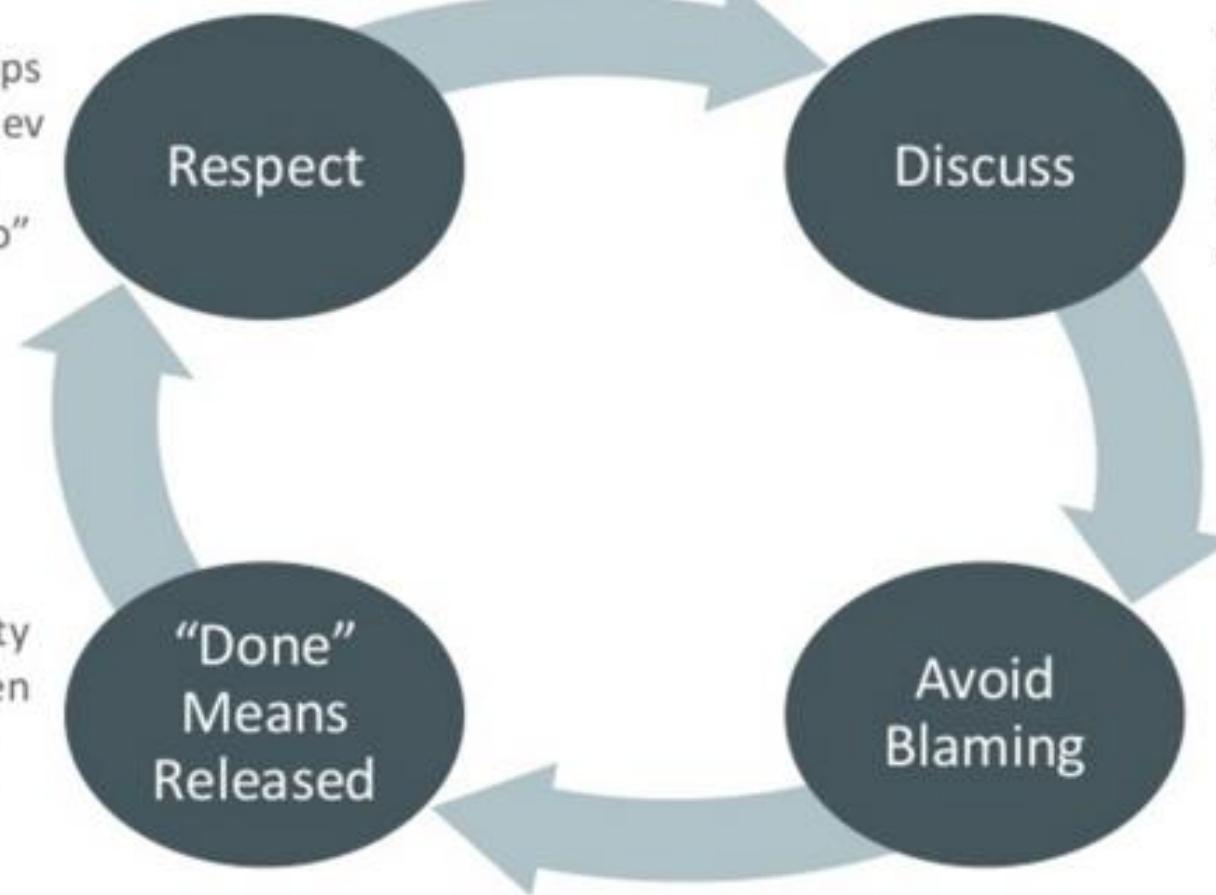
DevOps = Culture + Technology Movement

Culture is what's behind DevOps; Technology is the enabler



DevOps Tenet #1: Culture

- Dev respect for ops
- Ops respect for dev
- Don't stereotype
- Don't just say "no"



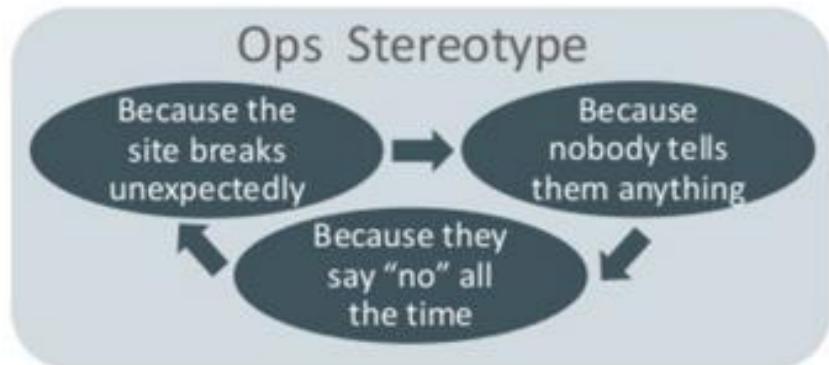
- Dev's responsibility does not end when it's in production
- “Throwing it over the wall” is dead

- Don't hide things
- Ops should be in dev discussions
- Dev should be in ops discussions
- Shared runbooks/escalation plans
- Ops should give devs access to systems

- No fingerpointing!

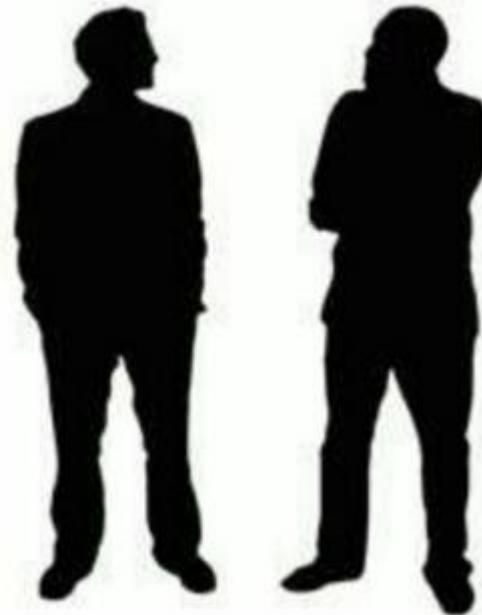
Build Respect

- Developers should respect Operations
- Operations should respect Developers
- Don't just say "no"
- Don't stereotype!



Discuss

- Don't hide things! Open communication on both sides
- Operations should be in Developer discussions
- Developers should be in Operations discussions
- Build shared runbooks/escalation plans
- Operations should give Developers direct access to systems



Avoid Blaming

- No fingerpointing!
- Development should have enough Operations culpability to share the blame in an outage
- Operations should have enough Development culpability to share the blame in an outage



Actively Build Trust

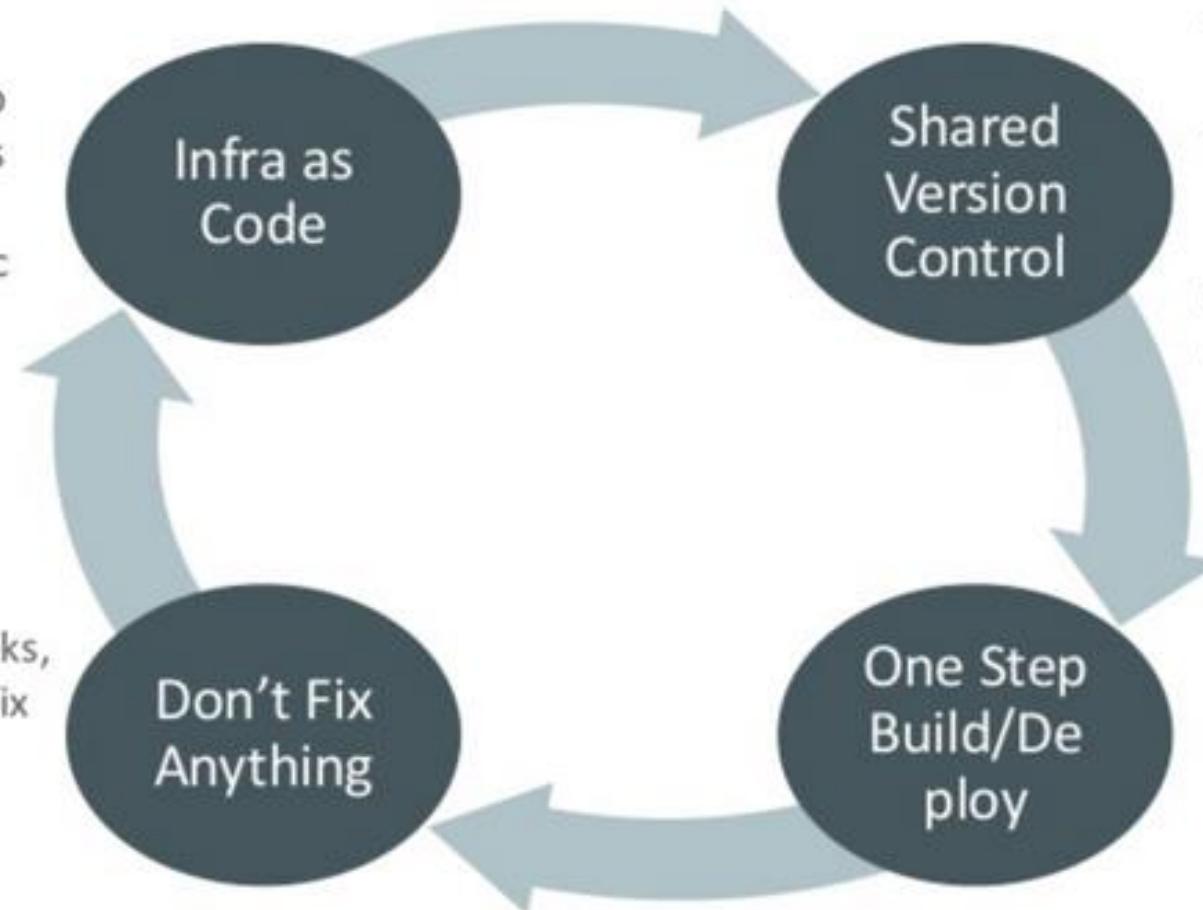
Trust is the #1 ingredient to a successful DevOps culture



Dev + Ops + Social Activity Outside Work + Time = Trust

DevOps Tenet #2: Technology

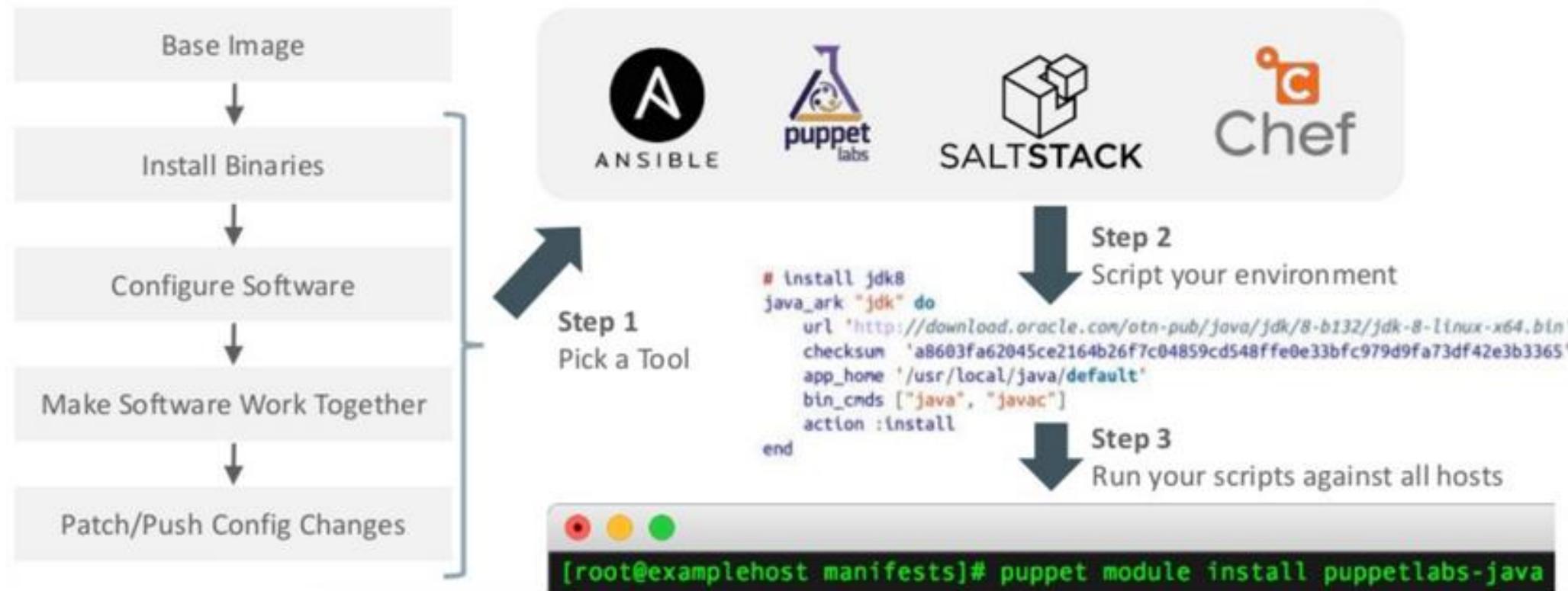
- Use config mgmt to build environments
- Scripts checked in and managed as src



- Single system for code and build artifacts
 - Every time someone commits a change it triggers a build and automated build verification tests
 - Ship trunk
 - Enable features through flags
-
- One button build/deploy (manual)
 - Scheduled builds/deploys
 - If verification fails, stop and alert

Infrastructure as Code

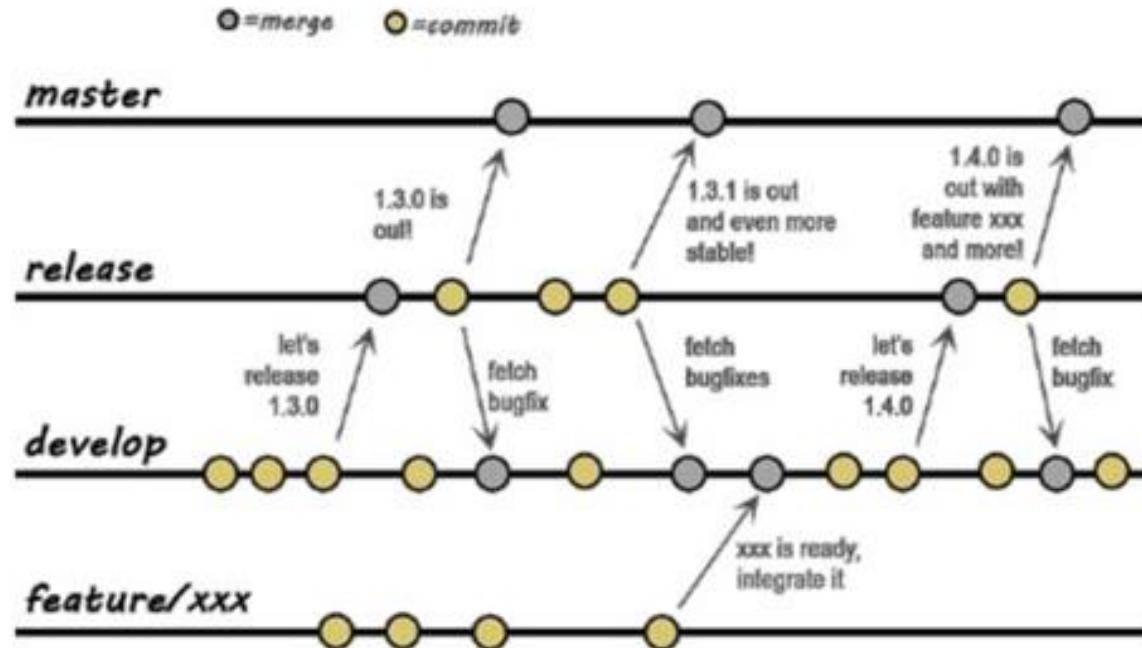
Manage it as you would any other source code



Shared Version Control

Surprisingly not well adopted

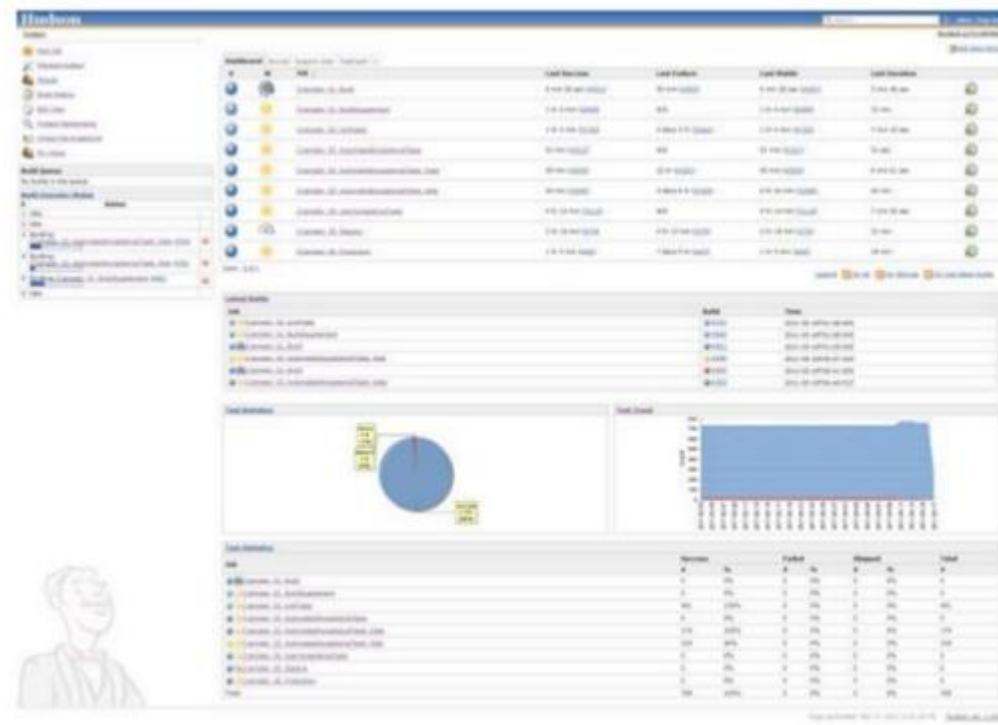
- Single system for code and build artifacts
- Every time someone commits a change, consider triggering a build + automatic verification tests
- Always ship trunk!
- Enable features through flags



One Step Build/Deploy

Set it and forget it

- Manual one button build/deploy
- Scheduled builds - every day, every week, etc
- Builds triggered by code checkins
- If post-build validation fails, report it



Automated Testing using Robot

1. Integrate Robot With Maven

```
1 <plugin>
2   <groupId>org.robotframework</groupId>
3   <artifactId>robotframework-maven-plugin</artifactId>
4   <version>1.2</version>
5   <executions>
6     <execution>
7       <id>robot</id>
8       <phase>integration-test</phase>
9       <goals>
10         <goal>run</goal>
11       </goals>
12     </execution>
13   </executions>
14 </plugin>
```

3. Run Test

```
mvn verify
```

2. Write a Simple Selenium-based Test

```
3 *** Settings ***
4 Library
5 Test Set Up
6 Test Tear Down
7
8 *** Test Cases ***
9
10 Basic Test
11   Open Browser
12   Page Should Contain
13   Page Should Contain Textfield
14   Page Should Contain Textfield
15   Page Should Contain Button
16   Input Text
17   Input Text
18   Click Button
19   Wait Until Page Loaded
20   Page Should Contain
21   Close Browser
22
23
24 SeleniumLibrary
25 Start Selenium Server
26 Stop Selenium Server
27
28 http://localhost:7001/basicWeb
29 Basic Webapp
30 j_idt10:name
31 j_idt10:amount
32 j_idt10:j_idt18
33 j_idt10:name
34 j_idt10:amount
35 j_idt10:j_idt18
36 The money have been deposited
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
317
318
319
319
320
321
322
323
324
325
326
327
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
346
347
347
348
349
349
350
351
352
353
354
355
356
357
357
358
359
359
360
361
362
363
364
365
366
367
367
368
369
369
370
371
372
373
374
375
376
377
377
378
379
379
380
381
382
383
384
385
386
387
387
388
389
389
390
391
392
393
394
395
396
397
397
398
399
399
400
401
402
403
404
405
406
407
407
408
409
409
410
411
412
413
414
415
415
416
417
417
418
419
419
420
421
422
423
424
425
426
426
427
428
428
429
430
430
431
432
433
434
435
436
436
437
438
438
439
440
440
441
442
443
444
445
446
446
447
448
448
449
450
450
451
452
453
454
455
456
456
457
458
458
459
460
460
461
462
463
464
465
466
466
467
468
468
469
470
470
471
472
473
474
475
476
476
477
478
478
479
480
480
481
482
483
484
485
486
486
487
488
488
489
489
490
490
491
492
493
494
495
496
496
497
498
498
499
499
500
500
501
502
503
504
505
505
506
507
507
508
509
509
510
511
511
512
513
513
514
515
515
516
517
517
518
519
519
520
521
521
522
523
523
524
525
525
526
527
527
528
529
529
530
531
531
532
533
533
534
535
535
536
537
537
538
539
539
540
541
541
542
543
543
544
545
545
546
547
547
548
549
549
550
551
551
552
553
553
554
555
555
556
557
557
558
559
559
560
561
561
562
563
563
564
565
565
566
567
567
568
569
569
570
571
571
572
573
573
574
575
575
576
577
577
578
579
579
580
581
581
582
583
583
584
585
585
586
587
587
588
589
589
590
591
591
592
593
593
594
595
595
596
597
597
598
599
599
600
601
601
602
603
603
604
605
605
606
607
607
608
609
609
610
611
611
612
613
613
614
615
615
616
617
617
618
619
619
620
621
621
622
623
623
624
625
625
626
627
627
628
629
629
630
631
631
632
633
633
634
635
635
636
637
637
638
639
639
640
641
641
642
643
643
644
645
645
646
647
647
648
649
649
650
651
651
652
653
653
654
655
655
656
657
657
658
659
659
660
661
661
662
663
663
664
665
665
666
667
667
668
669
669
670
671
671
672
673
673
674
675
675
676
677
677
678
679
679
680
681
681
682
683
683
684
685
685
686
687
687
688
689
689
690
691
691
692
693
693
694
695
695
696
697
697
698
699
699
700
701
701
702
703
703
704
705
705
706
707
707
708
709
709
710
711
711
712
713
713
714
715
715
716
717
717
718
719
719
720
721
721
722
723
723
724
725
725
726
727
727
728
729
729
730
731
731
732
733
733
734
735
735
736
737
737
738
739
739
740
741
741
742
743
743
744
745
745
746
747
747
748
749
749
750
751
751
752
753
753
754
755
755
756
757
757
758
759
759
760
761
761
762
763
763
764
765
765
766
767
767
768
769
769
770
771
771
772
773
773
774
775
775
776
777
777
778
779
779
780
781
781
782
783
783
784
785
785
786
787
787
788
789
789
790
791
791
792
793
793
794
795
795
796
797
797
798
799
799
800
801
801
802
803
803
804
805
805
806
807
807
808
809
809
810
811
811
812
813
813
814
815
815
816
817
817
818
819
819
820
821
821
822
823
823
824
825
825
826
827
827
828
829
829
830
831
831
832
833
833
834
835
835
836
837
837
838
839
839
840
841
841
842
843
843
844
845
845
846
847
847
848
849
849
850
851
851
852
853
853
854
855
855
856
857
857
858
859
859
860
861
861
862
863
863
864
865
865
866
867
867
868
869
869
870
871
871
872
873
873
874
875
875
876
877
877
878
879
879
880
881
881
882
883
883
884
885
885
886
887
887
888
889
889
890
891
891
892
893
893
894
895
895
896
897
897
898
899
899
900
901
901
902
903
903
904
905
905
906
907
907
908
909
909
910
911
911
912
913
913
914
915
915
916
917
917
918
919
919
920
921
921
922
923
923
924
925
925
926
927
927
928
929
929
930
931
931
932
933
933
934
935
935
936
937
937
938
939
939
940
941
941
942
943
943
944
945
945
946
947
947
948
949
949
950
951
951
952
953
953
954
955
955
956
957
957
958
959
959
960
961
961
962
963
963
964
965
965
966
967
967
968
969
969
970
971
971
972
973
973
974
975
975
976
977
977
978
979
979
980
981
981
982
983
983
984
985
985
986
987
987
988
989
989
990
991
991
992
993
993
994
995
995
996
997
997
998
999
999
1000
1000
```

4. View Results

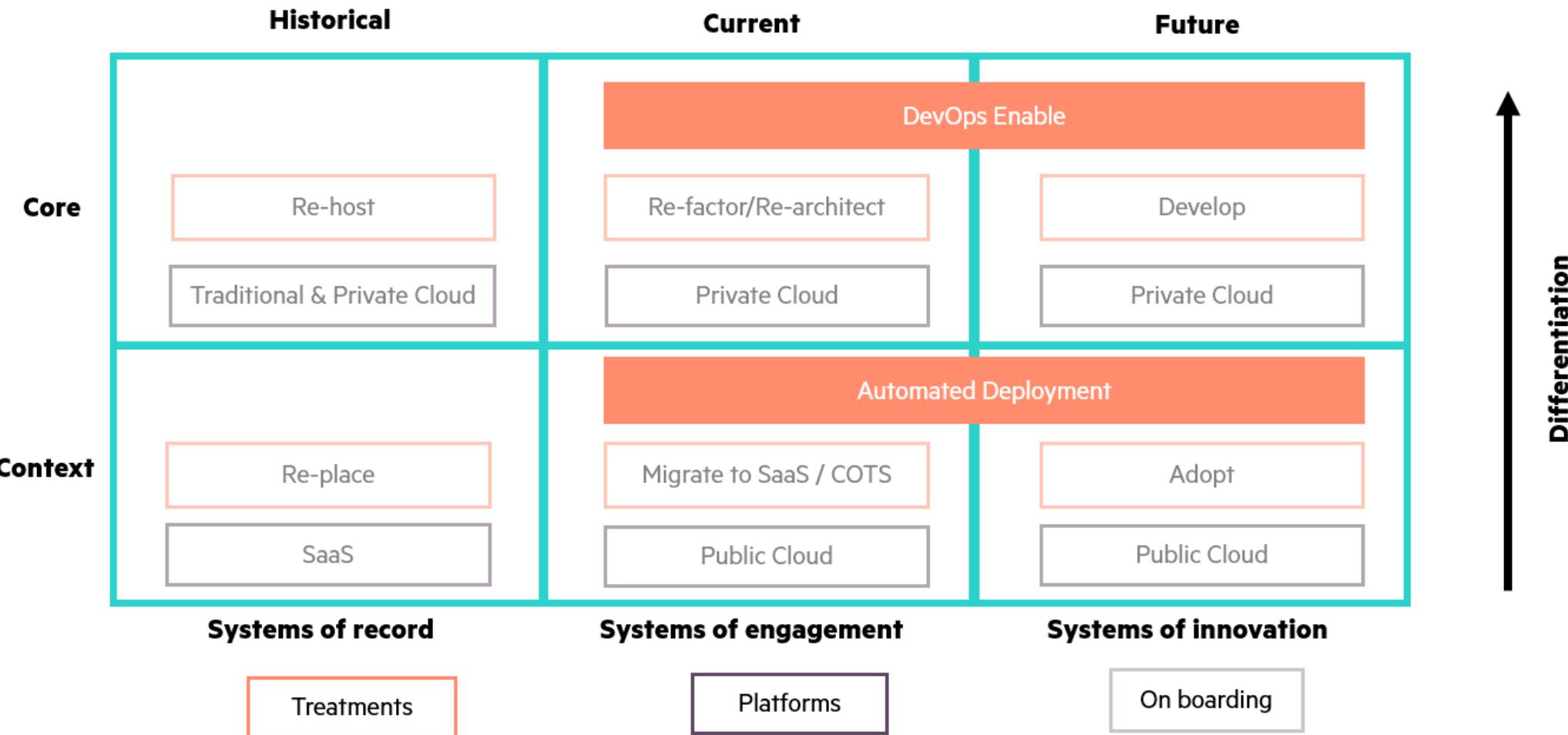
```
[INFO] --- robotframework-maven-plugin:1.2:run (robot) @ my-basic-
-----
1 Acceptance
-----
2 Acceptance.Basic Test
-----
3 Basic Test
-----
4 Acceptance.Basic Test
5 1 critical test, 1 passed, 0 failed
6 1 test total, 1 passed, 0 failed
-----
7 Acceptance
8 1 critical test, 1 passed, 0 failed
9 1 test total, 1 passed, 0 failed
-----
```

Core Tools required

Concept	Description	Sample Tools		
Source Code Management	<p>Central Repository for all code</p> <p>All code (application, ops, qa) checked-in</p> <p>Version control, labelling and release management</p>	Source Code Mgmt	git	Oracle Developer Cloud
Continuous Build & Integration	<p>Centralized system used to schedule and control jobs</p> <ul style="list-style-type: none"> • Build • Integration and Deployment • Test jobs 	Hudson	Jenkins	Oracle Developer Cloud
Infrastructure as Code	<p>Frameworks to enable the creation of automation to</p> <ul style="list-style-type: none"> • Provision VMs (OS Images, CPU, Mem...) • Storage • Ability to provision software, patches, dependencies dynamically 	Chef	Puppet	
Cloud Infrastructure	<p>Cloud provides for dynamic provisioning of Resources</p> <ul style="list-style-type: none"> • VMs • Storage • PaaS Services (DBCS) 	Oracle Cloud	Oracle Cloud Machine	
Test Automation	Test Automation Frameworks	Junit	Selenium	

Where to use DevOps

Analyze your portfolio

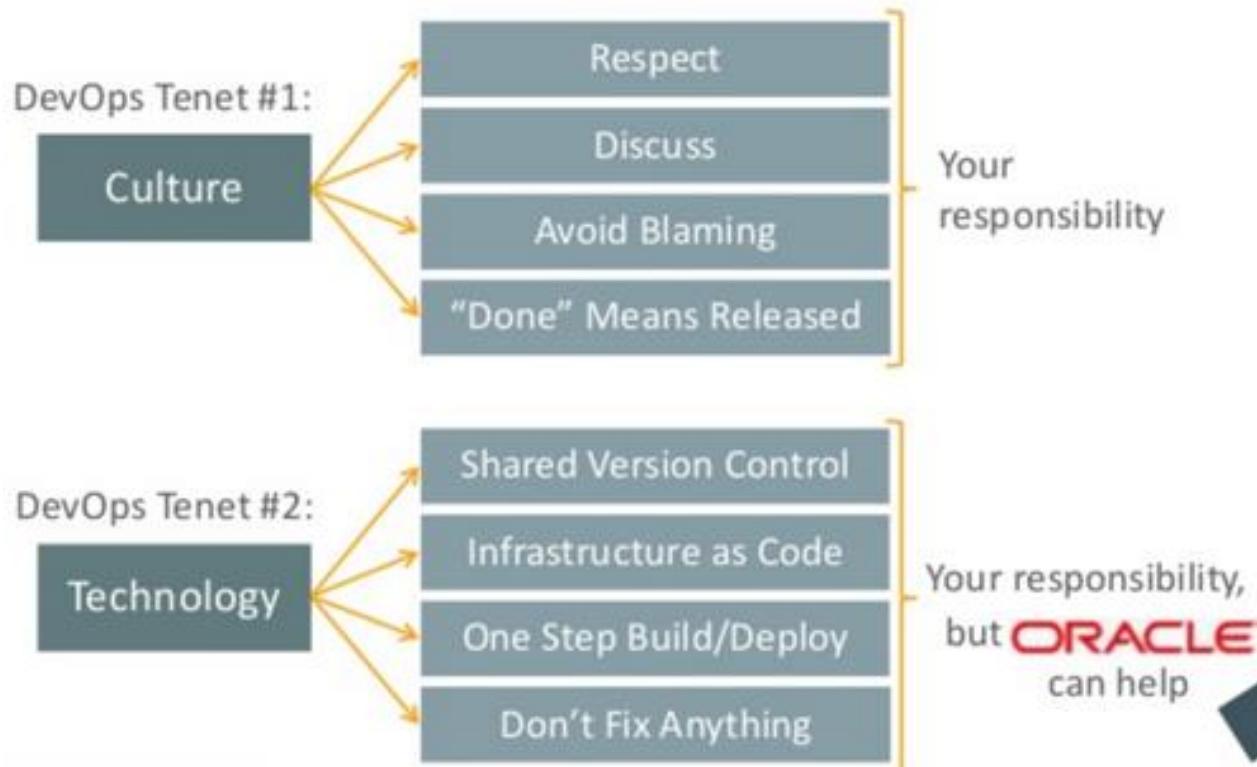


Oracle Cloud

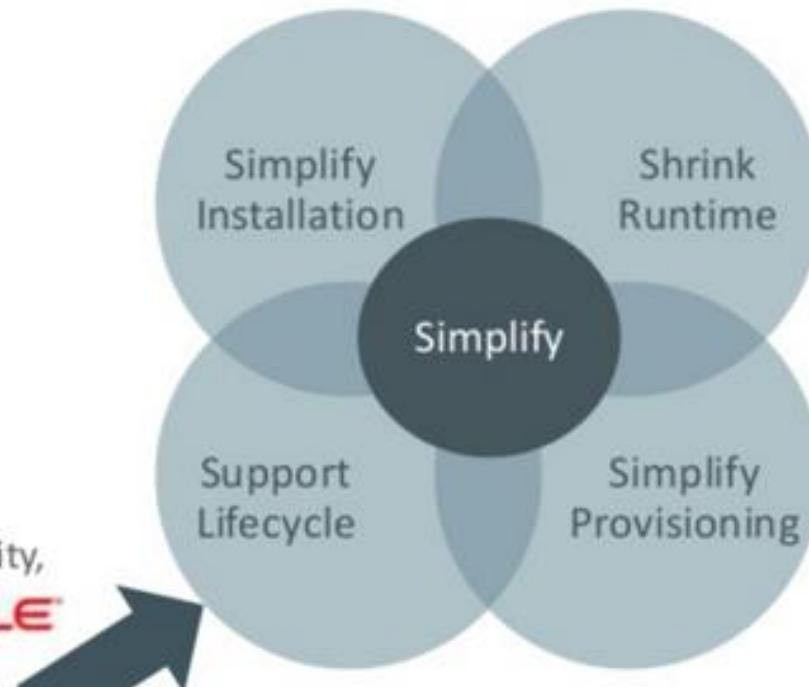


Oracle can help you Lead change in your Organization

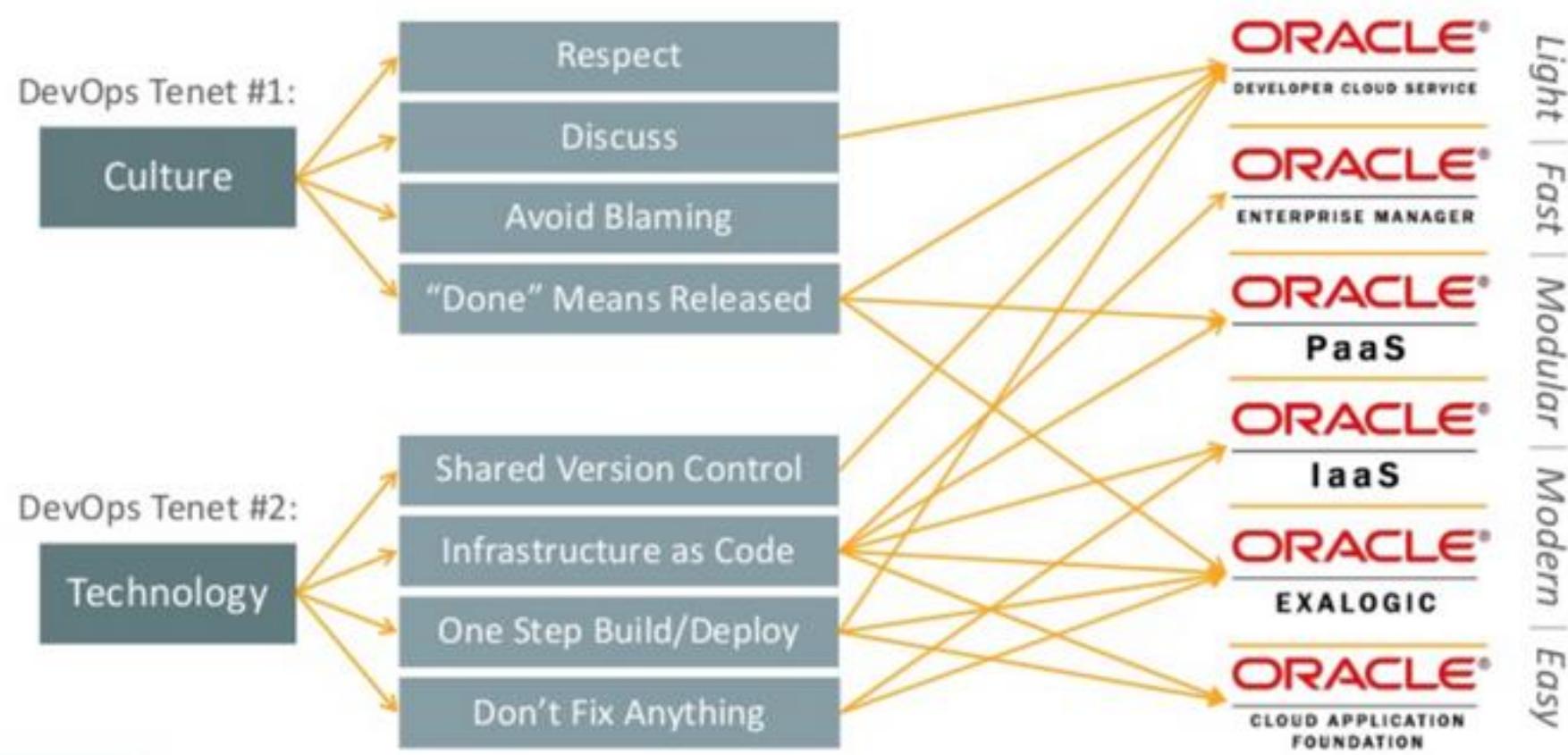
But real change begins with you



How Oracle Can Help with DevOps



Oracle Products support DevOps



Oracle Developer Cloud Service – What's In It

- Version Management – Git
- Build Automation
 - Ant, Maven, Gradle, npm, Grunt, Bower, Gulp, Command line
- Continuous Integration - Hudson
- Issues Tracking
- Code review
- Deployment automation
- Agile process management
- wiki
- Activity stream



Development Experience

Oracle Developer Cloud Service

- Effortless Project Management
- Teamwork Through Integrated Tools
- What You Need, Before You Need It
- From Just an Idea, to Product Release

The screenshot displays the Oracle Developer Cloud Service interface. At the top, the navigation bar includes links for Home, Code, Snippets, Merge Requests, Issues, Build, Deploy, Wiki, Agile, and Administration. A search bar labeled "Search Activities" is also present. The main content area shows a timeline of activities for January 2016, with Shay Shmeltzer's comments on Feature 1 and Defect 7. A prominent message box announces "Oracle DevCS 16.1.5 is now available". On the right, the "TEAM" section lists three members: Shay Shmeltzer (Owner), Alex Admin, and Clara Coder.

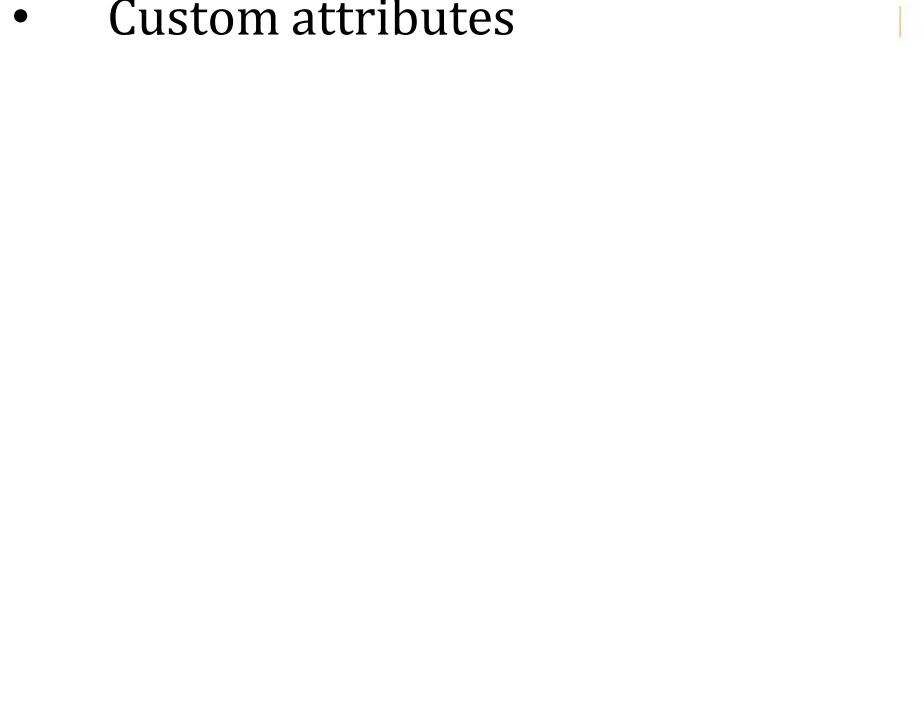
Project Management

- Team members
- Activity stream
- Usage tracking
- Repositories
- Custom attributes

The screenshot shows the Oracle Developer Cloud Service interface for the 'CustomerTracker' project. The top navigation bar includes links for Home, Code, Snippets, Merge Requests, Issues, Build, Deploy, Wiki, Agile, and Administration. A search bar labeled 'Search Activities' is also present. On the left, a sidebar displays a timeline of activity from January 2016, showing multiple comments and updates from user 'Shay Shmeltzer' regarding Feature 1 and Defect 7. A prominent yellow banner at the top right announces 'Oracle DevCS 16.1.5 is now available'. On the right side, there is a 'TEAM' section with tabs for All, Owners, and Members, and a 'New Member' button. The 'Members' tab is selected, listing three users: Shay Shmeltzer (Owner), Alex Admin, and Clara Coder, each with their email addresses and profile icons.

Requirements/Issue Tracking

- Create Requirements/Bugs/ERs
- Assign to team members and sprints
- Custom attributes



Eclipse Issue View

The screenshot shows the Oracle Developer Cloud Service interface. At the top, there's a navigation bar with links for Home, Code, Merge Requests, Issues, Develop, Build, Deploy, Wiki, and Administration. The user is logged in as 'alex.admin'. Below the navigation is a search bar labeled 'Search Issues'.

The main area is titled 'Track Issues' and features two tabs: 'Advanced Searches' and 'Recently changed'. The 'Recently changed' tab is selected and highlighted with a blue border. It lists three recent issues:

ID	Summary	Component	Status	Owner
41	Create web methods for CRUD	Default	Unconfirmed	
2	Update the index.jsp file and add a setContact method in ContactsService.java	Default	Resolved	don.devel
1	Create a branch and push all index.jsp updates to the new branch	Default	Resolved	Alex Admin

Below the table, there's a 'My Searches' section and a 'Page 1 of 1 (1-3 of 3 items)' indicator. To the right of the table, there's an 'Outline' view showing a tree structure of tasks and issues, and a 'Task List' view showing a list of tasks. A red arrow points from the text 'Eclipse Issue View' to the 'Task List' view.

Agile Process Management

- Create dashboard
- Manage issues backlog
- Manage development sprints
- View team/tasks status
- Reports

The screenshot shows a web-based Agile management tool interface. At the top, there's a navigation bar with links: Home, Code, Snippets, Merge Requests, Issues, Build, Deploy, Wiki, Agile (which is selected), and Administration. Below the navigation is a header for "Q1Progress". On the right side of the header are buttons for Backlog, Active Sprints, and Board, with "Board" currently selected. A sub-header indicates the sprint: "Sprint: JanSprint - 1/7/2016 4:06 PM - 1/21/2016 4:06 PM" and a "Complete Sprint" button. Below this, there are dropdown menus for sorting: "Sort Issues by: Priority" and "Sort Swimlanes by: User Name". The main area displays a board with three columns: To Do, In Progress, and Completed. The "To Do" column has 2 items, the "In Progress" column has 1 item, and the "Completed" column has 4 items. The board is organized into swimlanes based on team members: clara.coder, dana.singleterry@oracle.com, and shay.shmeltzer@oracle.com. Each swimlane contains tasks with their descriptions, priority levels (indicated by green and yellow bars), assignees (represented by icons), and status (e.g., UNCONFIRMED, ASSIGNED, RESOLVED : FIXED). For example, under clara.coder, there is a task "Feature 8" with priority 2 and status UNCONFIRMED. Under dana.singleterry@oracle.com, there is a task "Defect 2" with priority 2 and status ASSIGNED.

Source Code Management

- Git repositories
- Branch, tag, merge
- Web interface
- View changes online
- Accessible from any Git client
- External repositories integration (for example GitHub)
- Snippets – for reusable code

The image shows two screenshots of the Oracle Developer Cloud Service interface. The top screenshot displays a repository named 'summitADFApp' with a commit history. The commit history shows the initial check-in of files like .adf / META-INF, Model, src / META-INF, ViewController, build.properties, and build.xml. The bottom screenshot shows a detailed view of a file named 'index.jsf' with a diff view comparing two versions. The diff highlights changes in lines 10 and 11, specifically adding new ADF components like facets and panels.

```
+1 -1 index.jsf ViewController/public_html
@@ -7,7 +7,7 @@
    id="pt1">
<af:facet name="center">
<af:panelTabbed id="pt2">
@@ -10,12 +10,12 @@
    <af:showDetailItem text="Welcome" id="sd1"/>
    <af:showDetailItem text="Hello" id="sd1"/>
    <af:showDetailItem text="Summit Management" id="sd12" disclosed="true"
                      stretchChildren="first">
@@ -11,12 +11,12 @@
    <af:region value="#{bindings.customerTaskFlowDefinition1.regionModel}">
```

Code Reviews

- Request code review
- Invite team members
- Comment on Code
- Accept / Reject / Iterate Reviews
- Merge Code
- Merge Conflict Resolution

The image displays two screenshots of the Oracle Developer Cloud Service interface.

Screenshot 1: Merge Requests

This screenshot shows the "Merge Requests" page. The navigation bar includes Home, Code, Merge Requests (which is selected), Issues, Develop, Build, Deploy, Wiki, and Administration. A search bar at the top right is labeled "Search Requests". The main area is titled "Merge Requests" and contains tabs for "Standard Searches" and "My Requests" (which is selected). On the left, there are filters for "All Open Requests", "Assigned To Me", "Completed", and "Closed". A table lists two merge requests:

ID	Summary	Status	Repository	Branch	Submitter	Created
21	Sample merge request	OPEN	contacts.git	006	Alex Admin	1/16/2015
3	Review the code	COMPLETED	contacts.git	NewBranch	Alex Admin	1/12/2015

Screenshot 2: Code Review

This screenshot shows a code review for a commit. The title is "hello request | Commit 2b46258". It shows a message from "shay.shmeltzer" dated "April 21 2015 3:37 PM -0700" with the note "updated welcome message to hello". Below this is a diff view for the file "index.jsf". A modal window titled "Add Comment" is open, containing the text "Need to use meaningful names for variable". At the bottom of the modal are "OK" and "Cancel" buttons.

```
+1 -1 index.jsf ViewController/public_html
+1 -1 index.jsf ViewController/public_html
 7 7 @@ -7,7 +7,7 @@
 8 8         id="pt1">
 9 9             <af:facet name="center">
10 10                 <af:panelTabbed id="pt2">
11 11                     <af:showDetailItem text="Welcome" id="sd11"/>
12 12                     <af:showDetailItem text="Hello" id="sd11"/>
13 13                     <af:showDetailItem text="Summit Management" id="sd12" disclosed="true"
                           stretchChildren="first">
                           <af:region value="#{bindings.customertaskflowdefinition1.regionModel}">
```

Project Builds

- Maven
- Ant
- Gradle
- Node.JS – npm, grunt, bower, gulp
- Dashboard
- Logs and Audit

The screenshot displays a project build interface with two main sections: a top-level dashboard and a detailed configuration page.

Top Dashboard:

- Header:** Home, Code, Snippets, Merge Requests, Issues, Build, Deploy, Wiki, Agile, Administration.
- Breadcrumbs:** < Jobs Overview | ContactsFeatureBuild
- Buttons:** Build Now, Configure, Disable, Delete.
- Icons:** Console, Changes, Git Logs, Audit.
- Description:** Maven POM build file generated by Oracle Enterprise Pack for Eclipse.
- Artifacts of Last Successful Build:** CustomerTracker-1.0.0-SNAPSHOT.war (all files in zip).
- Notifications:** On (selected), Off, CC Me.
- Build History:** A table showing five successful builds (#7 to #3) with details like date, time, duration, and console link.
- Build Trend:** A line chart showing build duration in seconds over time, with a legend indicating "Success".

Bottom Configuration:

- Header:** < Jobs Overview | ContactsFeatureBuild | Configure build job
- Tabs:** Main, Build Parameters, Source Control, Triggers, Environment, **Build Steps** (selected), Post Build, Advanced.
- Configure Build Steps:** Invoke Maven 3.
- Invoke Maven 3:**
 - Maven 3: (Bundled)
 - Goals: clean install deploy
 - Properties: (empty)
 - POM File: CustomerTracker/pom.xml
- Add Build Step:** A dropdown menu listing options: Execute shell, Invoke Ant, Invoke Maven 2 (Legacy), Invoke Maven 3, Invoke Gradle, Invoke NodeJS, Copy Artifacts.

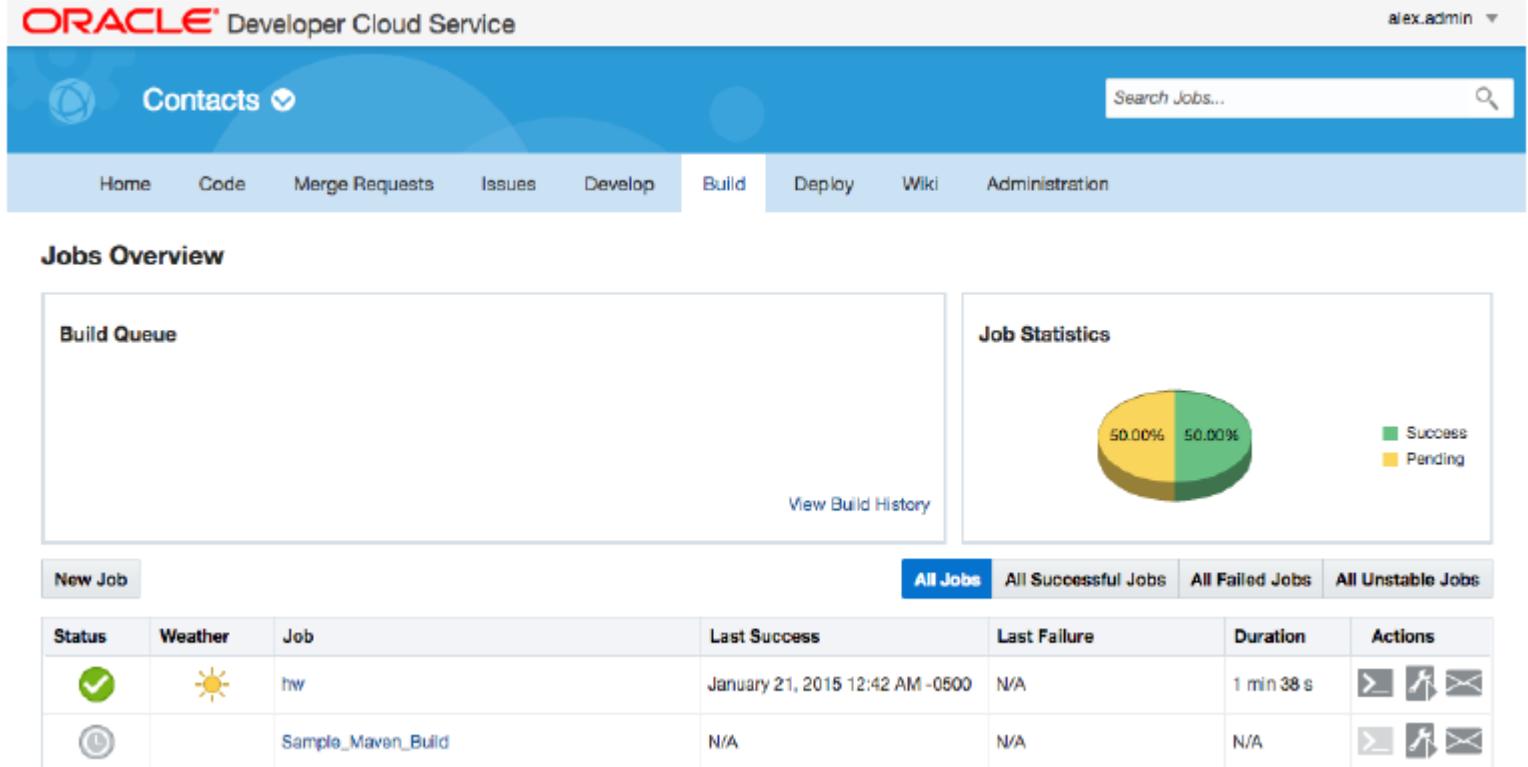
Deployment Automation

- Create deployment configurations
- Start/Stop a deployment
- Redeploy/Un-deploy applications
- In the cloud or on premise deployment

The screenshot displays two views of the Oracle Developer Cloud Service interface. The top view shows the 'Deployments' section under the 'Contacts' tab, listing a successful deployment to 'hw / 1 / target/helloworld.war' with logs for virus-scan, whitelist, wls-appc, and redeploy. The bottom view shows the 'New Deployment Configuration' dialog for the 'CustomerTracker' project, where a configuration named 'QAServer' is being created for an application named 'QAServer' on a deployment target named 'oracletemplates / jcs1'. The deployment target dropdown lists options like 'Java Service - SaaS Extension' and 'Application Container Cloud'.

Continuous Integration

- Hudson
- Automate
- Triggers
- Schedule
- Dashboard



The screenshot shows the Oracle Developer Cloud Service interface. At the top, there's a navigation bar with links for Home, Code, Merge Requests, Issues, Develop, Build (which is currently selected), Deploy, Wiki, and Administration. A search bar labeled "Search Jobs..." is also present. The main content area is titled "Jobs Overview". It features two main sections: "Build Queue" and "Job Statistics". The "Build Queue" section has a button to "View Build History". The "Job Statistics" section contains a pie chart showing 50.00% Success and 50.00% Pending. Below these sections is a "New Job" button and a table with columns for Status, Weather, Job, Last Success, Last Failure, Duration, and Actions. The table has two rows: one for a job named "hw" which last succeeded on January 21, 2015 at 12:42 AM, and another for a job named "Sample_Maven_Build" which has N/A for both success and failure dates.

Status	Weather	Job	Last Success	Last Failure	Duration	Actions
		hw	January 21, 2015 12:42 AM -0500	N/A	1 min 38 s	
		Sample_Maven_Build	N/A	N/A	N/A	

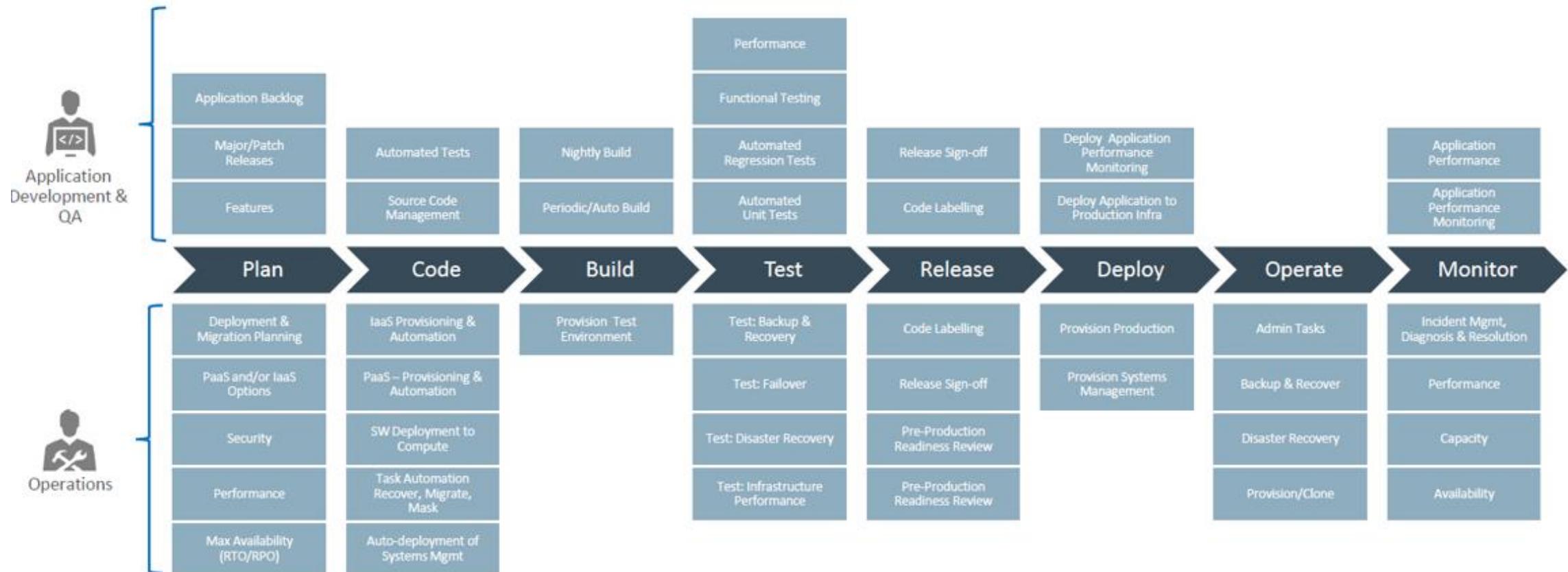
Wikis

- Share information
- Attachment support
- Wiki markup of choice

The screenshot shows the Oracle Developer Cloud Service interface. At the top, the navigation bar includes the ORACLE logo, the project name "summitADFAApp", a search bar, and a user account. Below the header, a secondary navigation bar features links for Home, Code, Merge Requests, Issues, Build, Deploy, Wiki (which is currently selected), and Administration. The main content area is titled "Edit Page" and displays a "Page Title" input field containing "Requirements Gathering". Below the title, there are tabs for "Page Text (Markdown)" (selected), "Preview", "Attachments", and "Access Rights". The preview section contains the heading "Interview with Manager" and the text "basic process steps" followed by a bulleted list: "• Get phone call", "• Locate record for customer", and "• Go through sales script".

Merger of disciplines

Iterative Planning, Development and Release



Q & A



Thank you

Sai Penumuru

Contacts – ps.janakiram@gmail.com ; twitter - @sai_penumuru
