

Docker and Regression Testing

NOVA DevOps Meetup

Doug Toppin

dougtoppin@gmail.com

@dougtoppin

23-Mar-2017

Agenda

- Regression testing
- Regression test tools
- Example

Regression testing

- Confirm that something works
- Confirm that something still works (after I changed it)
- Confirm that what I added works as expected
- Provides insight on how something works (working examples of using an interface)
- Provides insight on the flow of how something works
- Developers should be able to run locally or in the cloud anything required anytime they want to without complicated stuff already being in place (services, databases/schemas)

Regression test tools

- docker/docker-compose
 - Spin up what you need when you need it and then get rid of it
 - Don't install stuff that you can run in containers (so that you don't have to manage versions, space requirements)
 - Test using different versions (the latest NodeJS or MongoDB for example) by just changing a version tag
- Postman
 - ReST interface requests and tests (UI)
 - Request results can be checked
 - Result data can be extracted and saved as environment variables
 - Requests can be chained
 - Good example of a vendor providing tooling support for learning/testing
<https://www.quovo.com/blog/postman-and-quovo/>
- Newman
 - Run Postman collections from the shell

docker-compose

- Run multi-container applications
- Define the services that you need
- Orchestrate how to run them
- Configure network connections (internal, container to container, external)
- Start/stop them with a single command

docker-compose

```
version: '2.1'
services:
  infinispn:
    build: .
    container_name: infinispn
    ports:
      - "127.0.0.1:8080:8080"
      - "127.0.0.1:11222:11222"
    healthcheck:
      test: ["CMD", "curl", "-f", "http://localhost:8080"]
      interval: 1s
      timeout: 1s
      retries: 3
    test:
      depends_on:
        infinispn:
          condition: service_healthy
      image: postman/newman_ubuntu1404
      volumes:
        - ./etc/newman
      command: run /etc/newman/InfinispnDockerNewmanExample.json.postman_collection
```

Example

```
#!/bin/sh
# start up the cache server in the background so that we can continue to the next
step
docker-compose up -d infinispn

# execute the tests in the postman collection
docker-compose up test

# tell the cache to shutdown
docker-compose exec infinispn sh /opt/jboss/infinispn-server/bin/ispn-cli.sh -c
command="shutdown"

# play it safe by making sure that we do not leave anything running
docker-compose down
```

Example

- <https://github.com/dougtoppin/docker-infinispan-example>
- Example of using Docker containers to run regression test on a ReST interface
- Requires
 - docker
 - docker-compose (supporting \geq version 2.1 files)
- Try this
 - git clone <https://github.com/dougtoppin/docker-infinispan-example>
 - cd docker-infinispan-example
 - ./regtest-compose.sh

Postman/Newman output

Requests

put example

PUT http://infinispan:8080/rest/default/a02 [200 OK, 144B, 115ms]

Status code is 200

get example

GET http://infinispan:8080/rest/default/a02 [200 OK, 255B, 12ms]

Status code is 200

Body is correct

get - negative test

GET http://infinispan:8080/rest/default/somethingnotfound [404 Not Found, 151B, 5ms]

Status code is 404

Postman/Newman output

	executed	failed
iterations	1	0
requests	3	0
test-scripts	3	0
prerequisite-scripts	0	0
assertions	4	0
total run duration: 244ms		
total data received: 9B (approx)		
average response time: 44ms		

Links

- Docker - <https://www.docker.com/>
- Postman - <https://www.getpostman.com/>
- Newman - <https://github.com/postmanlabs/newman>