



Freedom as a Service

DEVOPS - THE GATEWAY TO THE API FRONTIER



Dave Pemberton, Software AG

API, Integration & IoT Practice Manager (UK & Nordic)

Professional Services



Dave.Pemberton@SoftwareAG.com



<https://www.linkedin.com/in/pembertondave>



@sagpembo

{ API:WORLD™ }

WE ARE DATA PIONEERS

- Founded in 1969 in the belief that data will change the world
- Headquartered in Darmstadt, Germany
- Possessing a 50-years-bold heritage
#50yearsbold
- Strong partner network. For better customer-centric collaboration
- Strong alliance with competitors to ensure openness for our customers.
- Major shareholder: Software AG Foundation
- Current leadership in Industry 4.0 and Industrial Internet of Things



Present in more than
70 countries



4,700+
employees worldwide



2 Million+
Developers



Cooperation with more
than **1,350 universities**

Software AG Again Positioned in Leaders Quadrant

Gartner's 2019 Magic Quadrant for Full Life Cycle API Management



"Leaders are vendors that execute strongly and that lead and influence the market...The most distinctive attribute of Leaders is that they can address the widest variety of API use cases: mobile and multiexperience, integration using APIs, data as a service, B2B and open banking...The pulverization of the API economy into a growing number of sweeping digital transformations and the rise of the platform/ecosystem business model have put API programs on the agendas of CIOs and, in some cases, CEOs.

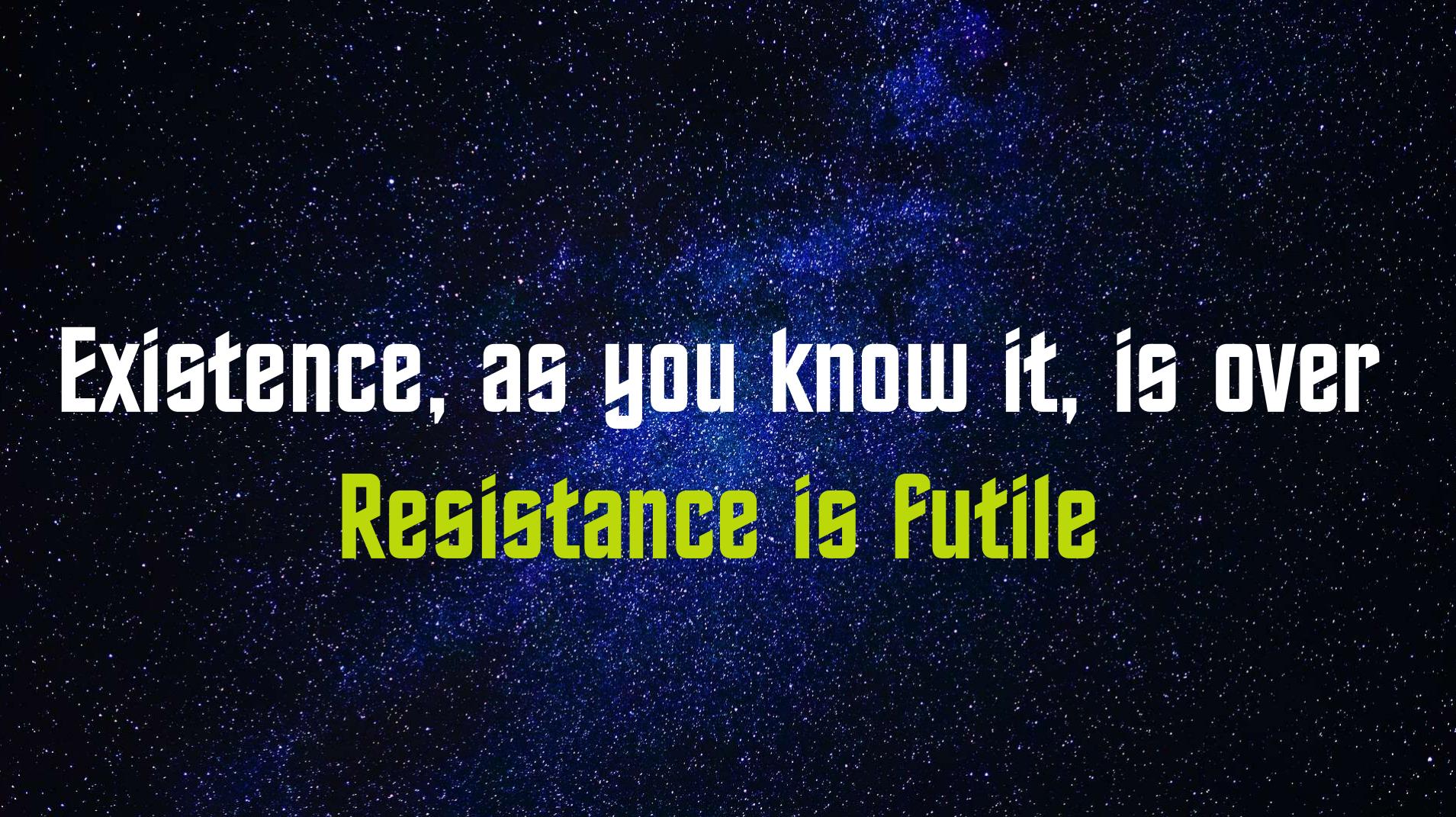
API programs frequently start small with innovative ideas, and might involve a few hackathons before they take off, but then they need to execute very quickly. Leaders have made sure their offerings can help clients thrive in this dynamic environment...by addressing digital transformations and their integration challenges head-on, with thought leadership and product functionality, and offering fully functional API management solutions.

Leaders understand the market trends that will benefit them and their clients' business strategies, frequently in the form of digital transformations. Leaders see the business potential of API programs, communicate this potential to business units, and help their clients realize it."

Source: Gartner, Inc., Magic Quadrant for Full Life Cycle API Management, Paolo Malinverno, Mark O'Neill, et al., October 9, 2019. This graphic was published by Gartner, Inc. as part of a larger research document and should be evaluated in the context of the entire document. The Gartner document is available upon request from Software AG.

Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

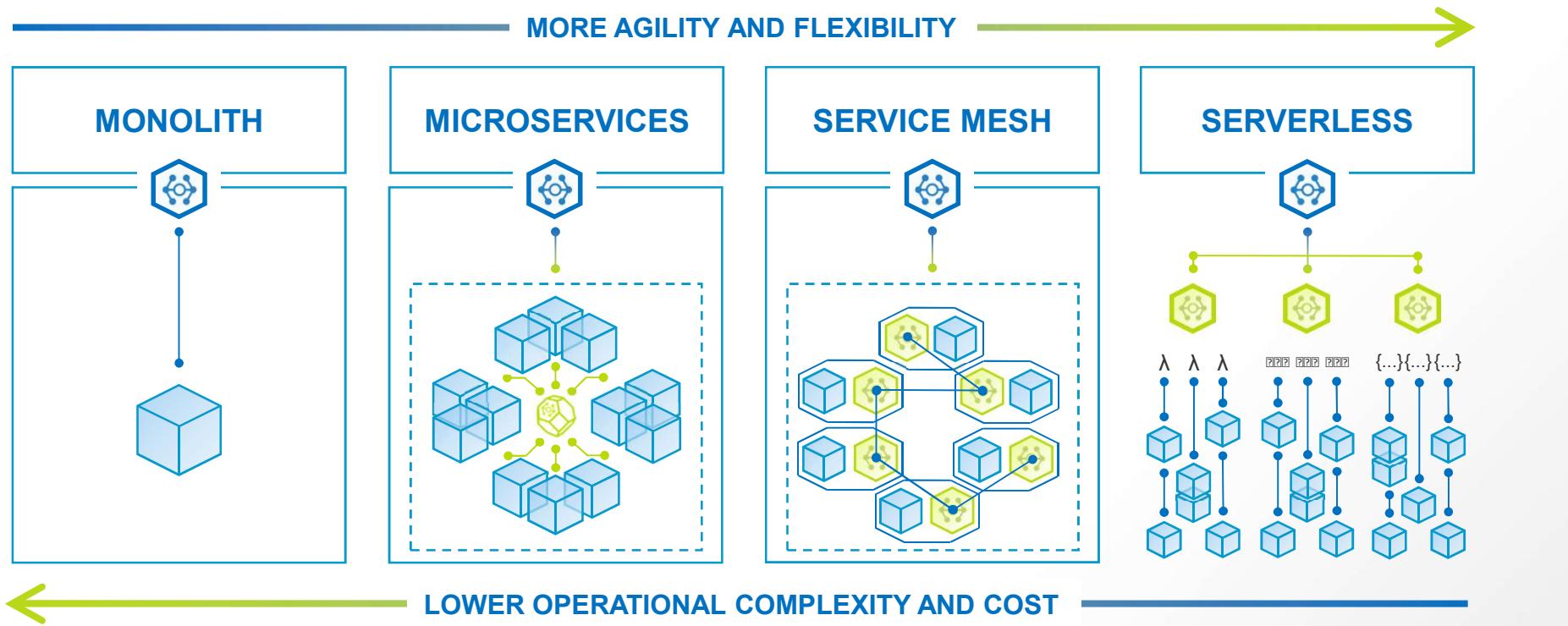




Existence, as you know it, is over

Resistance is Futile

DEPLOYMENT ARCHITECTURES



WHY DEVOPS

LIFE OF A TYPICAL DEVELOPER

```
while(!projectFinished){  
    codeComplete = writeCode();  
  
    //Creating infinite loops  
    if(!codeComplete && makeStupidMistakes){  
        drink("coffee"); //Or other energy source!  
    }  
  
    //Sleep only if possible  
    if(feelingReallyTired && !codeComplete && !nearDeadline){  
        sleep();  
    }  
  
    if(codeComplete)  
    {  
        if(!atDeadline)writeTests();  
        else projectFinished = true;  
    }  
    handleProjectManager();  
    if(atDeadline)projectFinished = true;  
}
```

- Works on multiple projects
- Often multiple languages/frameworks
- Focused on the requirements
- Testing *can often* be an afterthought
 - Especially when time pressured
 - It works... (on my machine)
 - Infrastructure – do I really need to care?

DEVOPS



WHY DEVOPS

A DEVELOPER PERSPECTIVE

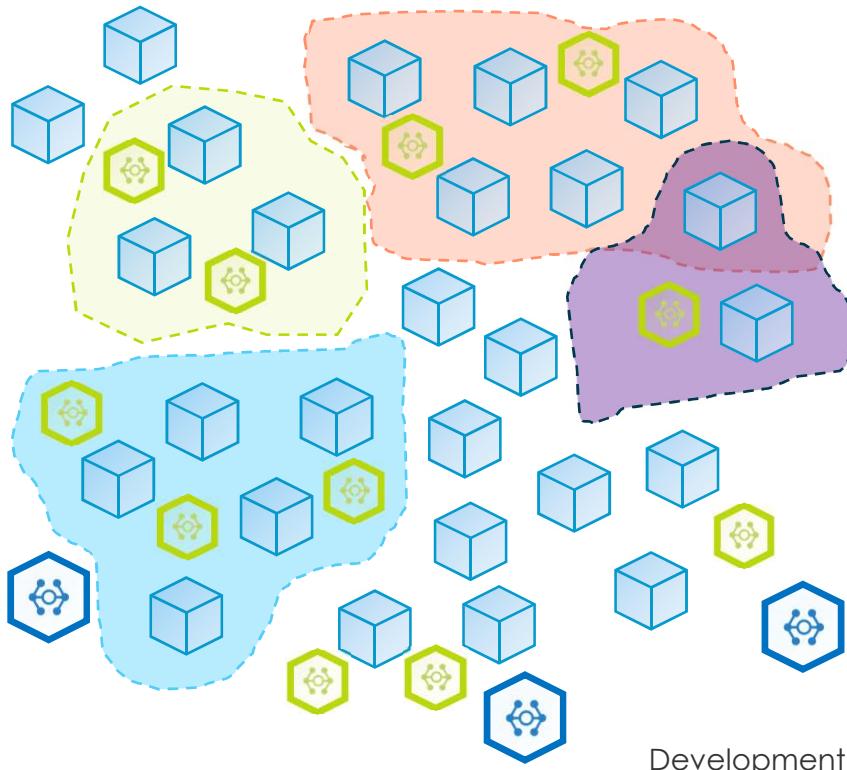
Developer

- What does my service need to do for the business?
- I want to develop in a language I like!
- How do I fix/build/change somebody else's code?
- How do I know it works?
- What code quality / style do I need to 'comply' with?
- I don't care what it has to run on – that's for operations to worry about!
- The others in the team better not break what I'm doing!
- Operations can sort the deployment so I can write more code!

I just want to write code 😊

LOOKING A LITTLE WIDER - A MILLION MOVING PARTS

HOW TO HANDLE THESE

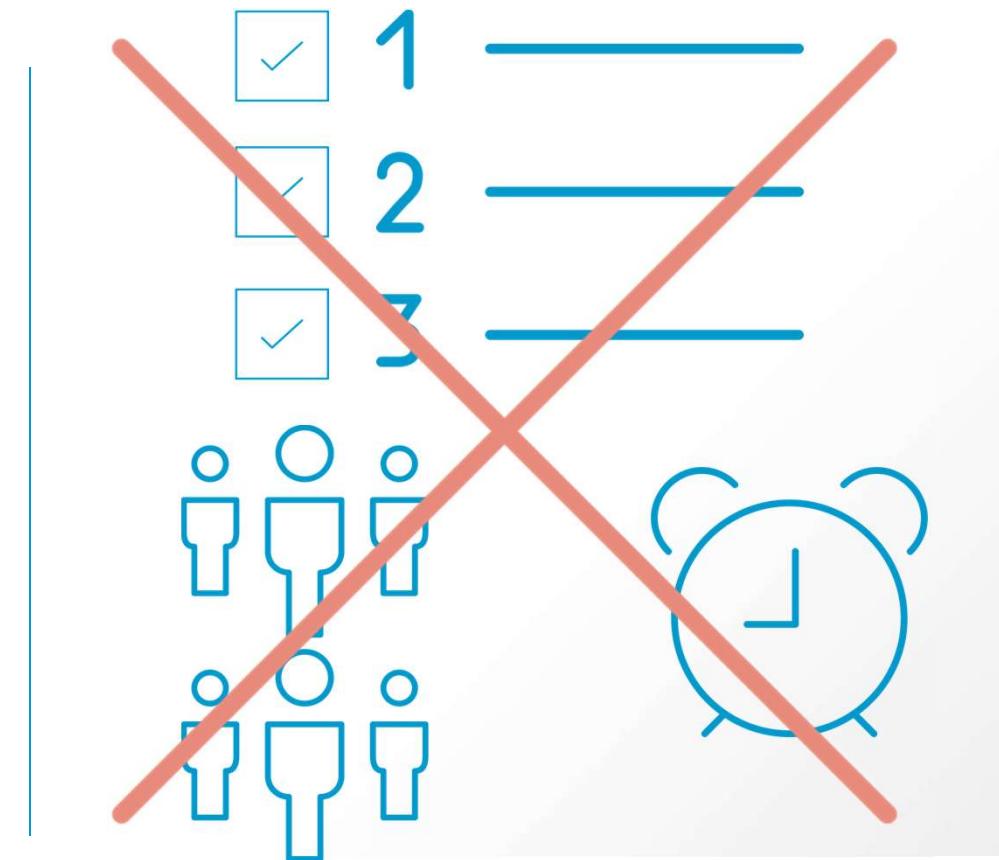
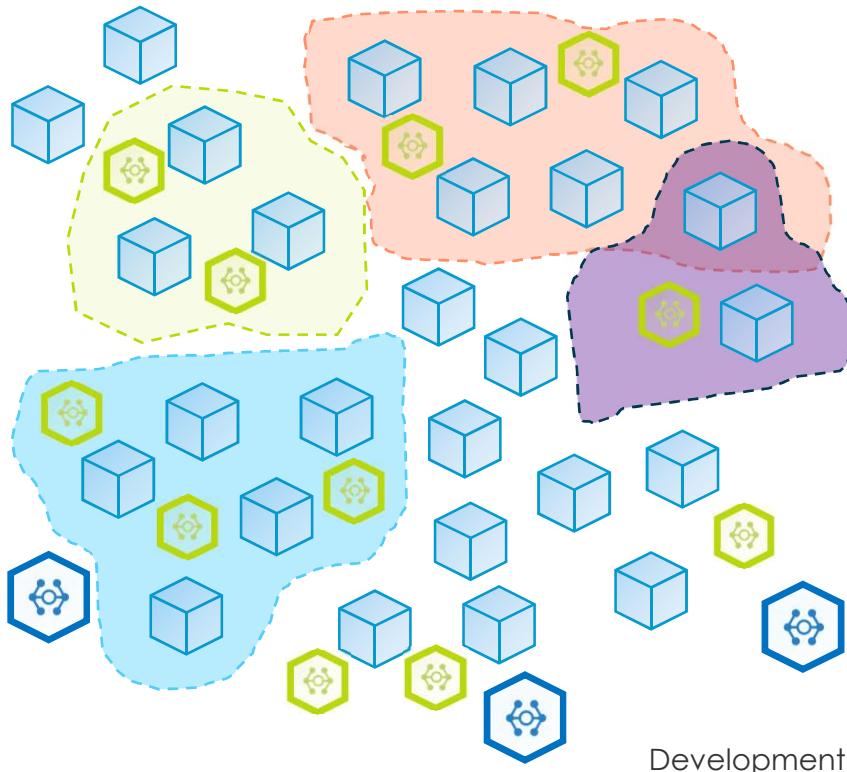


- How to build?
- How to deploy?
- What to deploy?
- Dependencies 😬?
- Hardware compatibility?
- Rollback..

Production

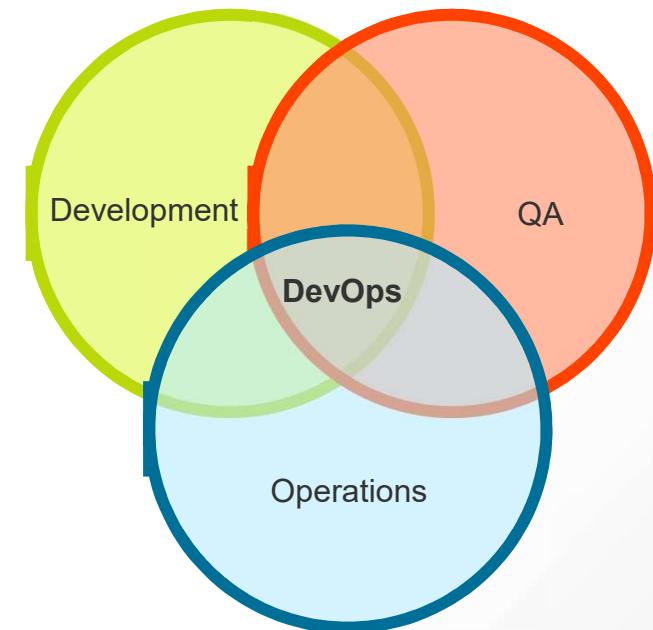
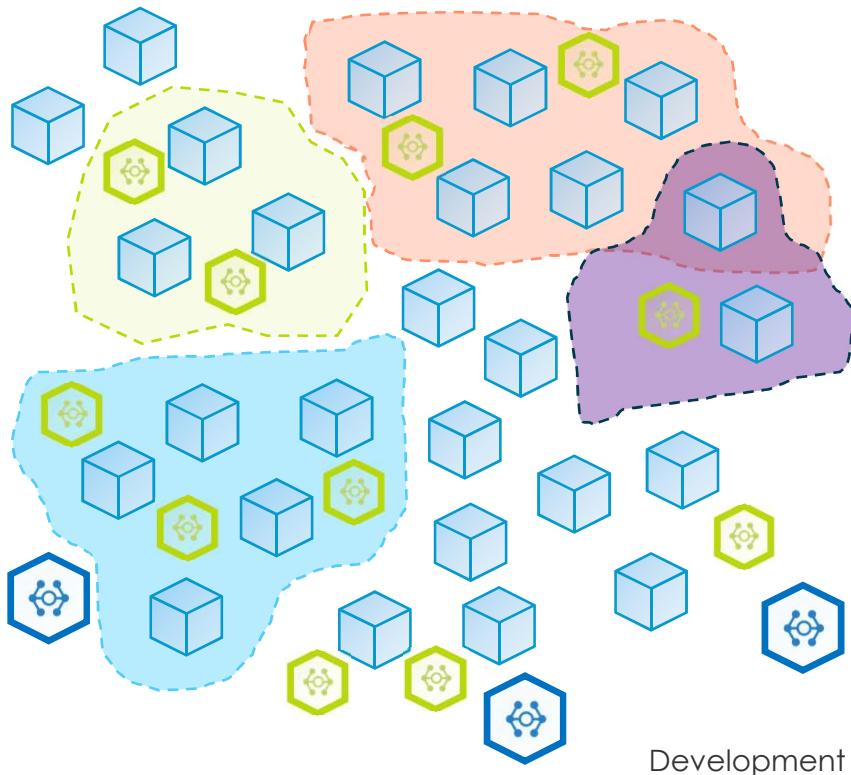
LOOKING A LITTLE WIDER - A MILLION MOVING PARTS

THE UBER TASK LIST?



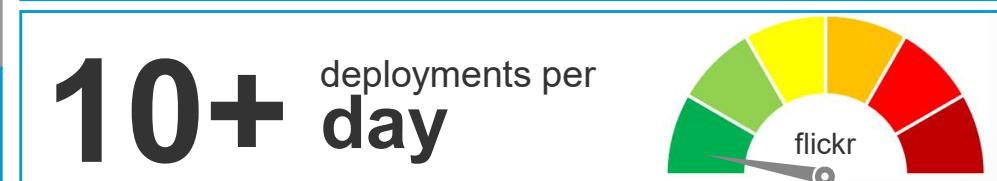
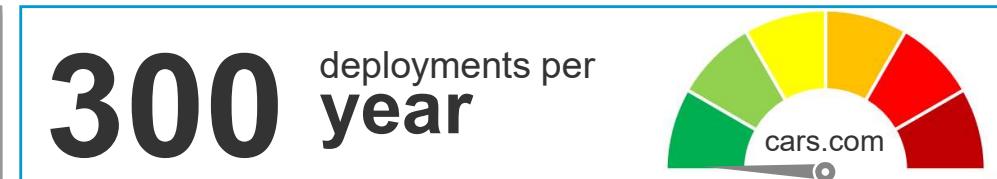
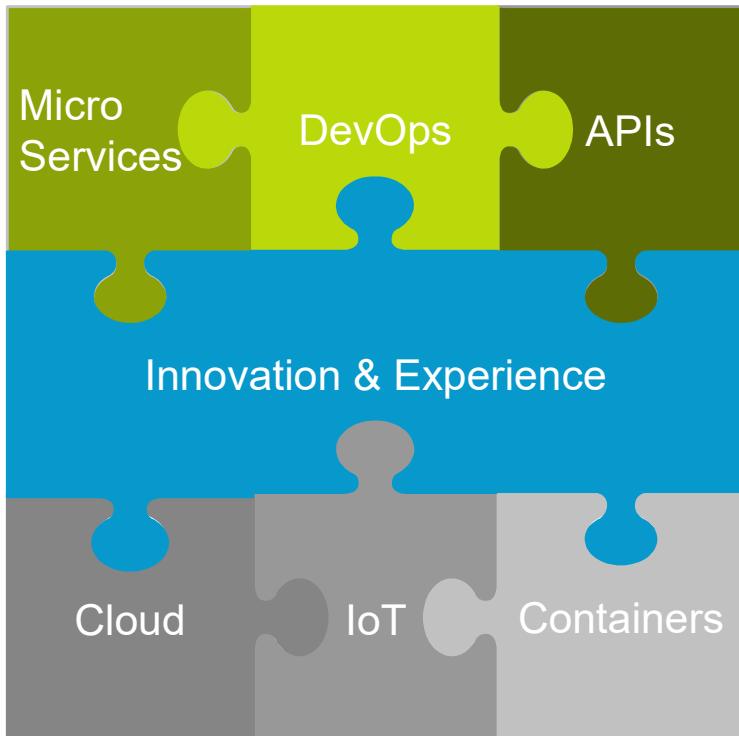
A MILLION MOVING PARTS

DEVOPS



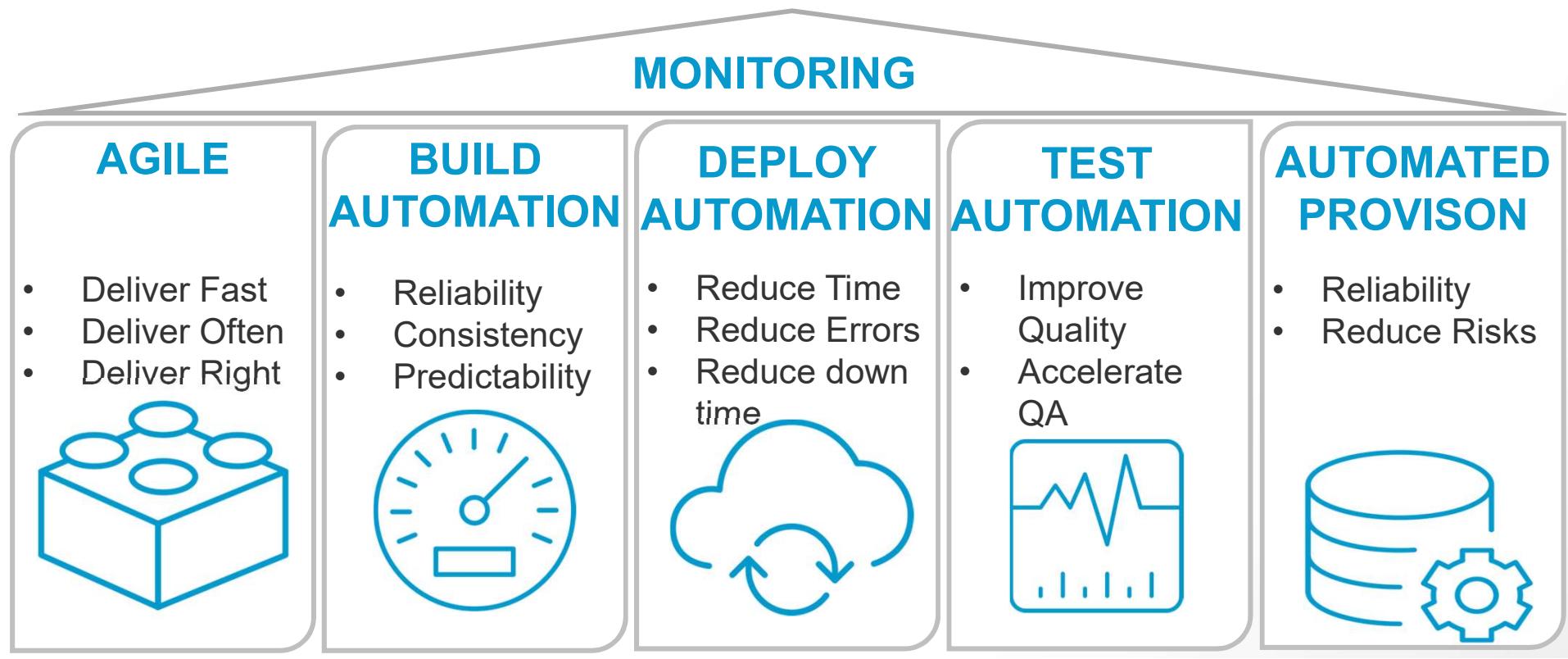
DEVPOS

I FEEL THE NEED... THE NEED FOR SPEED



DEVOPS

THE FOUNDATIONS

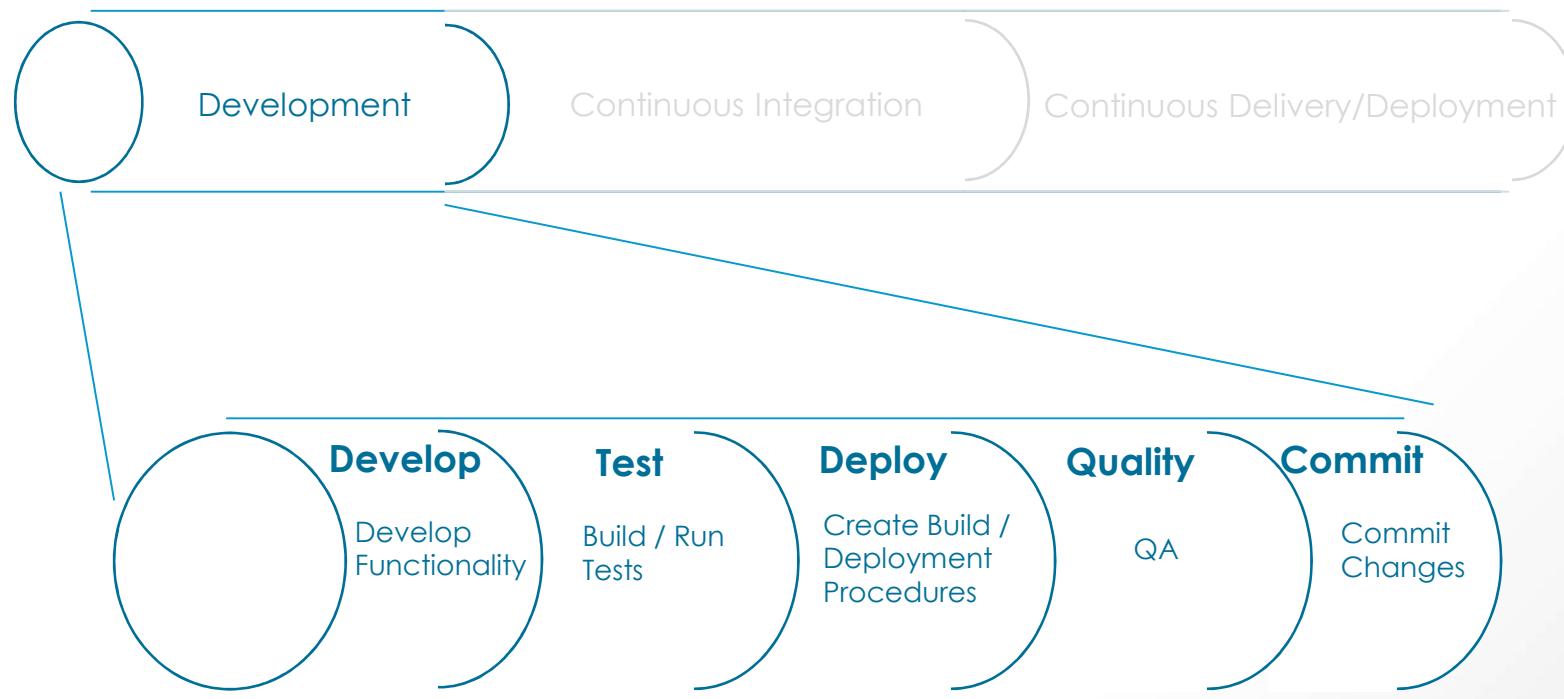


DEVOPS PIPELINE



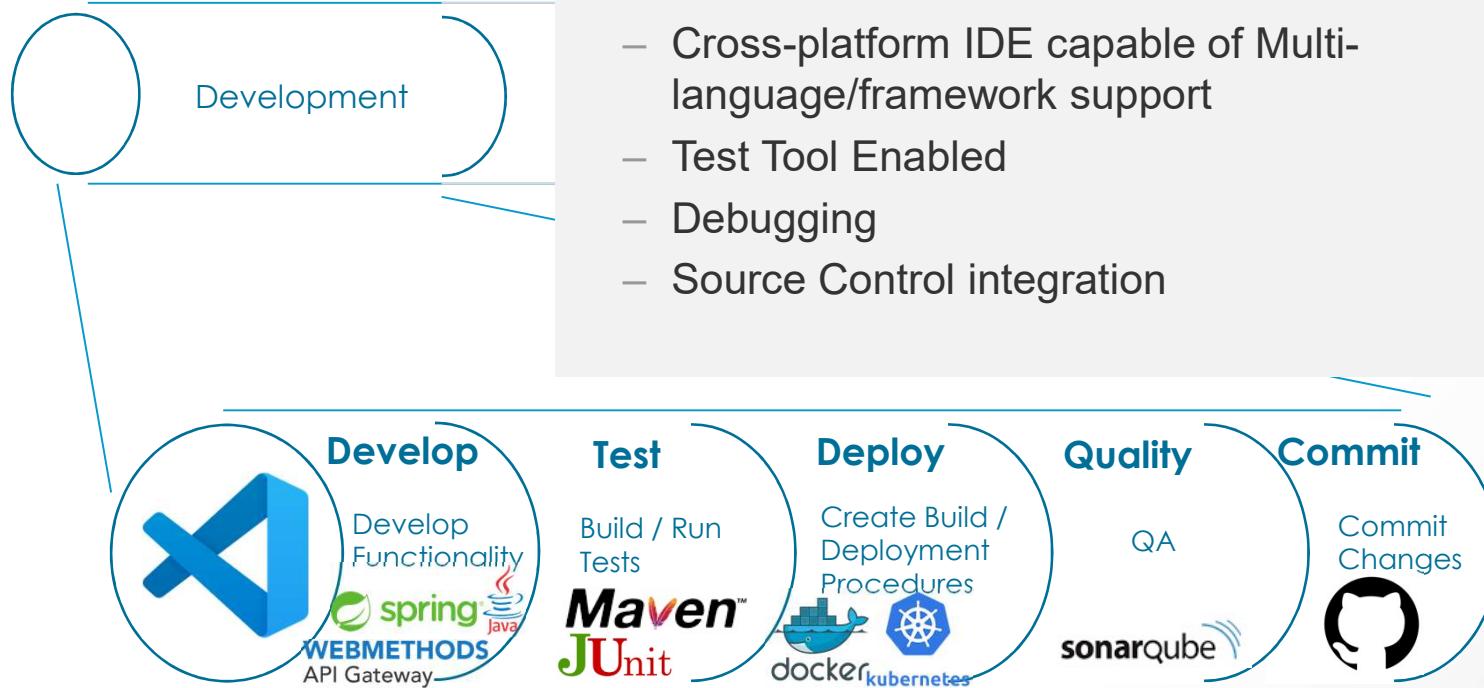
DEVOPS PIPELINE

DEVELOPMENT



DEVOPS PIPELINE

DEVELOPMENT



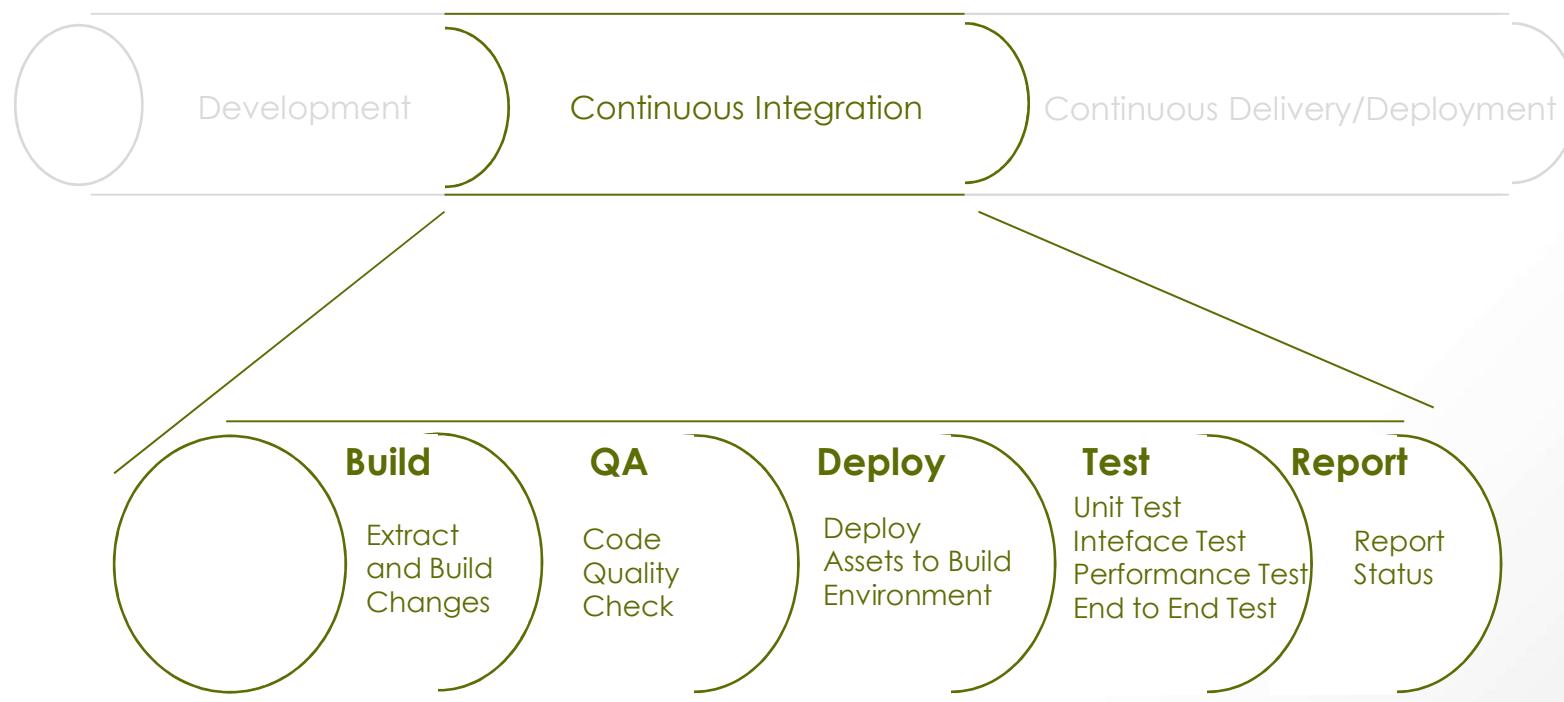
DEVOPS PIPELINE

CONTINUOUS INTEGRATION



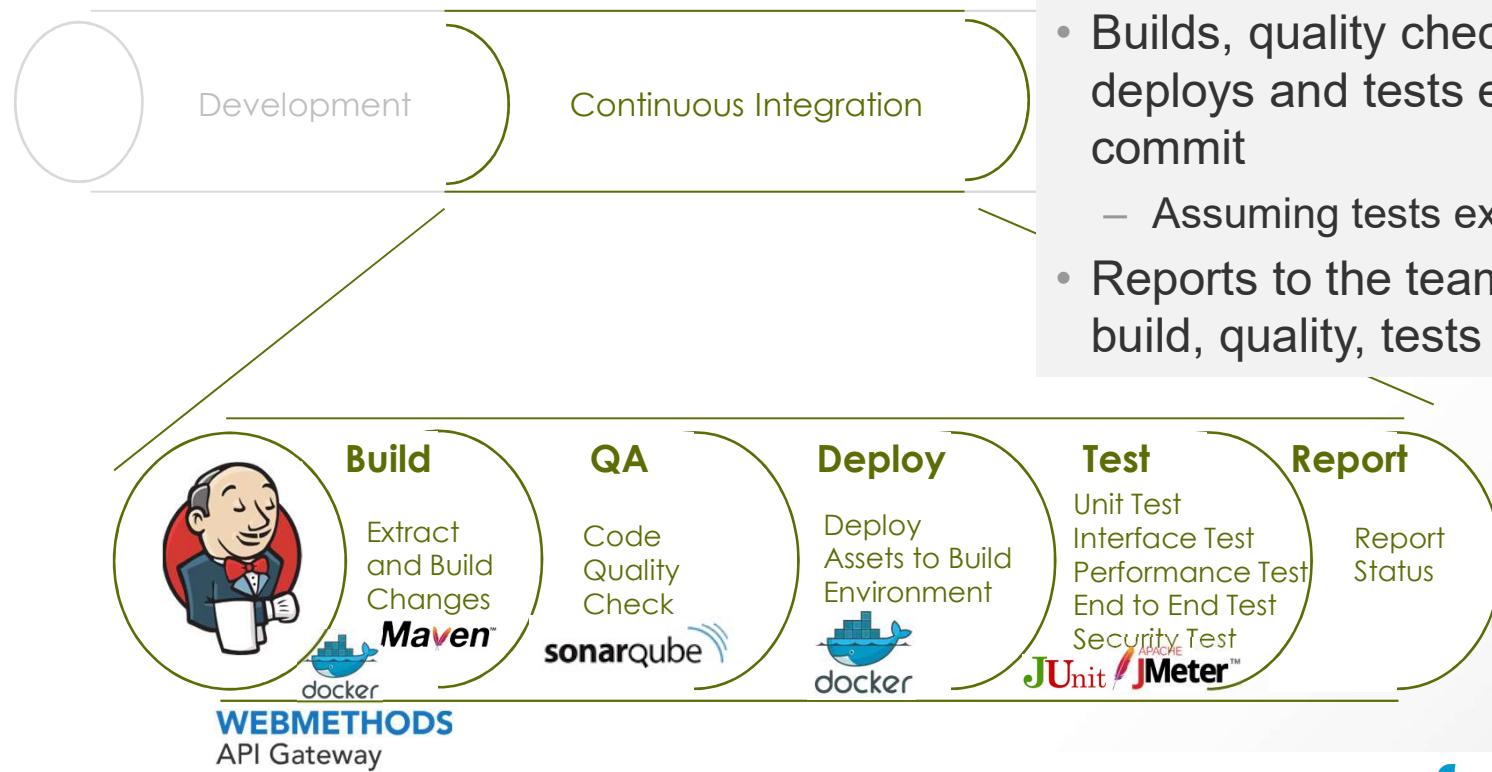
DEVOPS PIPELINE

CONTINUOUS INTEGRATION



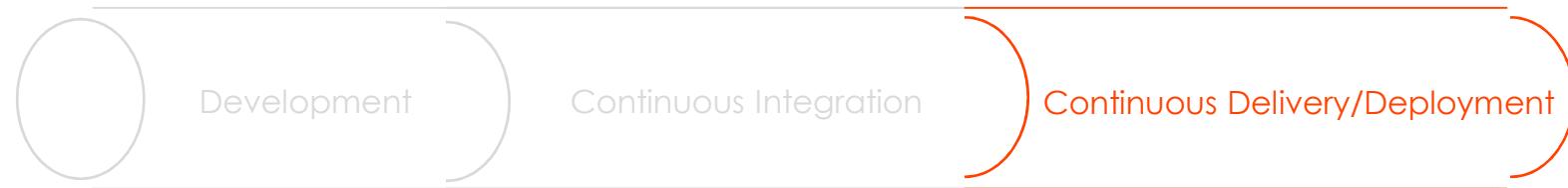
DEVOPS PIPELINE

CONTINUOUS INTEGRATION



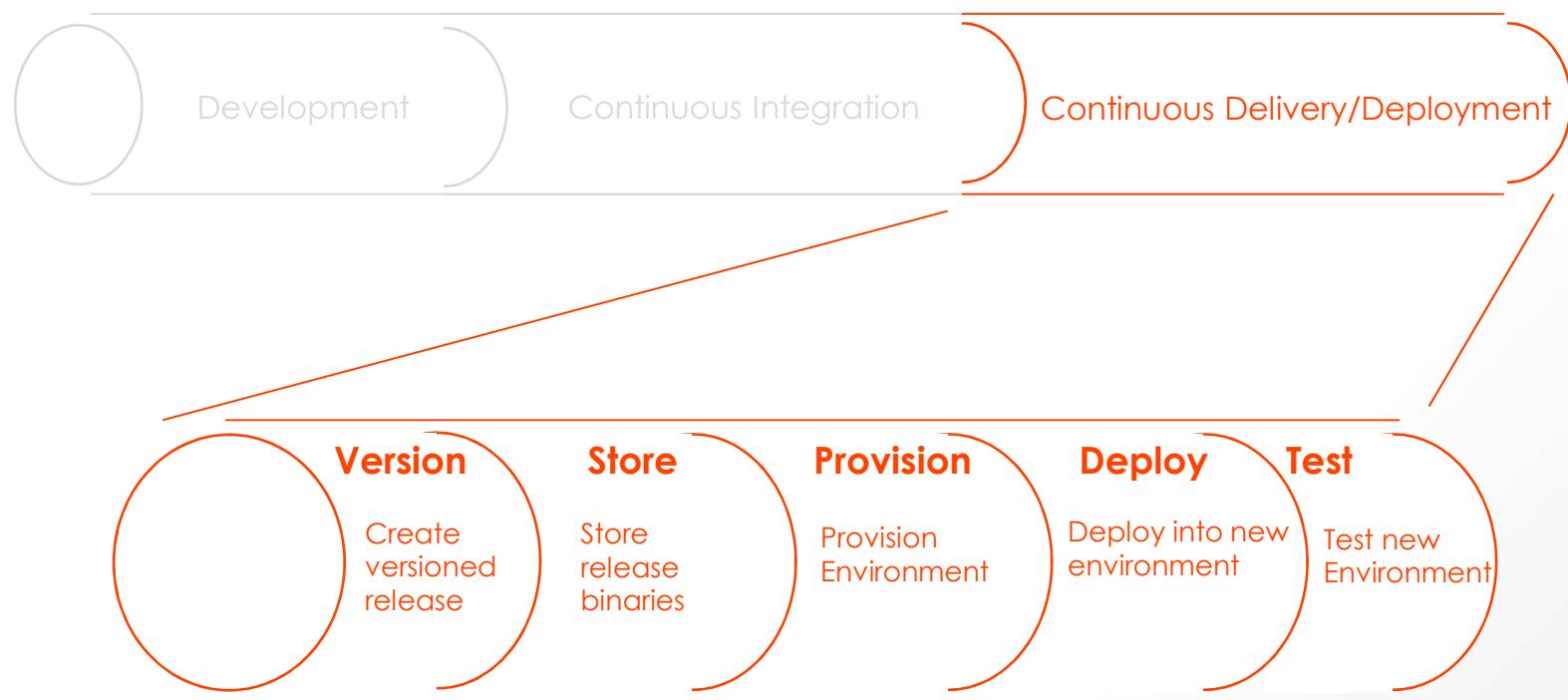
- Runs in a CI tool of choice (and there's a lot of choice)
- Builds, quality checks, deploys and tests every commit
 - Assuming tests exist ☺
- Reports to the team status of build, quality, tests

DEVOPS PIPELINE



DEVOPS PIPELINE

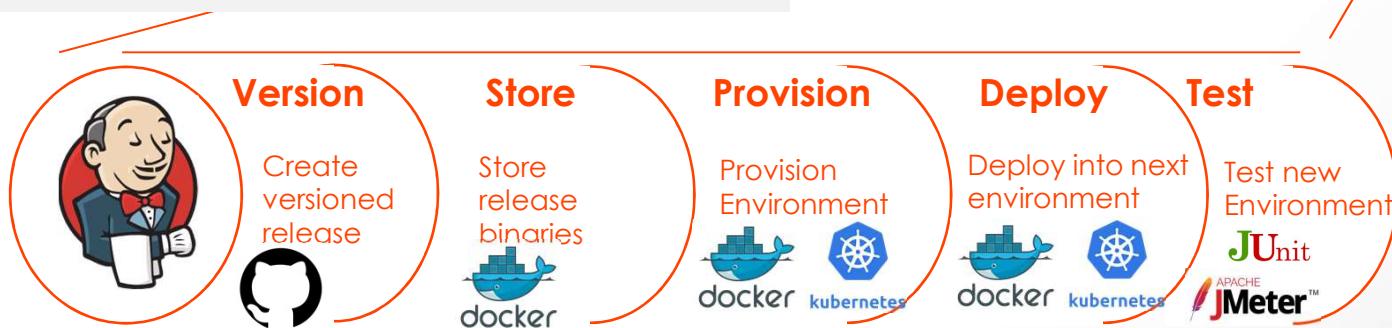
CONTINUOUS DELIVERY / DEPLOYMENT



DEVOPS PIPELINE

CONTINUOUS DELIVERY / DEPLOYMENT

- Branch/ Tag Source
- Store Binaries
- Automatically create and deploy environment with zero downtime through a rolling update.
- Test as required



API GATEWAYS



WHY API GATEWAY

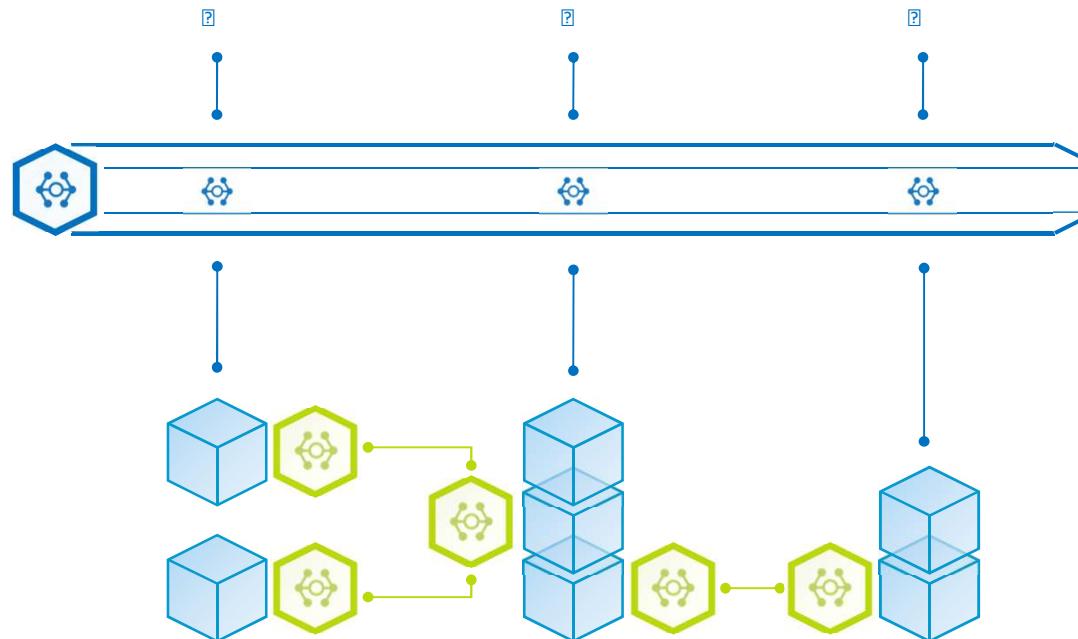
A DEVELOPER'S PERSPECTIVE

- I want my microservice to be micro
 - I don't want to create aggregated services or concern myself with latency— it's fast on my machine!
- I don't want to have to worry about security
 - I don't want to bloat out the implementation with sometimes complex security requirements (authorization/ssl)
- I don't want to have to spend my working life fixing issues caused by tight dependencies between client applications consuming my microservices
- I still want my microservice to be able to scale as intended
- I don't want to worry about load/caching requirements for my microservice

I just want to write interesting code 😊

API GATEWAY

TYPICAL ARCHITECTURE



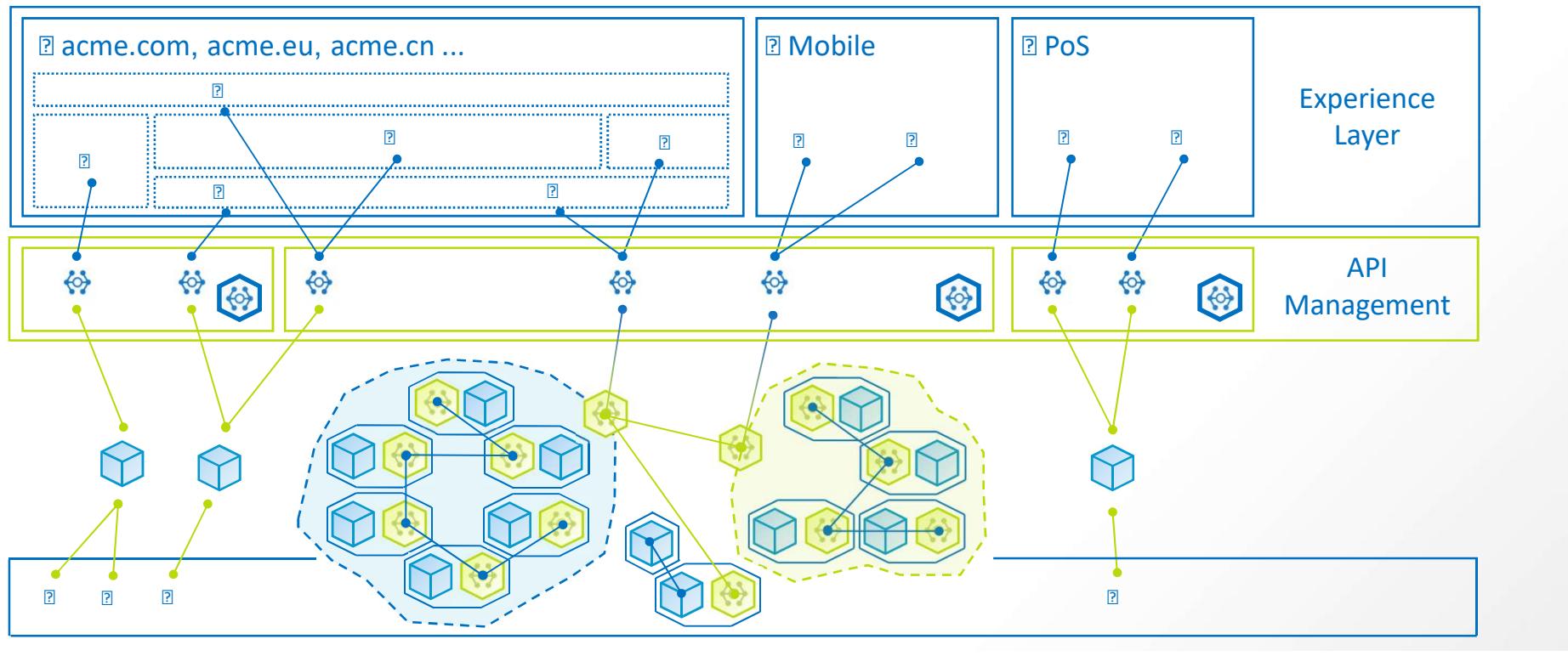
- Main / Edge gateway serves north-south traffic
- Protects and mediates public APIs



- Microgateways serve east-west traffic
- Protect and mediate private APIs
- They are light weight and only carry the policies they need right now

API GATEWAY

TYPICAL USAGE ARCHITECTURE(S)



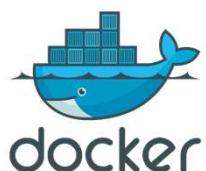


CONTAINERS & KUBERNETES

WHAT IS DOCKER & KUBERNETES (K8S)

Docker

- Docker is a tool designed to make it easier to create, deploy, and run applications by using containers which allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package.



Kubernetes

- Modern applications are increasingly built using containers—microservices packaged with their dependencies and configurations. Kubernetes, or k8s for short, is open-source software for deploying and managing those containers at scale. With Kubernetes, you can build, deliver, and scale containerized apps faster.



kubernetes

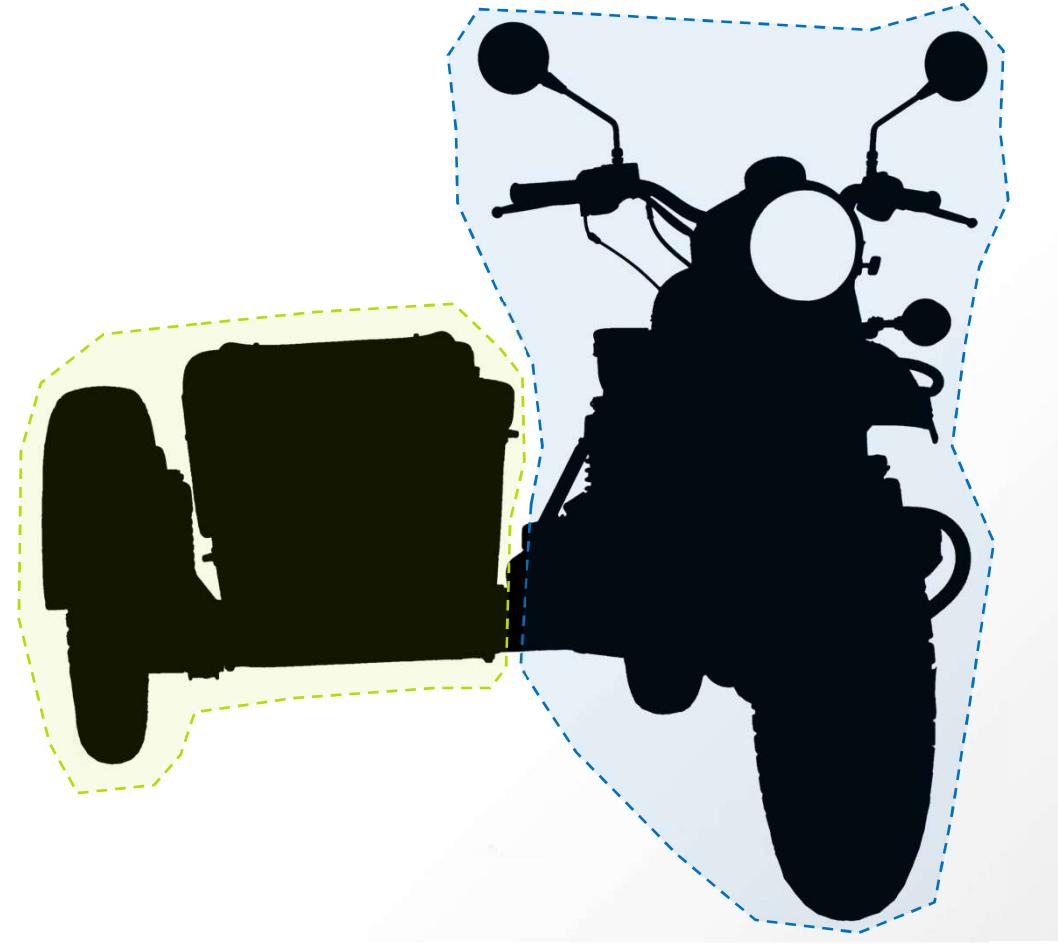
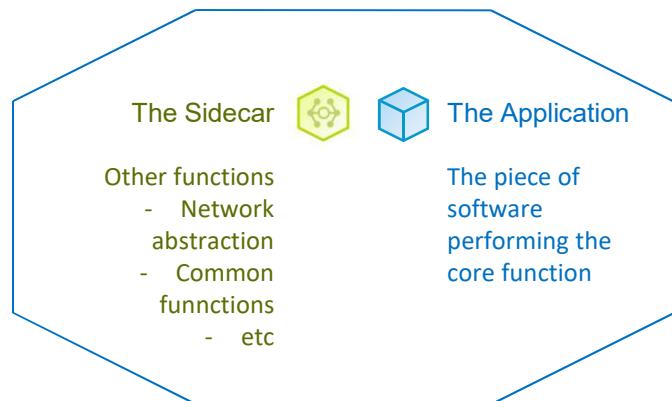
WHY DOCKER / KUBERNETES

A DEVELOPERS PERSPECTIVE

- I don't want to know/care about what infrastructure my microservice runs on
- My service is stateless - I shouldn't have to worry about scale
- Deployment – that's boring
- **I just want to write some code ☺**

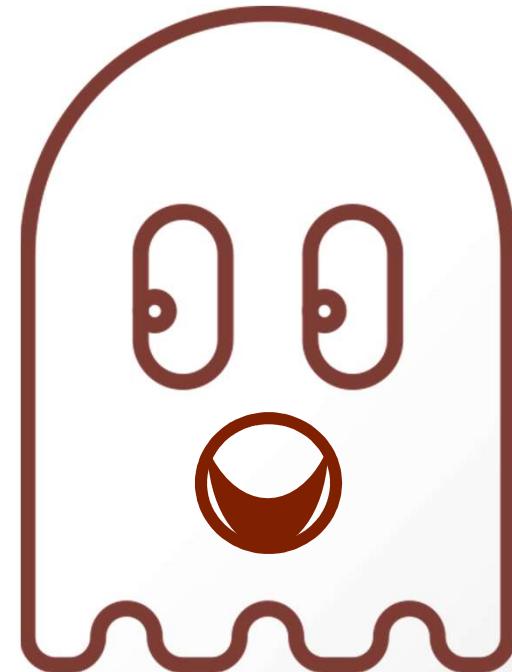
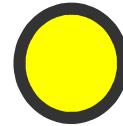
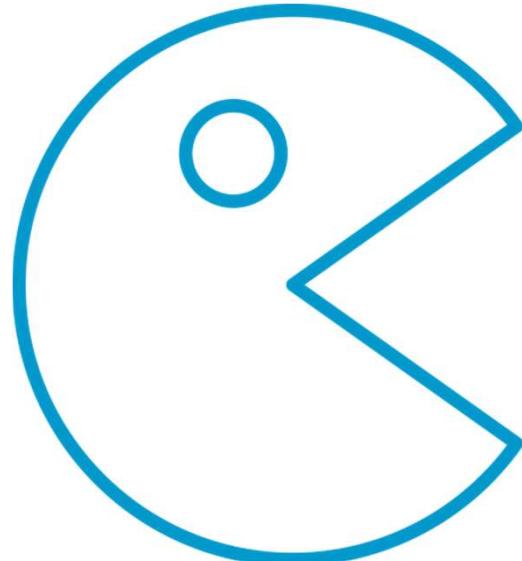
MICROGATEWAY

SIDECAR

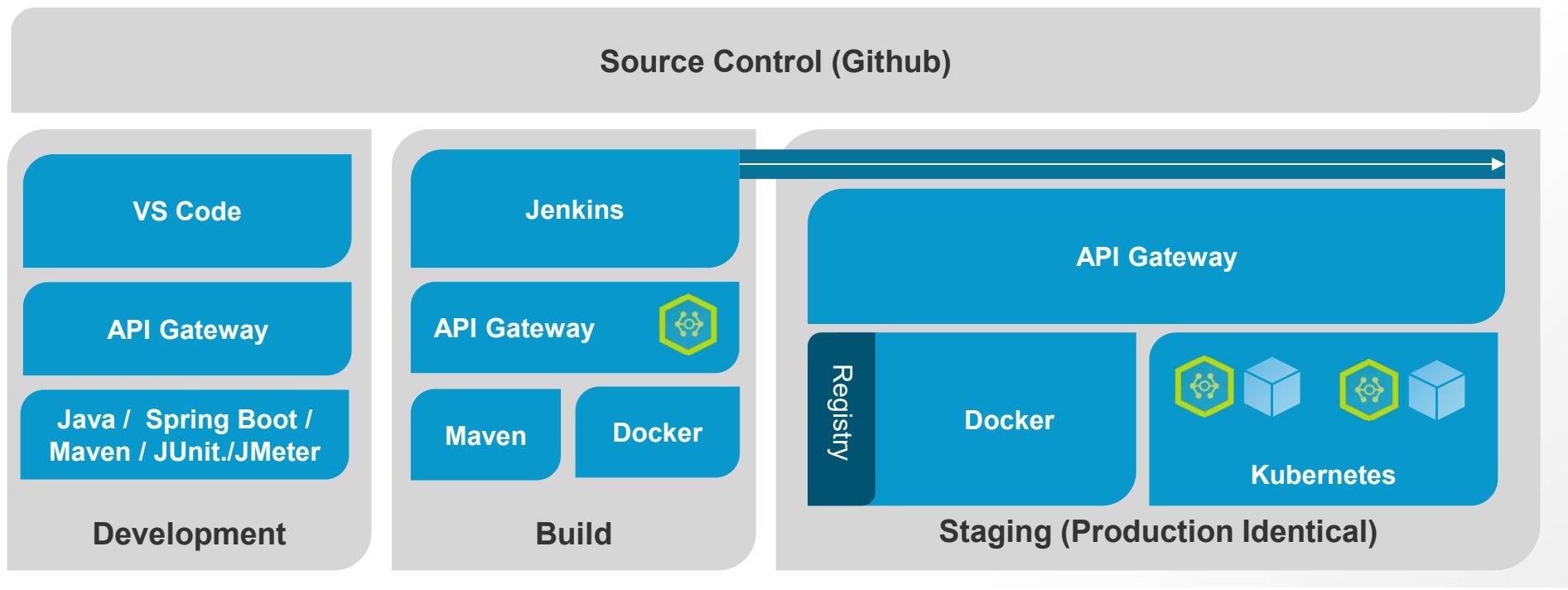


MURPHY'S LAW – IT'S A LIVE DEMO

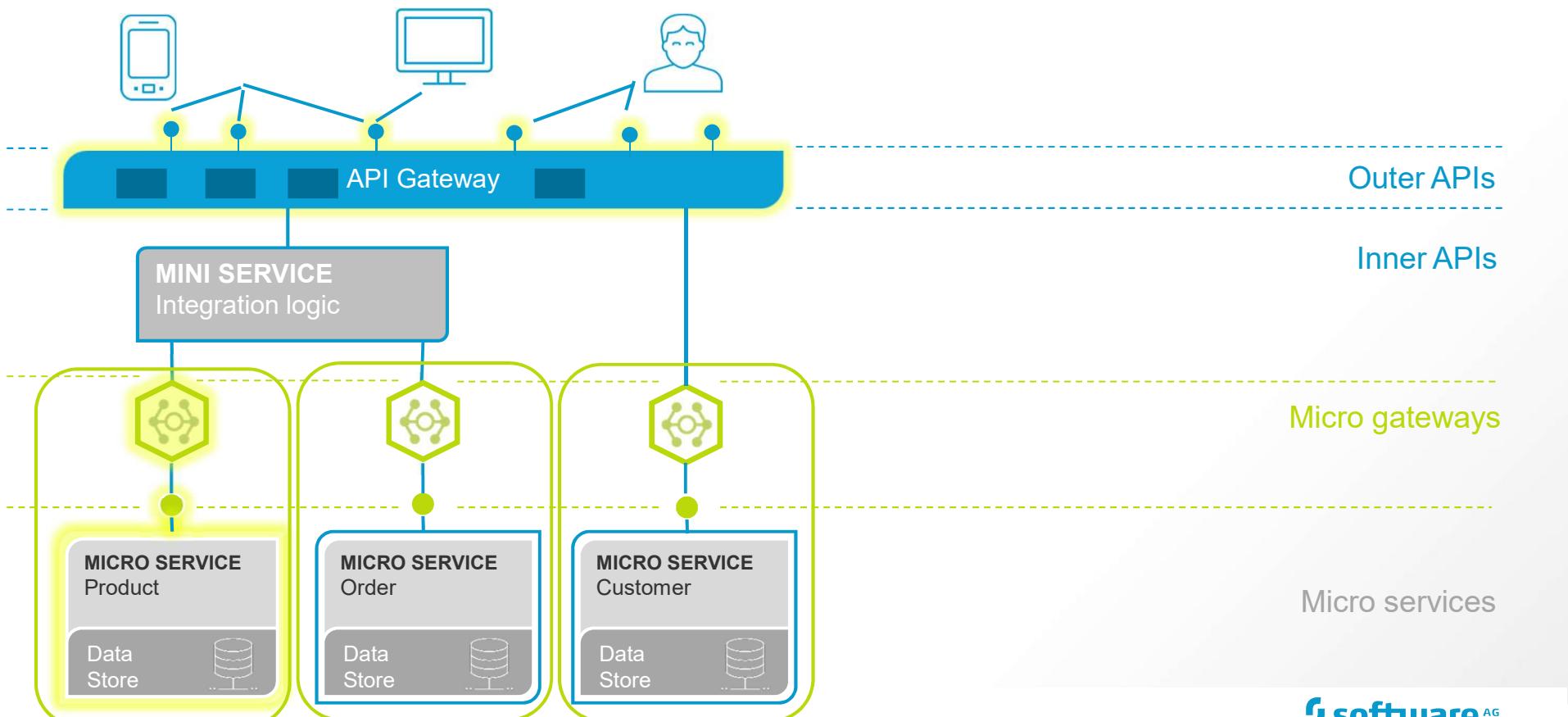
ANYTHING THAT CAN GO WRONG.....



THE DEMO ENVIRONMENT



THE DEMO BUILD!



DEVELOPMENT

MICROSERVICE

- Throughout the rest of this presentation, we're going to make some changes and get them into production in a **MINUTES** with **ZERO DOWNTIME**
- Lets take a look and my Product Microservice and make a change to this and the Gateway



```
Empty.html
send_form_email.php
HTML
CSS
  fonts
  skins
custom.css
ie.css
theme.css
theme-animate.css
theme-blog.css
theme-elements.css
theme-shop.css
chimp
mailer
php-captcha
backgrounds
s
x.php

  78      . ltrim(preg_replace('/\\\\\\\\', '/', $image_src), '/');
  79      $_SESSION['_CAPTCHA']['config'] = serialize($captcha_config);
  80    }
  81    return array(
  82      'code' => $captcha_config['code'],
  83      'image_src' => $image_src
  84    );
  85  }
  86
  87
  88  /**
  89   * If function exists('hex2rgb') {
  90   *   hex2rgb($hex_str, $return_string = false, $separator = ',') {
  91   *     $hex_str = preg_replace("/[^0-9A-Fa-f]/", '', $hex_str); // Gets a proper hex string
  92   *     $rgb_array = array();
  93   *     if( strlen($hex_str) == 6 ) {
  94   *       $color_val = hexdec($hex_str);
  95   *       $rgb_array['r'] = 0xFF & ($color_val >> 0x10);
  96   *       $rgb_array['g'] = 0xFF & ($color_val >> 0x8);
  97   *       $rgb_array['b'] = 0xFF & $color_val;
  98   *     } elseif( strlen($hex_str) == 3 ) {
  99   *       $rgb_array['r'] = hexdec(str_repeat(substr($hex_str, 0, 1), 2));
 100   *       $rgb_array['g'] = hexdec(str_repeat(substr($hex_str, 1, 1), 2));
 101   *       $rgb_array['b'] = hexdec(str_repeat(substr($hex_str, 2, 1), 2));
 102   *     } else {
 103   *       return false;
 104   *     }
 105   *   }
 106   * }
 107   * return $return_string ? implode($separator, $rgb_array) : '';
 108 }
 109 // Draw the image
 110 if( isset($_GET['c']) )
 111   //
```

The image shows a file explorer interface with a dark theme. On the left, a sidebar lists several files: Empty.html, send_form_email.php, HTML, CSS, fonts, skins, custom.css, ie.css, theme.css, theme-animate.css, theme-blog.css, theme-elements.css, theme-shop.css, chimp, mailer, php-captcha, backgrounds, s, and x.php. The main panel displays a portion of a PHP file with line numbers 78 through 111. The code implements a hex2rgb function. It first removes non-hex characters from the input string. Then it checks if the string length is 6 or 3. For a length of 6, it converts the string to an integer and then divides it into three bytes (red, green, blue) using bit shifting and masking. For a length of 3, it repeats each character twice to form a 6-digit hex string and then converts it to an integer, dividing it into three bytes. If the input length is neither 6 nor 3, it returns false. Finally, it returns the color array as a string if \$return_string is true, otherwise it returns the array itself.

DEVELOPMENT

THE MICROSERVICE

```
54  
55  
56  
57  
58     @GetMapping("/product")  
59     public List<ProductItem> product(){  
60  
61         //Populate values if empty  
62         if(getProductItems()==null||getProductItems().isEmpty())populateProductList();  
63         return new ArrayList<ProductItem>(getProductItems().values());  
64     }  
65  
66     @GetMapping("/productcount")  
67     public int productCount(){  
68  
69         if(getProductItems()==null||getProductItems().isEmpty())populateProductList();  
70         return getProductItems().size();  
71     }  
72  
73     @GetMapping("/product/{id}")  
74     public List<ProductItem> product(@PathVariable(value="id") String id){  
75  
76         //Populate values if empty  
77         if(getProductItems()==null||getProductItems().isEmpty())populateProductList();  
78     }
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL

. . .
:: Spring Boot :: (v2.1.5.RELEASE)

2019-10-07 18:36:03.332 INFO 15448 --- [main] c.e.p.ProductServiceApplication : St

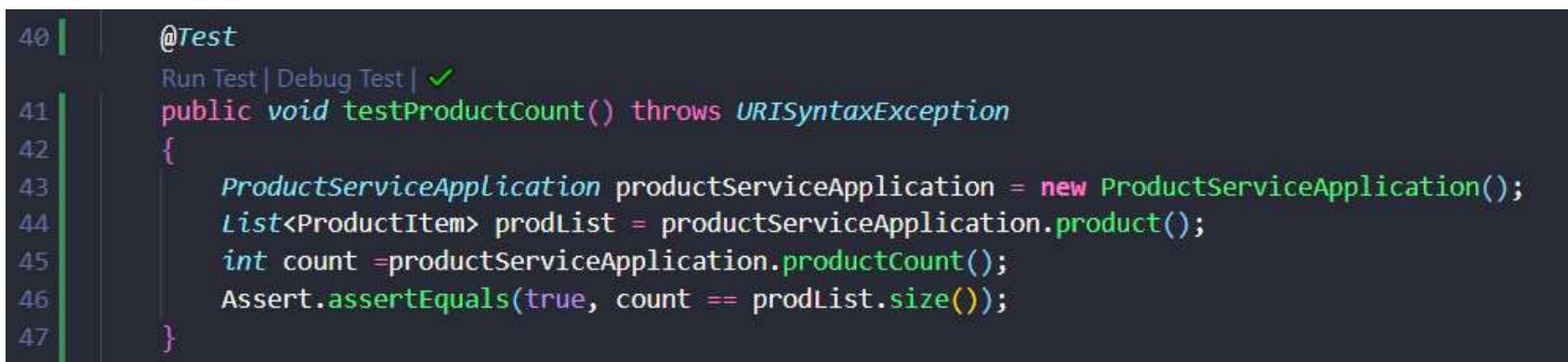
The screenshot shows a REST client interface with two requests:

- Retrieve product**:
 - Method: GET
 - URL: http://localhost:8090/product/1
 - Params tab (selected):

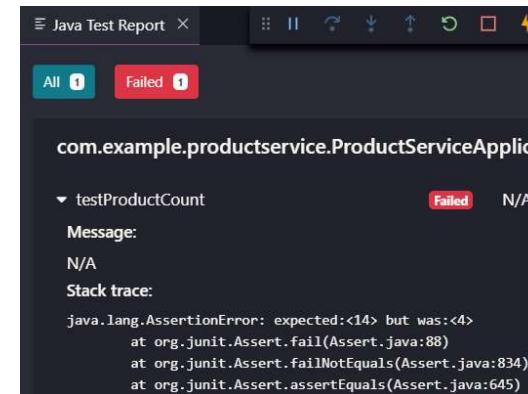
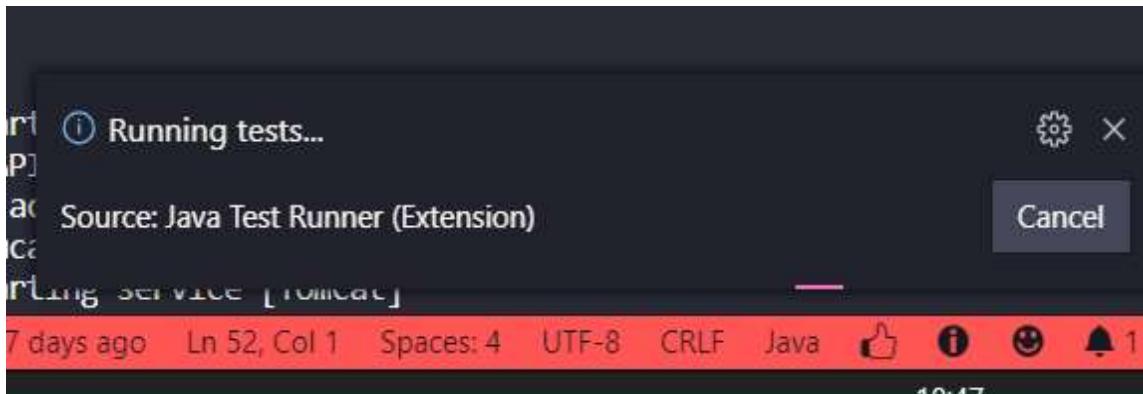
KEY	VALUE	DESCRIPTION
Key	Value	Description
 - Body tab (disabled): Status: 200 OK Time: 717ms Size: 199 B Save
- productcount**:
 - Method: GET
 - URL: http://localhost:8090/productcount
 - Pretty tab (selected):

```
1 [  
2 {  
3   "id": 1,  
4   "productName":  
5   "productDescrip  
6 }]  
7 ]
```
 - Raw tab: 1 4
 - Preview tab: 1 4
 - Visualize BETA tab: JSON

THE MICROSERVICE UNIT TESTS



```
40  @Test
41  Run Test | Debug Test | ✓
42  public void testProductCount() throws URISyntaxException
43  {
44      ProductServiceApplication productServiceApplication = new ProductServiceApplication();
45      List<ProductItem> prodList = productServiceApplication.product();
46      int count = productServiceApplication.productCount();
47      Assert.assertEquals(true, count == prodList.size());
48  }
```



DEVELOPMENT

API GATEWAY ➔ API RESOURCES

The screenshot shows the Software AG API Gateway user interface. The top navigation bar includes tabs for WEBMETHODS, APIs, Policies, Applications, Packages, and Microgateways. A search bar and user account information are also present. The main content area is titled "Product" and contains a sub-header "Update an API by providing the required information." Below this, there are several tabs: "API details" (selected), "Scopes", "Policies", "Mashups", "Applications", and "Analytics". The "API details" tab displays the following fields:

- Resource name***: productcount
- Resource path***: /productcount
- Description**: Gets a count of the product
- Tags**: Enter search terms to see tag suggestions
- Supported methods***:
 - GET
 - POST
 - PUT
 - DELETE
 - PATCH
 - HEAD

At the bottom left, the URL is shown as localhost:9072/apigatewayui#/welcome. The bottom right features the Software AG logo with the tagline "Freedom as a Service".

DEVELOPMENT

API GATEWAY ➔ POLICY CONFIGURATION

WEBMETHODS
API Gateway

APIs Policies Applications Packages Microgateways Type at least 3 characters.

Home > APIs > Product

Product
Update an API by providing the required information.

Policy catalog

- Threat protection
- Transport
- Identify & Access
- Request Processing
- Routing
- Traffic Monitoring
 - Log Invocation
 - Monitor Service Performance
 - Monitor Service Level Agreement
 - Throttling Traffic Optimization
 - Service Result Cache
- Response Processing
- Error Handling

Policy properties

Log Invocation

- Store Request
- Store Response
- Compress Payload Data

 Log Generation Frequency*

- Always

 Destination *

- API Gateway
- API Portal
- CentraSite
- Digital Events
- Elasticsearch
- Email
- JDBC
- Local Log

 Log Level

- Info

SNMP

Version: 10.4.0.0.510

software AG
Freedom as a Service

DEVELOPMENT

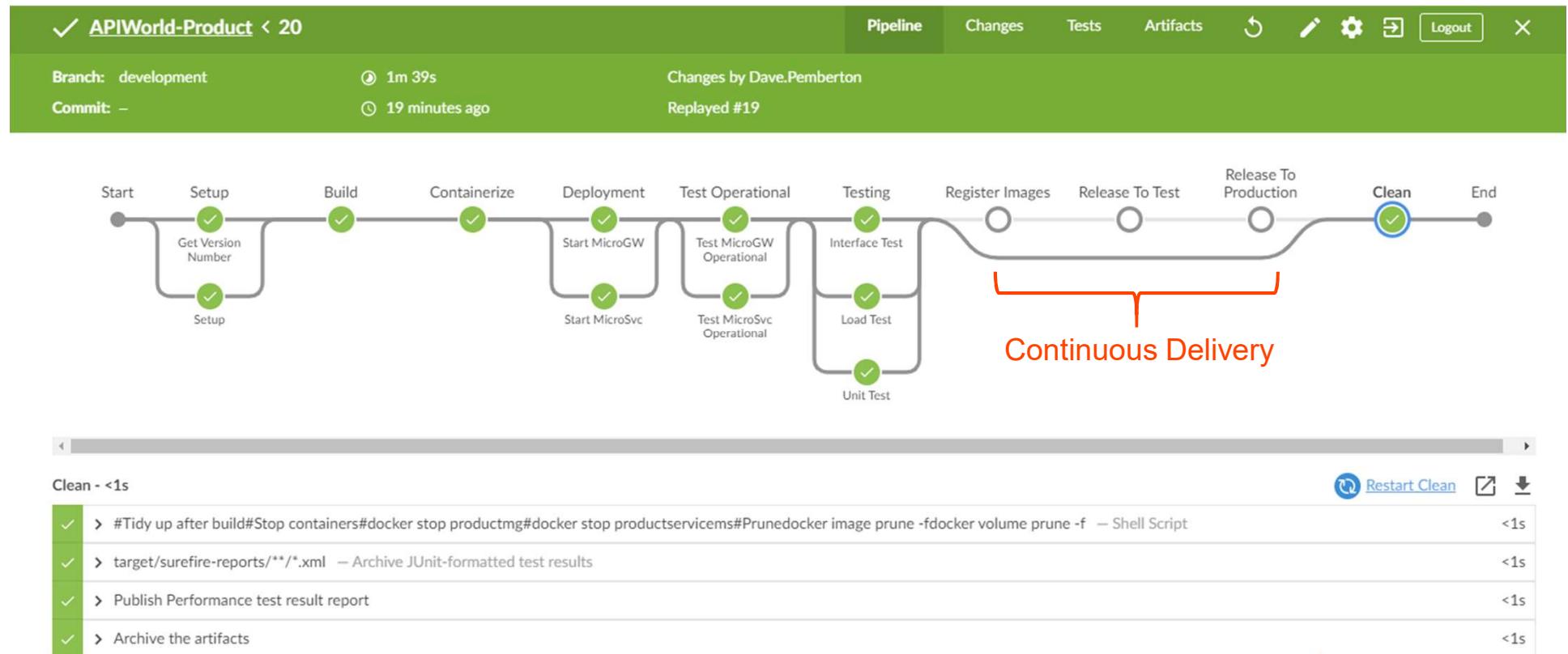
COMMIT/PUSH THE CHANGES



CONTINUOUS INTEGRATION

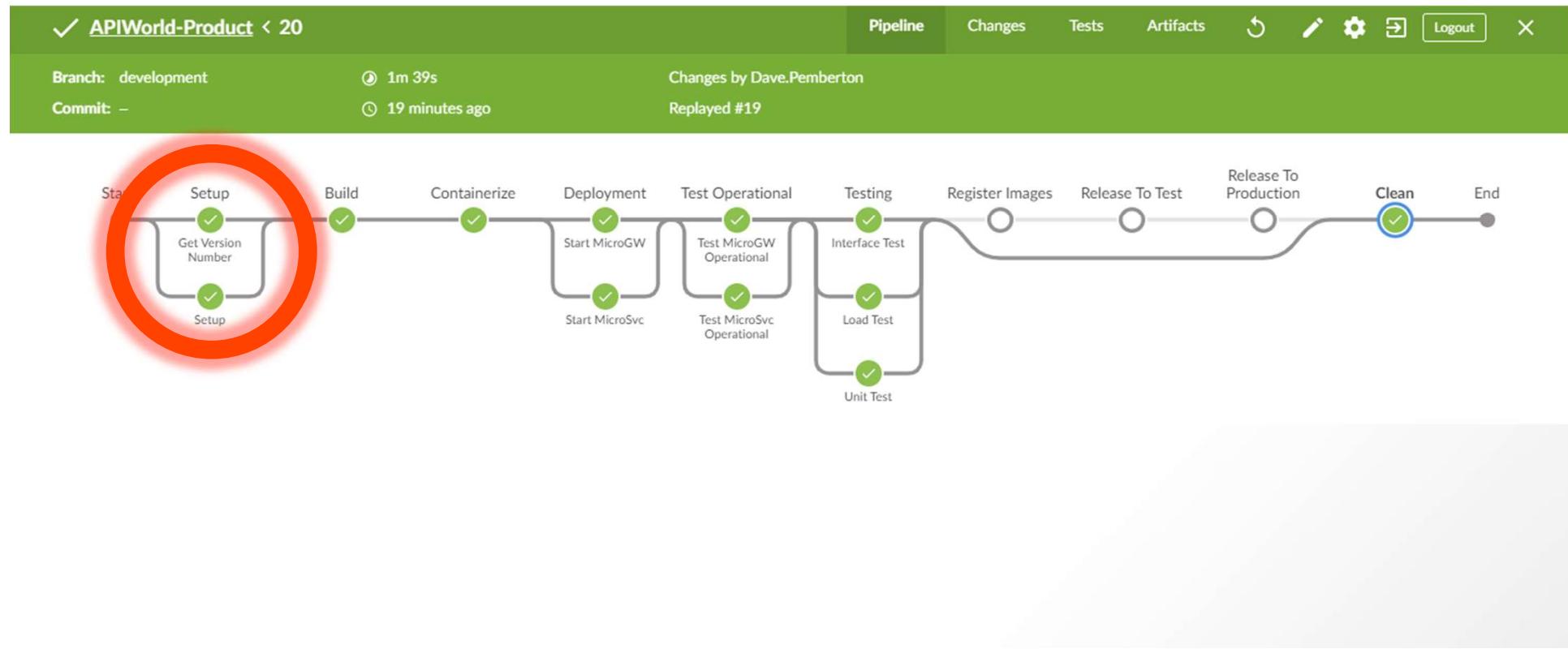
CONTINUOUS INTEGRATION

THE PROCESS



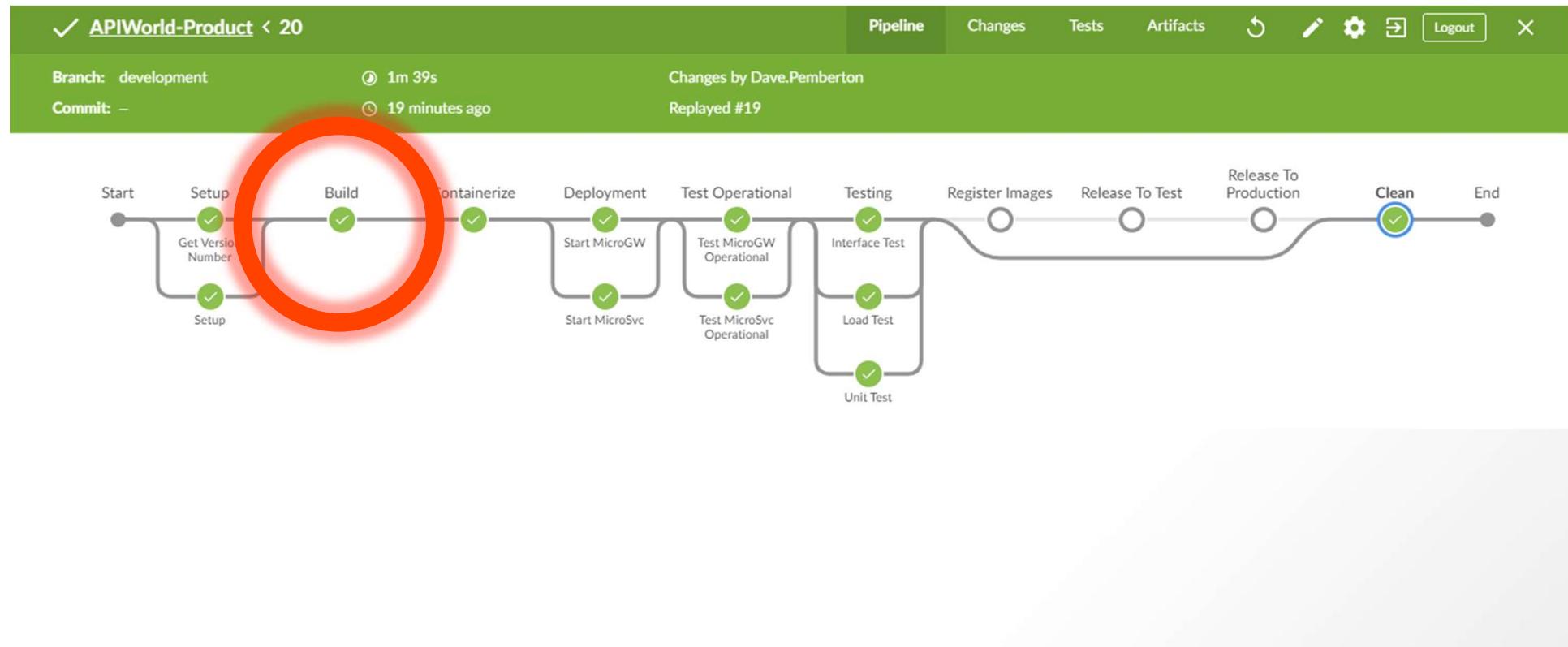
CONTINUOUS INTEGRATION

THE PROCESS



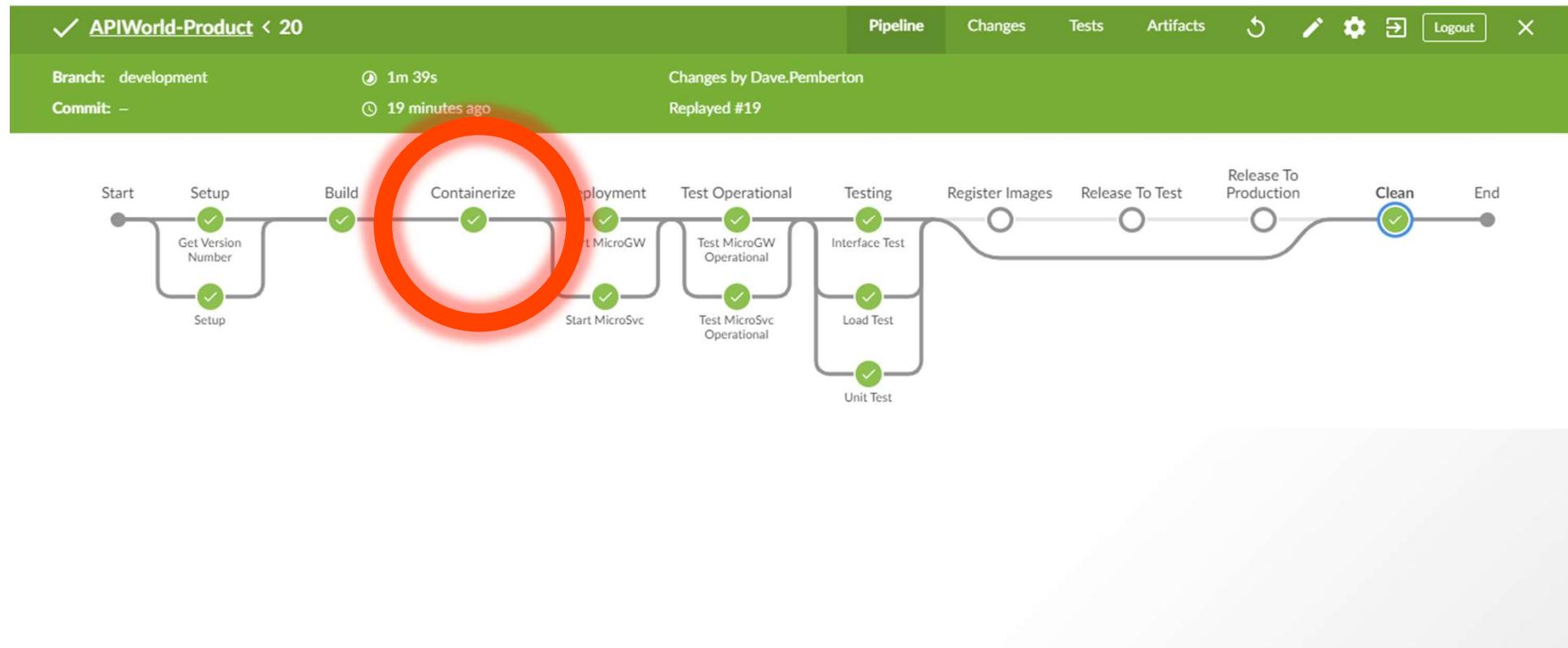
CONTINUOUS INTEGRATION

THE PROCESS



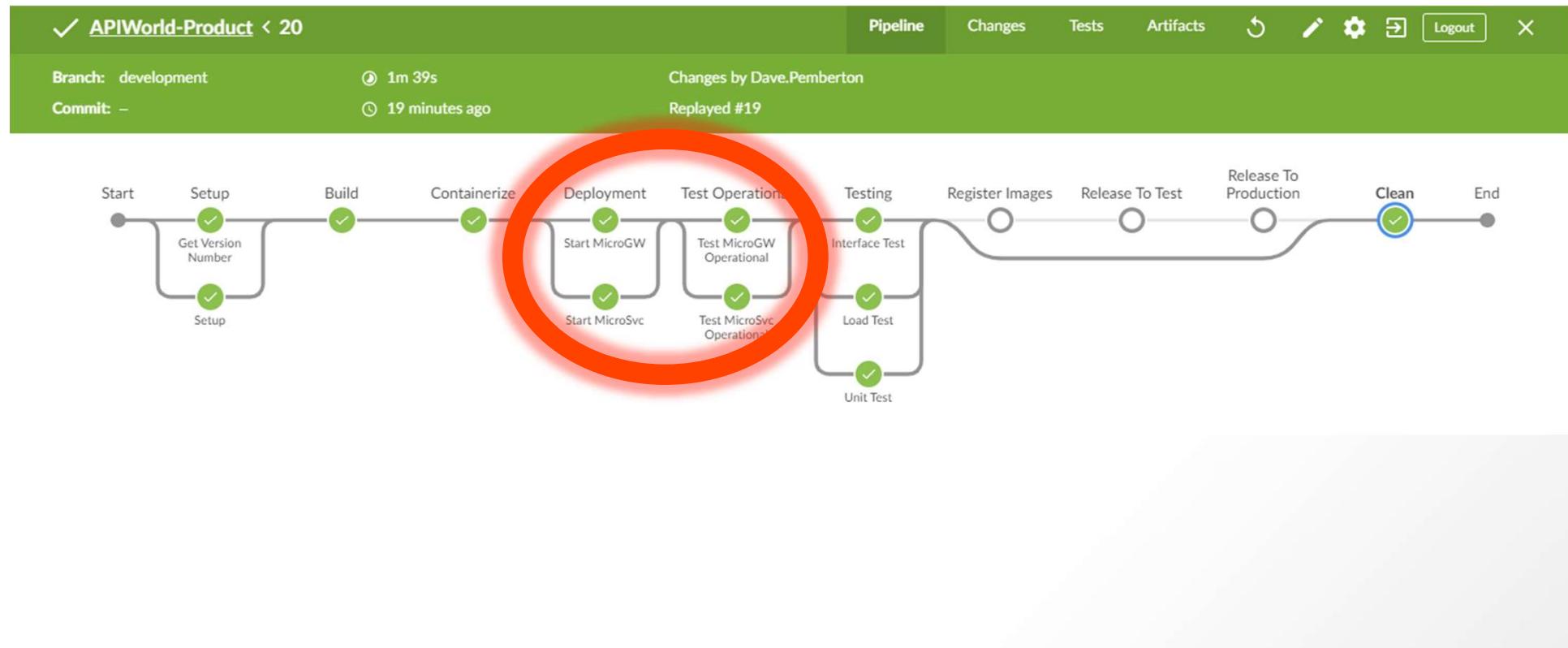
CONTINUOUS INTEGRATION

THE PROCESS



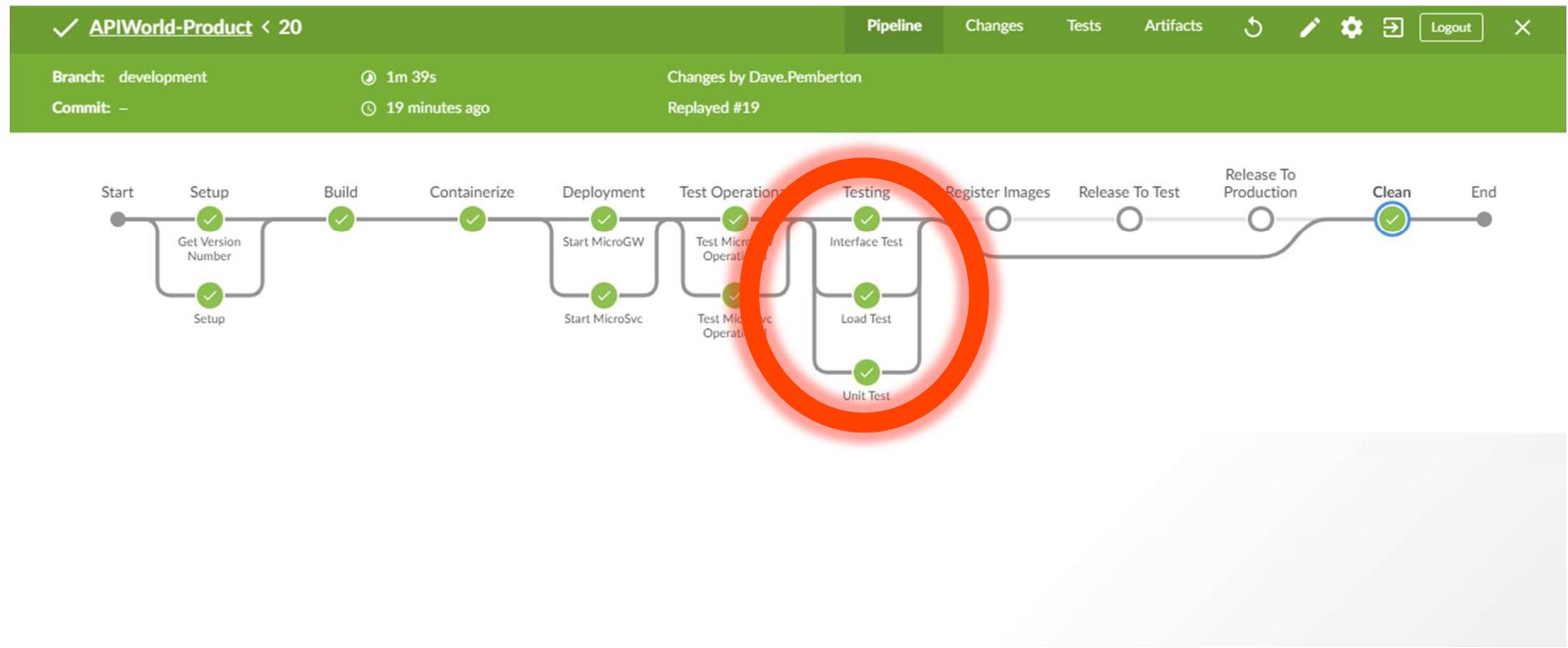
CONTINUOUS INTEGRATION

THE PROCESS



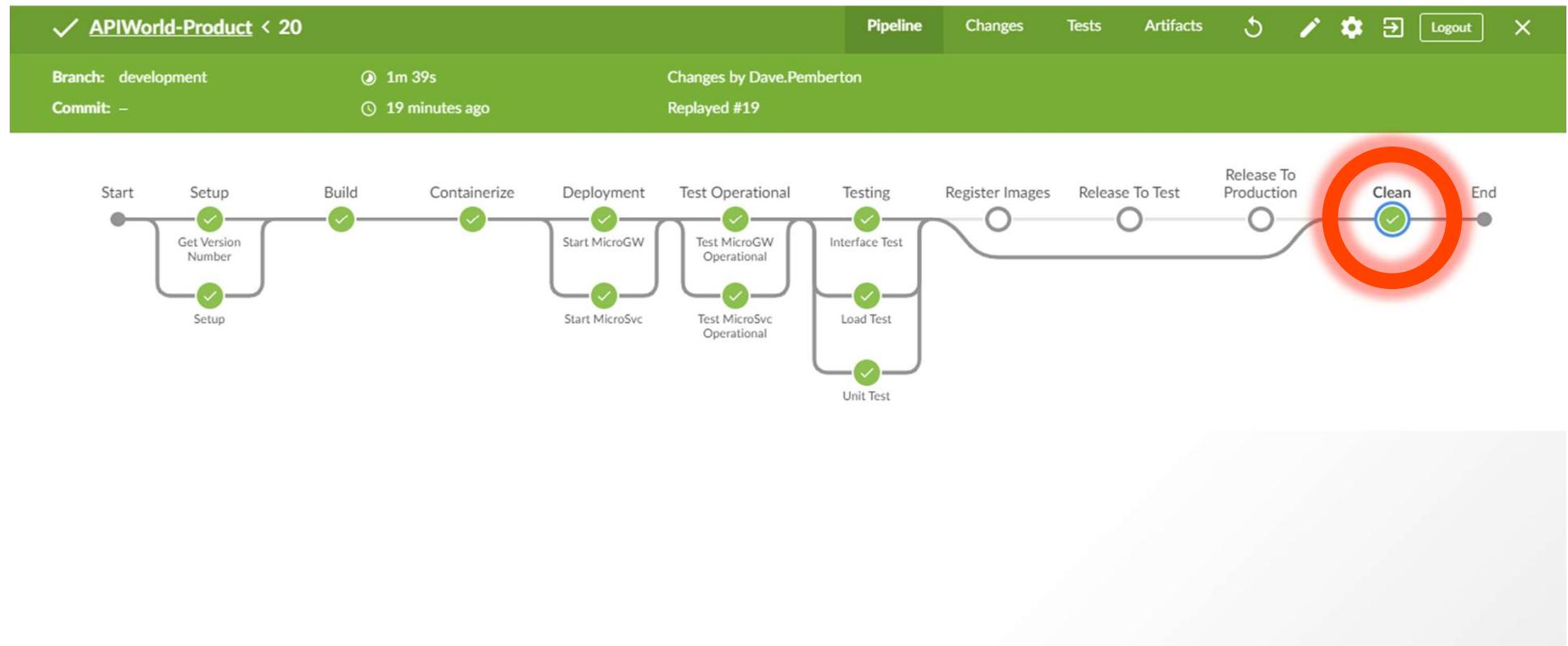
CONTINUOUS INTEGRATION

THE PROCESS



CONTINUOUS INTEGRATION

THE PROCESS



CONTINUOUS INTEGRATION

PERF TEST RESULTS

✓ APIWorld-Product < 20

Pipeline Changes Tests Artifacts Logout X

Branch: development 1m 39s Changes by Dave.Pemberton
Commit: a5f38cf 4 hours ago Replayed #19

All tests are passing

Nice one! All 3 tests for this pipeline are passing.

Passed - 3

✓ > testGetSingleProduct – com.example.productservice.ProductServiceApplicationTests	<1s
✓ > addProduct – com.example.productservice.ProductServiceApplicationTests	<1s
✓ > testProductList – com.example.productservice.ProductServiceApplicationTests	<1s

CONTINUOUS INTEGRATION

TEST RESULTS

✓ APIWorld-Product < 20

Pipeline Changes Tests Artifacts Logout X

Branch: development 1m 39s Changes by Dave.Pemberton
Commit: a5f38cf 4 hours ago Replayed #19

All tests are passing

Nice one! All 3 tests for this pipeline are passing.

Passed - 3

- ✓ > testGetSingleProduct - com.example.productservice.ProductServiceApplicationTests
- ✓ > addProduct - com.example.productservice.ProductServiceApplicationTests
- ✓ > testProductList - com.example.productservice.ProductServiceApplicationTests

Performance Trend

Throughput

Response time

90% line average median

Percentage of errors

Test Result Trend

errors

50 | © 2019 Software AG. All rights reserved. For internal use only and for Software AG Partners.

CONTINUOUS INTEGRATION

ARTIFACTS

✓ APIWorld-Product < 20

Pipeline Changes Tests Artifacts ⌂ ⌚ ⌚ Logout X

Branch: development ⓘ 1m 39s Changes by Dave.Pemberton
Commit: a5f38cf ⓘ 4 hours ago Replayed #19

NAME	SIZE	Actions
pipeline.log	-	🔗 ↴
dashBoard_result.xml	451 bytes	🔗 ↴
jmeter/result.jtl	8 KB	🔗 ↴
jmeter/result.log	17.5 KB	🔗 ↴
standardResults.xml	252 bytes	🔗 ↴

GIT STRATEGY

MOVE INTO STAGING



CONTINUOUS DELIVERY & DEPLOYMENT



CREATE THE PULL REQUEST

The screenshot shows two side-by-side GitHub pull request pages for a repository named 'APIWorld-Product'.

Left Panel (Development #13):

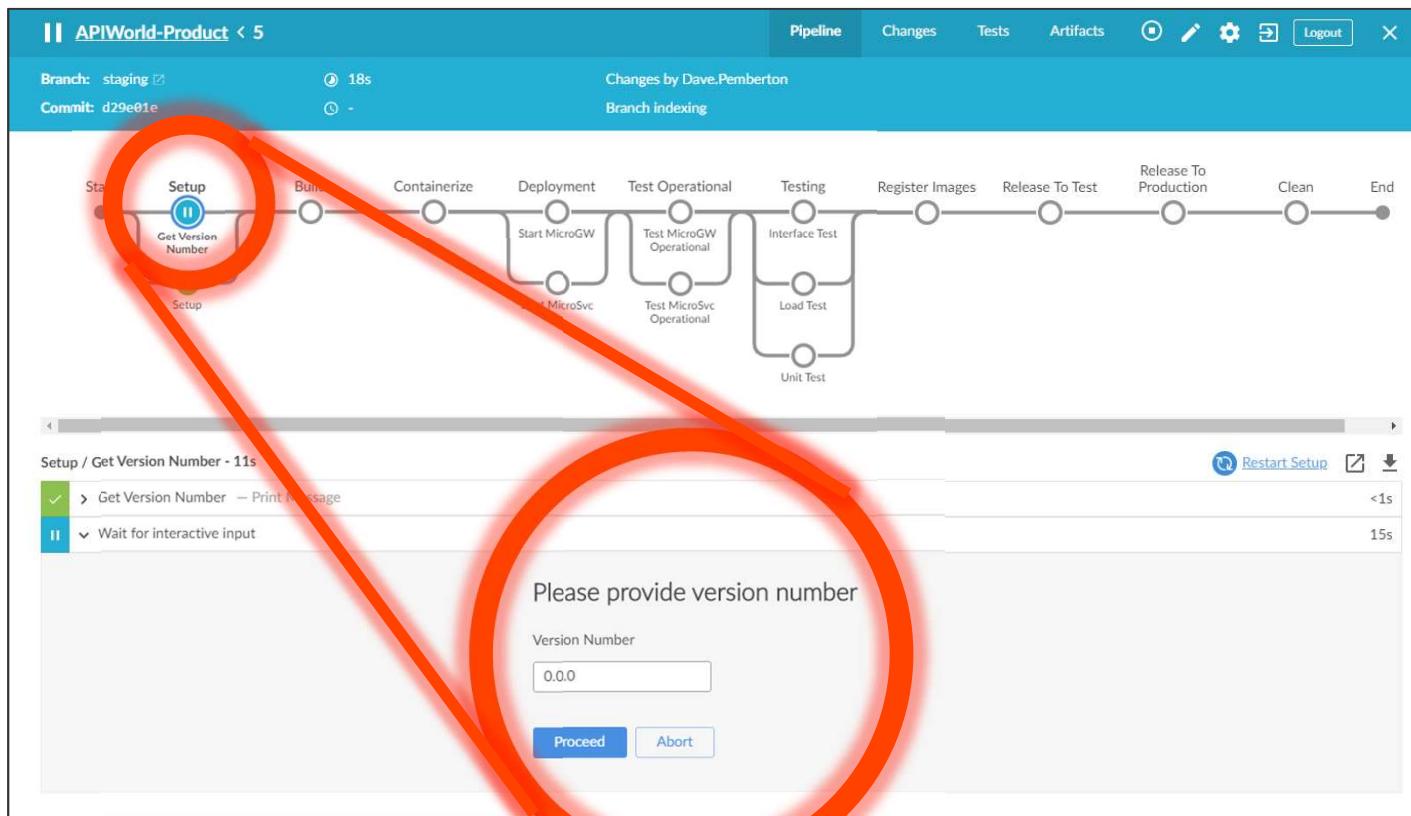
- Header:** Development #13
- Status:** Open
- Author:** dpembo wants to merge 2 commits into `staging` from `development`.
- Conversation:** 0
- Commits:** 2
- Checks:** 0
- Files changed:** 3
- Reviewers:** No reviews
- Assignees:** No one—assign yourself
- Labels:** enhancement (highlighted), staging
- Projects:** None yet
- Milestone:** No milestone
- Notifications:** You're receiving notifications because you authored the thread.
- Participants:** 1 participant (dpembo)
- Actions:** Merge pull request, Preview, Write, Lock conversation.

Right Panel (Development #13):

- Status:** Merged
- Author:** dpembo merged 2 commits into `staging` from `development` 39 seconds ago.
- Conversation:** 0
- Commits:** 2
- Checks:** 0
- Files changed:** 3
- Reviewers:** No reviews
- Assignees:** No one—assign yourself
- Labels:** enhancement (highlighted), staging
- Projects:** None yet
- Milestone:** No milestone
- Notifications:** You're receiving notifications because you modified the open/close state.
- Participants:** 1 participant (dpembo)
- Actions:** View details, Revert, Delete branch, Write, Preview, Lock conversation.

DEVOPS

CONTINUOUS DELIVERY



DEVOPS

CONTINUOUS DELIVERY

✓ APIWorld-Product < 11

Branch: staging 1m 51s Changes by Dave.Pemberton
Commit: c631ca5 4 minutes ago Replayed #10

Pipeline Changes Tests Artifacts Logout X

```
graph LR; Start((Start)) --> Setup((Setup)); Setup --> GetVersionNumber[Get Version Number]; GetVersionNumber --> Setup; Setup --> Build((Build)); Build --> Containerize((Containerize)); Containerize --> Deployment((Deployment)); Deployment --> StartMicroGW[Start MicroGW]; StartMicroGW --> StartMicroSvc[Start MicroSvc]; Deployment --> TestOperational((Test Operational)); TestOperational --> TestMicroGWOperational[Test MicroGW Operational]; TestOperational --> TestMicroSvcOperational[Test MicroSvc Operational]; TestOperational --> Testing((Testing)); Testing --> InterfaceTest[Interface Test]; Testing --> LoadTest[Load Test]; Testing --> UnitTest[Unit Test]; Testing --> RegisterImages((Register Images)); RegisterImages --> ReleaseToTest[Release To Test]; ReleaseToTest --> Clean((Clean)); Clean --> End((End));
```

Register Images - 12s

Restart Register Images

```
✓ ✓ #push image to registry#First tagdocker tag productservice:$VERSION apiworldref:5000/productservice:$VERSIONdock... — Shell Script 12s
```

```
1 + docker tag productservice:1.0.4 apiworldref:5000/productservice:1.0.4
2 + docker tag productmg:1.0.4 apiworldref:5000/productmg:1.0.4
3 + docker push apiworldref:5000/productservice:1.0.4
4 The push refers to repository [apiworldref:5000/productservice]
5 776a0d18ac7d: Preparing
6 ceaf9e1ebef5: Preparing
7 9b9b7f3d56a0: Preparing
8 f1b5933fe4b5: Preparing
9 ceaf9e1ebef5: Layer already exists
```

CONTINUOUS DELIVERY

RELEASE TO TEST

✓ APIWorld-Product < 11

Branch: staging ⌚ 1m 51s Changes by Dave.Pemberton
Commit: c631ca5 ⌚ 2 minutes ago Replied #10

Pipeline Changes Tests Artifacts ⚡ 🖊️⚙️🔗Logout X

```
graph LR; Start((Start)) --> GetVersionNumber[Get Version Number]; GetVersionNumber --> Setup1((Setup)); Setup1 --> Build((Build)); Build --> Containerize((Containerize)); Containerize --> Deployment((Deployment)); Deployment --> TestOperational((Test Operational)); TestOperational --> Testing((Testing)); Testing --> RegisterImages((Register Images)); RegisterImages --> ReleaseToTest((Release To Test)); ReleaseToTest --> ReleaseToProduction((Release To Production)); ReleaseToProduction --> Clean((Clean)); Clean --> End((End));
```

Release To Test - 3s

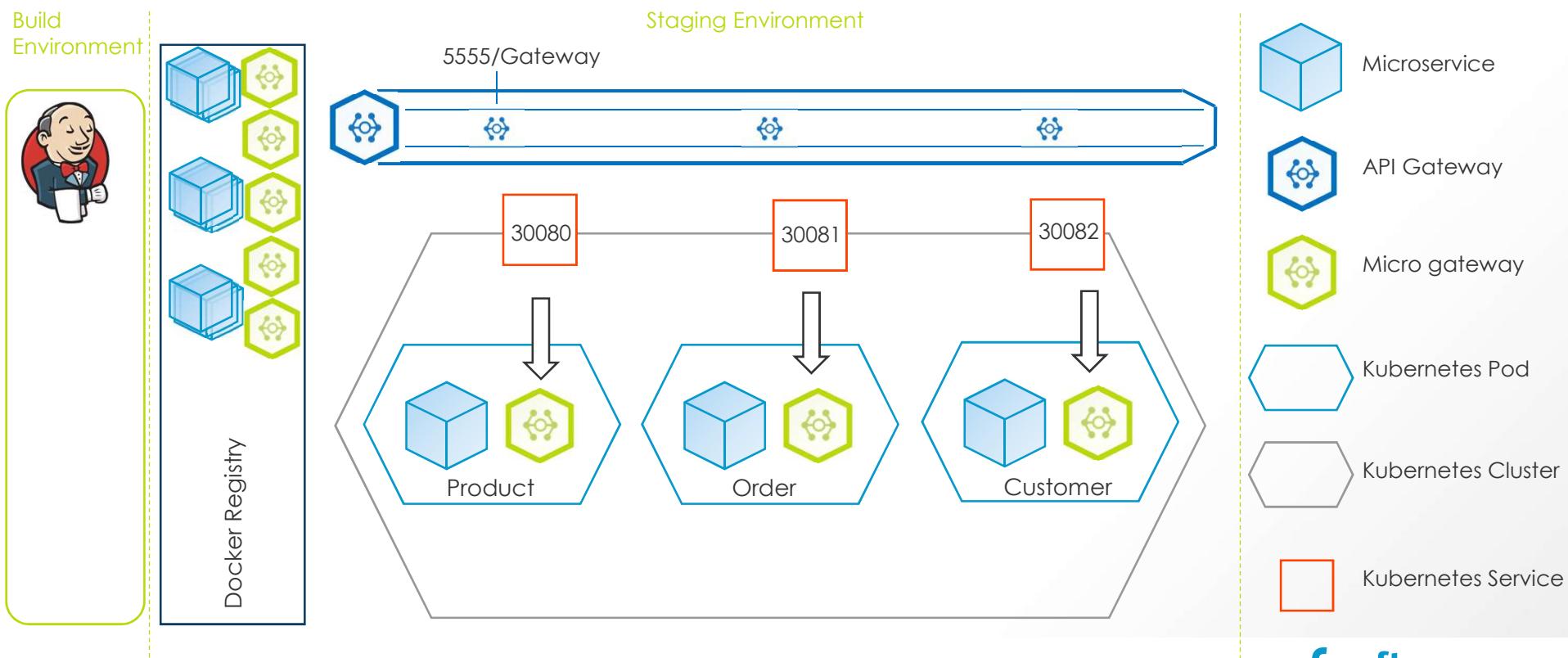
↻ Restart Release To Test ↻ ↴

Step	Description	Time
✓	> Check out from version control	1s
✓	> Release to test — Print Message	<1s
✓	Shell Script	1s

```
1  ++ kubectl get deployments.apps
2  ++ grep product-service-deployment
3  ++ wc -l
4  + deployActive=1
5  +'[' 1 -gt 0 ']'
6  + echo 'Perform Rolling Update'
7  Perform Rolling Update
8  + kubectl set image deployment.v1.apps/product-service-deployment product-service=apiworldref:5000/productservice:1.0.4
9  + kubectl set image deployment.v1.apps/product-service-deployment product-service-sidecar=apiworldref:5000/productmg:1.0.4
10 + kubectl rollout status deployment.v1.apps/product-service-deployment
11 deployment "product-service-deployment" successfully rolled out
```

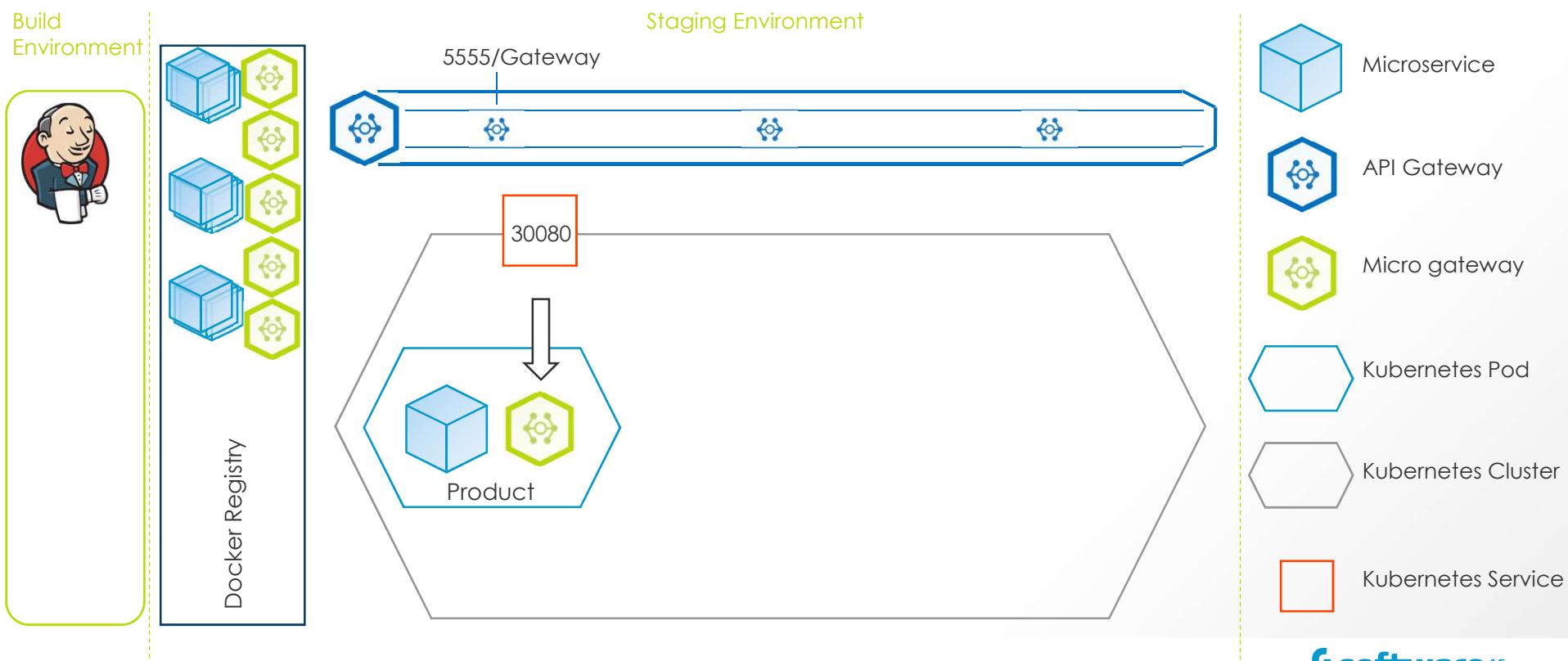
ROLLING RELEASE

THE ENVIRONMENT



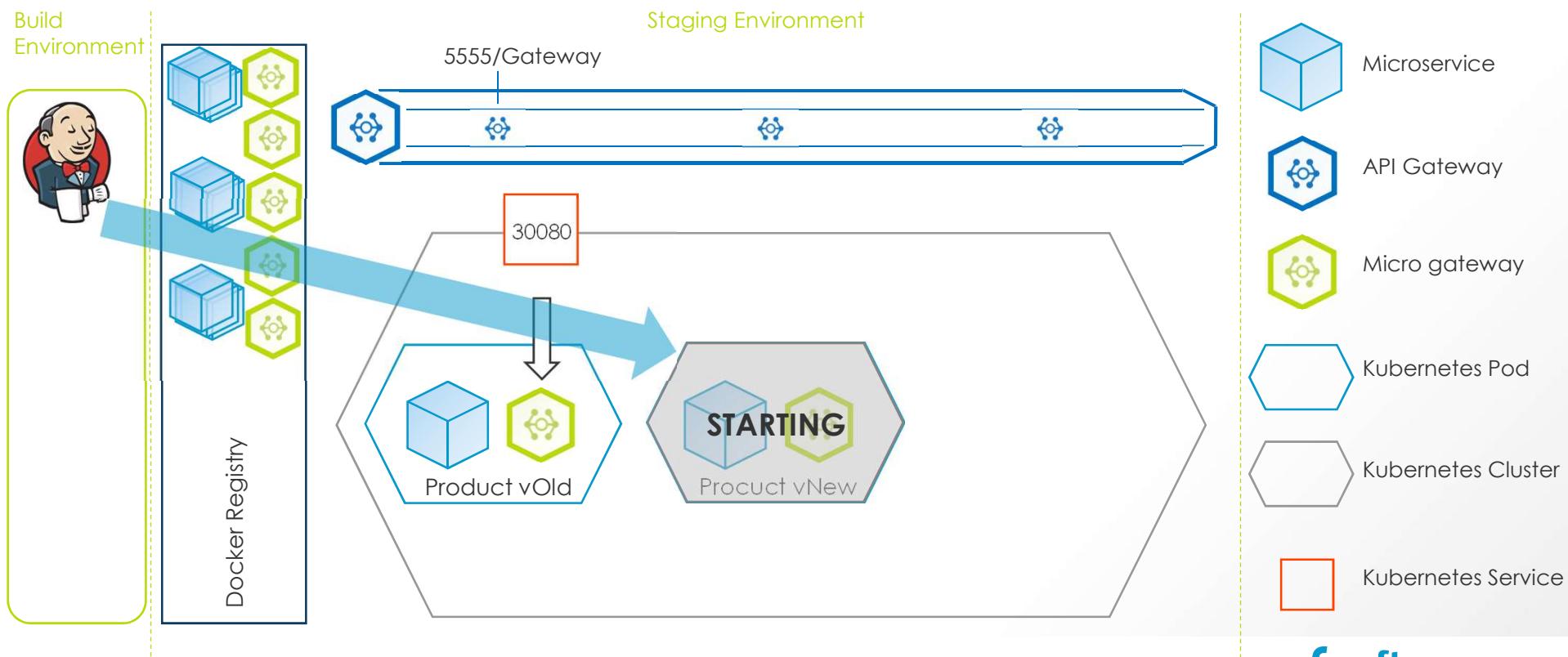
ROLLING RELEASE

THE ENVIRONMENT



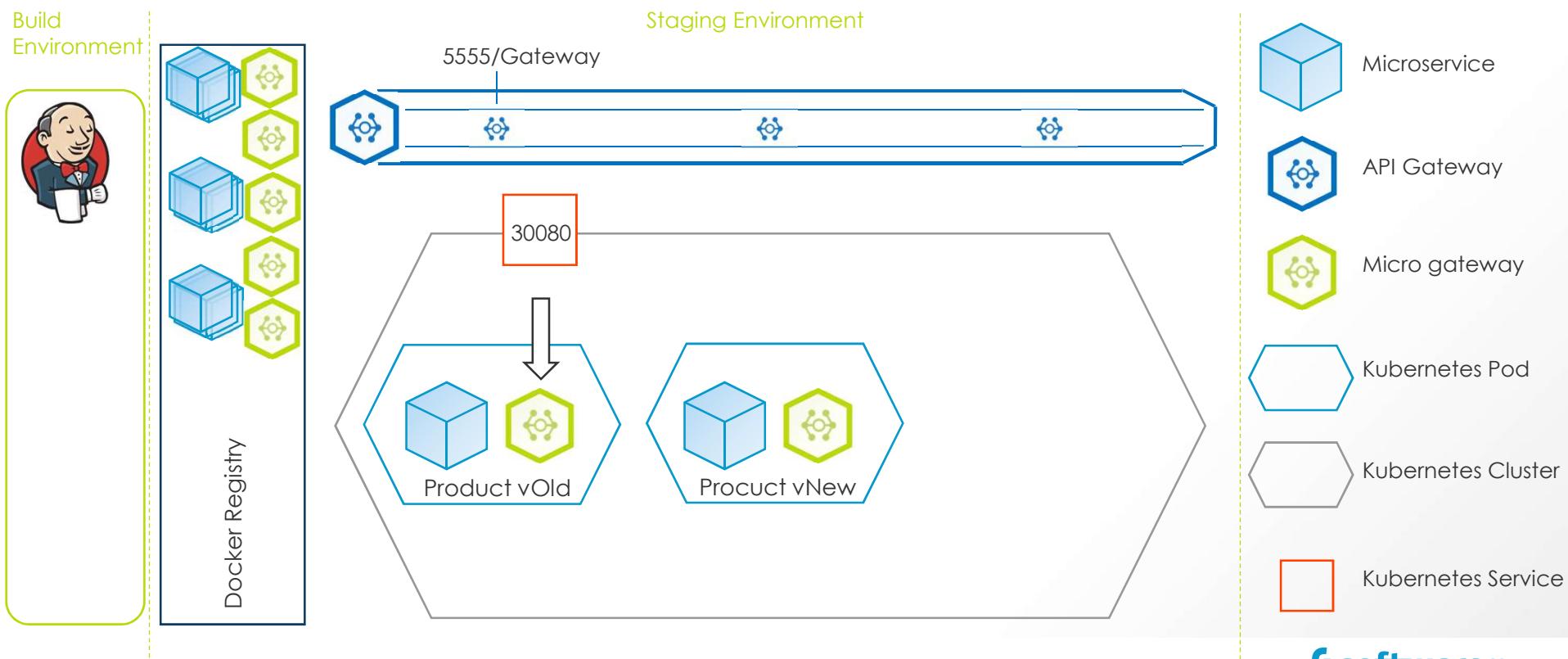
ROLLING RELEASE

START A PARALLEL MICROSERVICE/GATEWAY DEPLOY



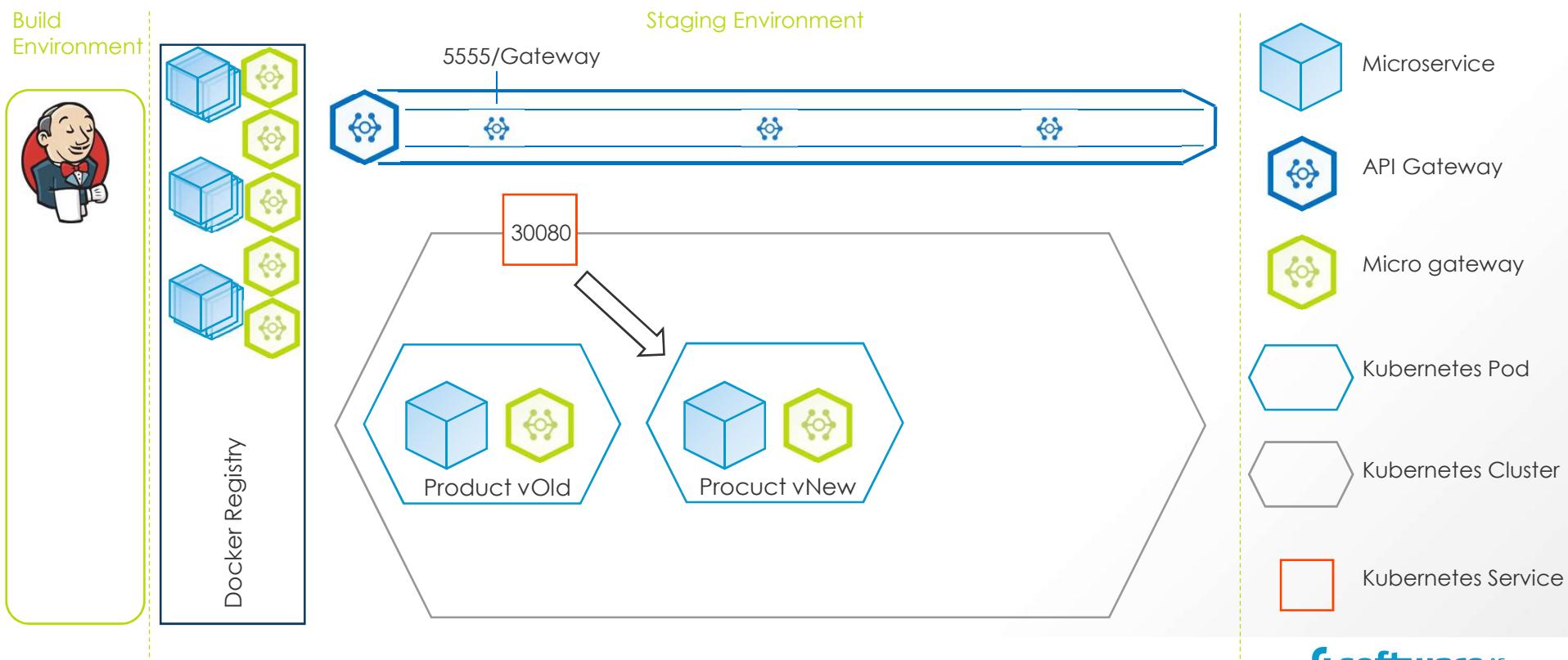
STAGING / PRODUCTION LAYOUT

START A PARALLEL MICROSERVICE/GATEWAY DEPLOY



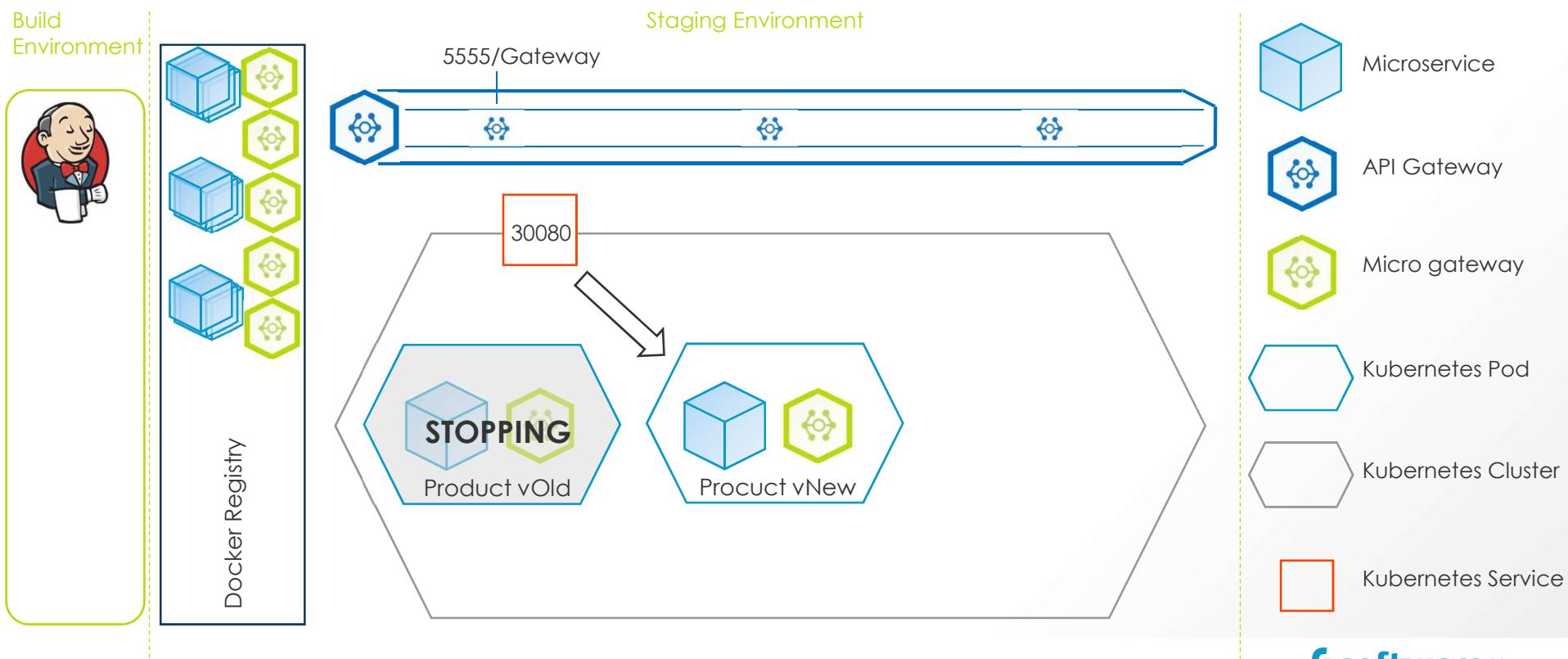
STAGING / PRODUCTION LAYOUT

CHANGE THE END POINT OVER



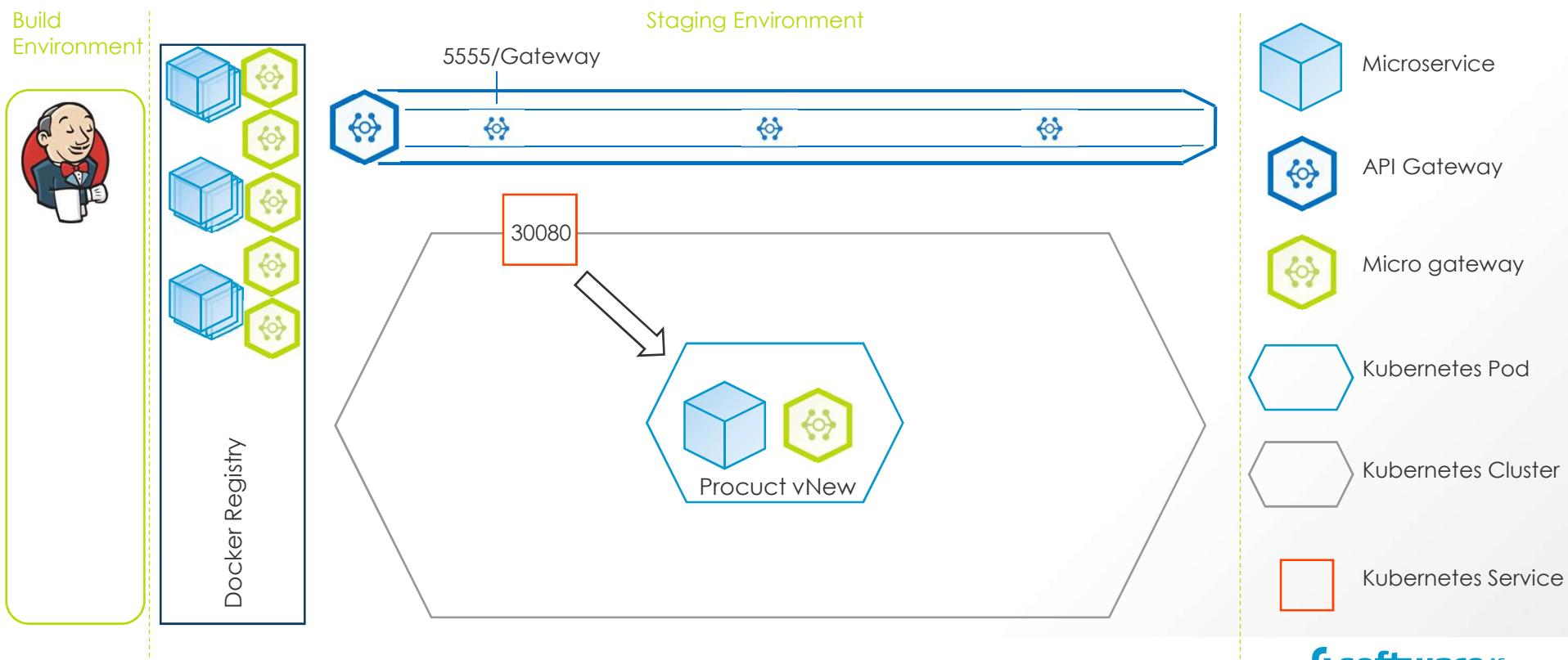
STAGING / PRODUCTION LAYOUT

STOP THE PREVIOUS SERVICE



STAGING / PRODUCTION LAYOUT

NEW NORMAL



AND...

KUBERNETES

```
[root@apiWorldRef ~]# kubectl get all
NAME                                         READY   STATUS    RESTARTS   AGE
pod/order-service-deployment-86c7486589-q5cx8   2/2     Running   0          67m
pod/product-service-deployment-75579b7dc4-mwhms   2/2     Running   0          20m

NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)        AGE
service/kubernetes   ClusterIP   10.96.0.1      <none>        443/TCP       6d2h
service/order-service-svc   NodePort    10.103.69.62    <none>        9090:30081/TCP  3d7h
service/product-service-svc   NodePort    10.104.231.57   <none>        9090:30080/TCP  5d1h

NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/order-service-deployment   1/1      1           1          3d7h
deployment.apps/product-service-deployment 1/1      1           1          13h

NAME           DESIRED  CURRENT  READY   AGE
replicaset.apps/order-service-deployment-6b7cc494d6  0         0        0      67m
replicaset.apps/order-service-deployment-6c58696dbf  0         0        0      11h
replicaset.apps/order-service-deployment-86c7486589  1         1        1      67m
replicaset.apps/product-service-deployment-5c948db756  0         0        0      105m
replicaset.apps/product-service-deployment-647d6bfcf5  0         0        0      72m
replicaset.apps/product-service-deployment-6d87fd9b8  0         0        0      10h
replicaset.apps/product-service-deployment-6d975b8cc8  0         0        0      11h
replicaset.apps/product-service-deployment-75579b7dc4  1         1        1      20m
replicaset.apps/product-service-deployment-75c8bdd865  0         0        0      105m
replicaset.apps/product-service-deployment-7774578498  0         0        0      6h32m
replicaset.apps/product-service-deployment-78975fbbcc  0         0        0      10h
replicaset.apps/product-service-deployment-79668869f8  0         0        0      20m
replicaset.apps/product-service-deployment-846958c789  0         0        0      6h32m
replicaset.apps/product-service-deployment-b4f9df8cd   0         0        0      72m
```

INVOCATION OF API (RUNNING IN KUBERNETES)

The screenshot shows the Postman application interface. At the top, there's a navigation bar with 'My Workspace', 'Invite', and 'Sign In'. Below the navigation is a toolbar with various icons for different operations like GET, POST, PUT, etc. The main workspace title is 'APIGW - Retrieve Product'. The request method is set to 'GET' and the URL is 'http://APIWorldref.eur.ad.sag:30080/gateway/Product/1.0/product'. Below the URL, there are buttons for 'Send' and 'Save'. The 'Params' tab is selected, showing a single query parameter 'Key' with 'Value'. Under the 'Body' tab, the response is displayed as JSON. The JSON data is as follows:

```
1 [ { 2   "id": 1, 3   "productName": "foo", 4   "productDescription": "Foo Description" 5 }, { 6   "id": 2, 7   "productName": "bar", 8   "productDescription": "Bar Description" 9 }, {10   "id": 3,11   "productName": "wgt",12   "productDescription": "Widget Description"13 }, {14   "id": 4,15   "productName": "dgb",16 }]
```

API METRICS

The screenshot shows the webMethods API Gateway interface. The top navigation bar includes links for WEBMETHODS, API Gateway, APIs, Policies, Applications, Packages, and Microgateways. A search bar and user authentication dropdown are also present.

The main content area displays the details for a microgateway named "product-service-deployment-75579b7dc4-mwhms". The title is "product-service-deployment-75579b7dc4-mwhms" and the subtext is "View microgateway details along with the APIs associated with the microgateway".

The "Basic information" section shows the following details:

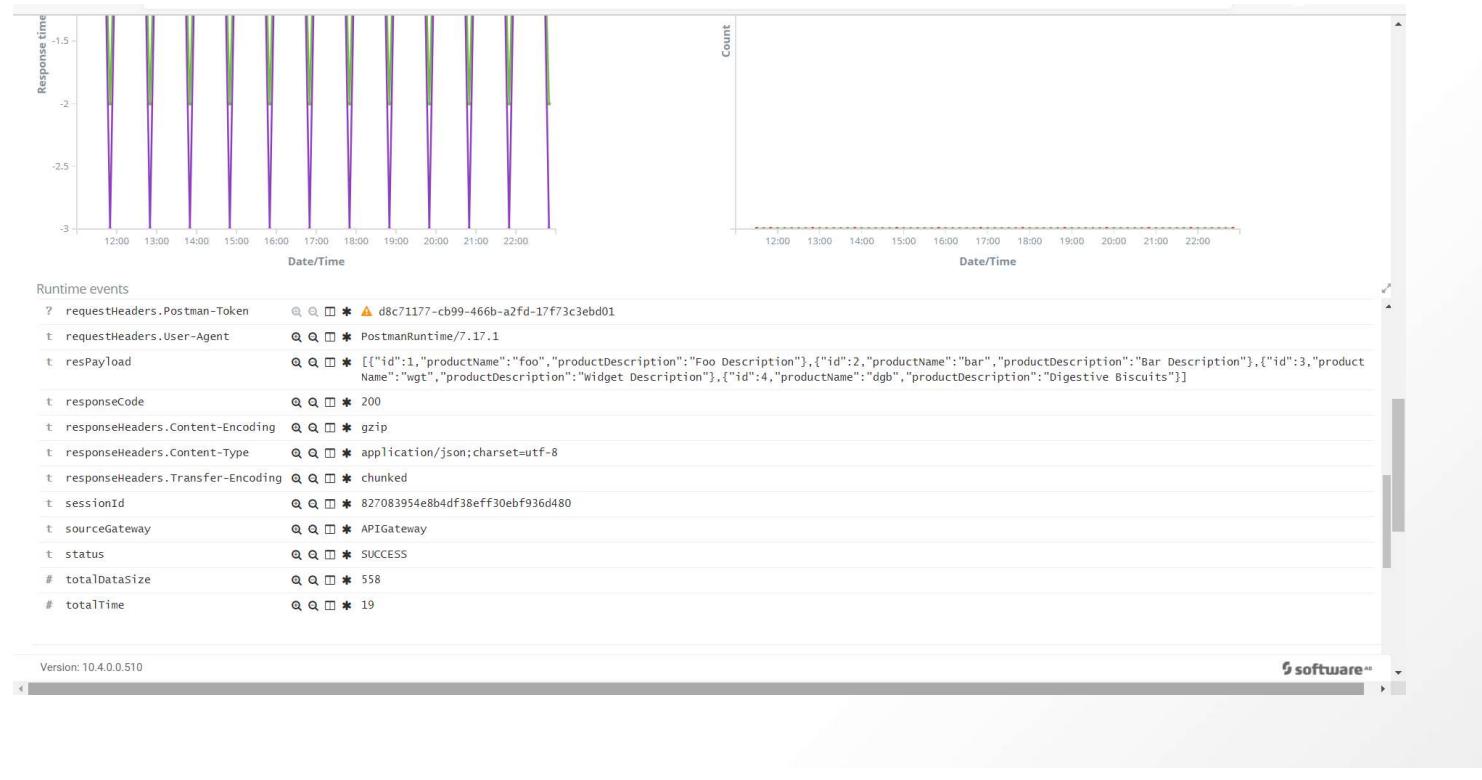
- Name: product-service-deployment-75579b7dc4-mwhms
- Description: webMethods Microgateway
- Version: 10.5.0.1.37
- Host: product-service-deployment-75579b7dc4-mwhms
- HTTP Port: 9090
- Microgateway pool name: Default

The "APIs" section lists one API named "Product" (REST), which is described as "API Product Service to fetch products". The API statistics are as follows:

Name	Description	Policies
Product	API Product Service to fetch products	1 0 0 1 0 0 0

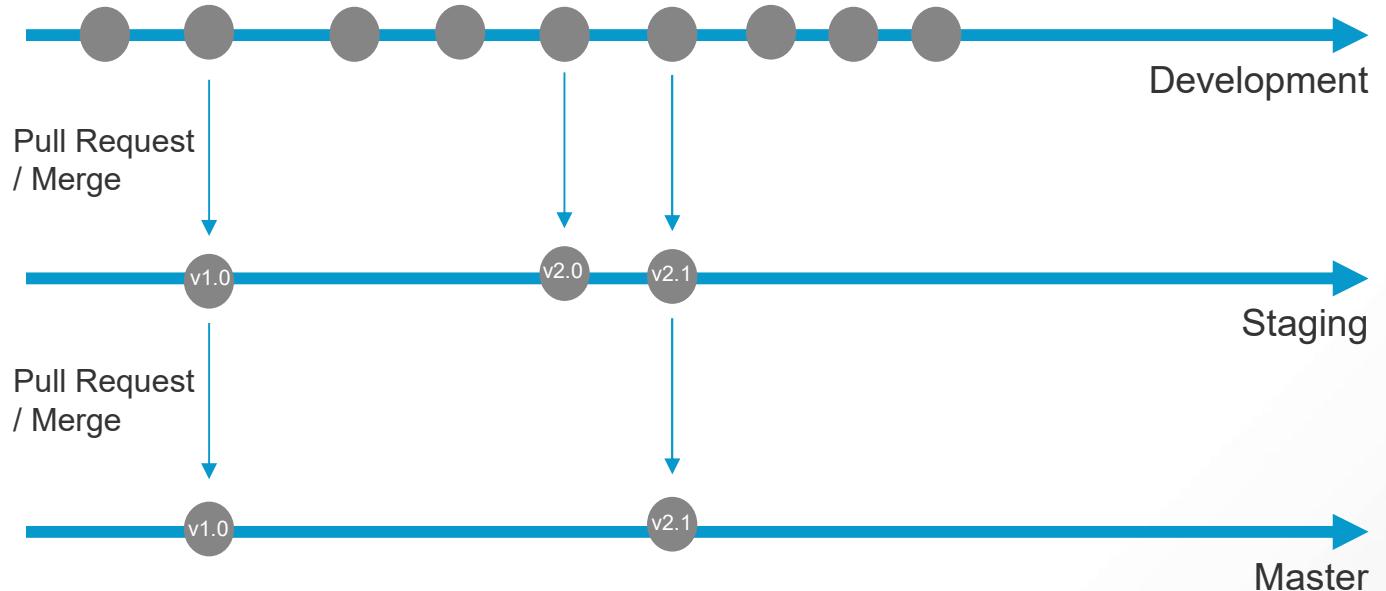
At the bottom left, the text "Version: 10.4.0.0.510" is visible. On the right, the Software AG logo is displayed with the tagline "Freedom as a Service".

API METRICS



GIT STRATEGY

INTO PRODUCTION – AND OFF WE GO AGAIN..



THANK YOU

3 KEY TAKEAWAYS

1

DevOps creates hands off automated quality gates through the CI process, and hands off, but manually approved automated test/production deployments to allow you to release at a much quicker pace whilst minimizing risk

2

Micro API Gateways in combination with a full API Gateway are a powerful mechanism to ensure your microservices remain focused on the task at hand and offload ancillary and sometimes complex tasks to avoid complicating the micro service

3

Kubernetes offers really powerful infrastructure as code and deployment techniques to allow you to truly update with ZERO downtime, and minimal fuss in doing so



WEBMETHODS API Gateway



kubernetes

<https://github.com/dpembo/APIWorld-Product>

