Dataverse Scaling: Sprint 5 Demo

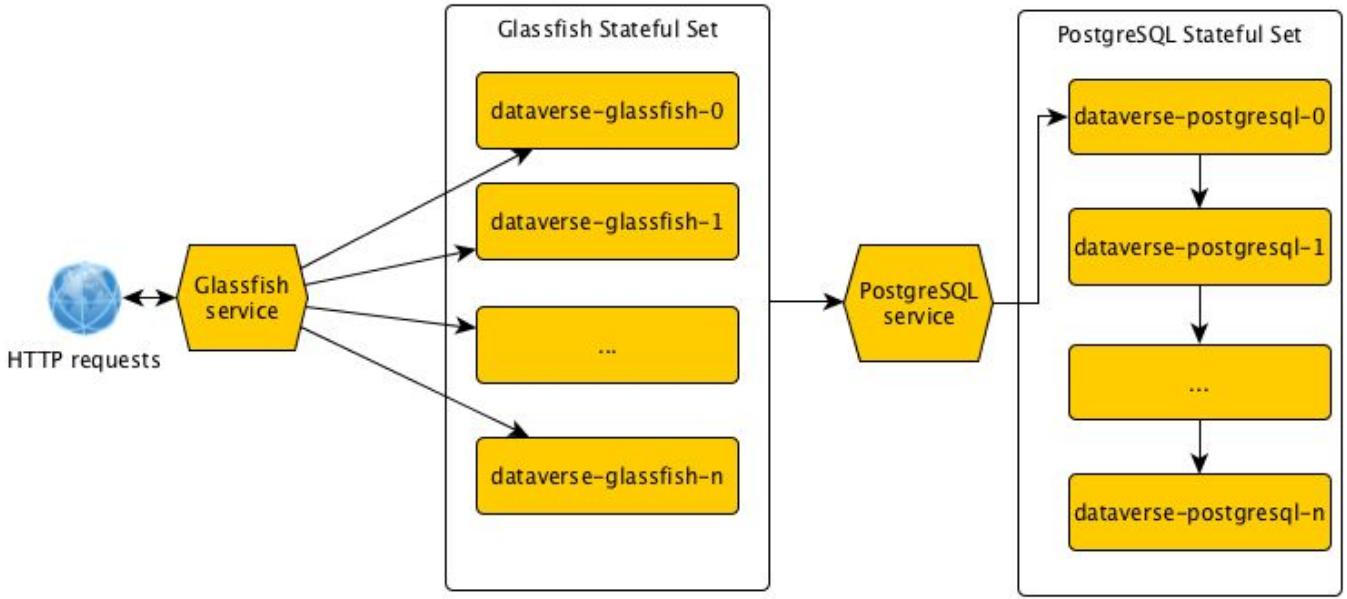
Students: Michael Clifford, Patrick Dillon, Ryan Morano & Ashwin Pillai

Mentors: Phil Durbin (Harvard), Dan McPherson & Solly Ross (both Red Hat)

Reminder of Project Goals & Scope

- Dataverse was developed as an N-tier web app
  - 1 HTTP server - Glassfish
  - 1 Database - Postgres
  - 1 Search Indexer - Solr
- Collaboration w/ Red Hat moved these components to Docker images
- Our project is to continue this work and create a configuration where Dataverse can scale these components on OpenShift

System overview



Work completed

- Set up and configured replicated PostgreSQL using Centos image
- Successful Deployment Dataverse to MOC's Openshift Container Platform.
- Updated RAM limits for Applications in Pods on MOC
- Began testing with Jmeter

MOC Deployment

- The ultimate goal of our project is to be able to deploy a scalable dataverse on the MOC.
- We have deployed our dataverse images and config.json on the MOC
- Successfully ported working minishift code to MOC openshift for glassfish.

MOC Deployment

https://openshift.massopen.cloud/console/project/dvs/overview

MOC Deployment

- Resource limitations on pods prevented us from successfully deploying on the MOC for our last sprint. 2Gb was insufficient.
- After increasing pod limit to 4Gb, application deployed successfully.

Postgres DB: Switching from Centos to Patroni back to Centos

- Last sprint we talked about moving from Centos to Patroni Postgresql images
- After discussing progress with mentors, they thought Patroni was too complex and progress was insufficient
- They gave us a new example which led us back to the original Centos image

Postgres DB: Centos solution

- All Docker images have a default command or entrypoint which runs at startup
- In OpenShift/Kubernetes, we specified a new command, which overrides the default command
- So at startup we check whether the pod should run as master or slave

Centos postgres new startup command

```
[[ `hostname` =~ -([0-9]+)$ ]]                    #get id created by statefulset
ordinal=${BASH_REMATCH[1]}                        # e.g. dataverse-postgresql-0
if [[ $ordinal -eq 0 ]]; then                     # if id == 0 it is master
      run-postgresql-master                       # run master binary
else                                              # if id != 0 it is a slave
     run-postgresql-slave                         # these binaries were already
fi                                                # in centos image
```

DEMO

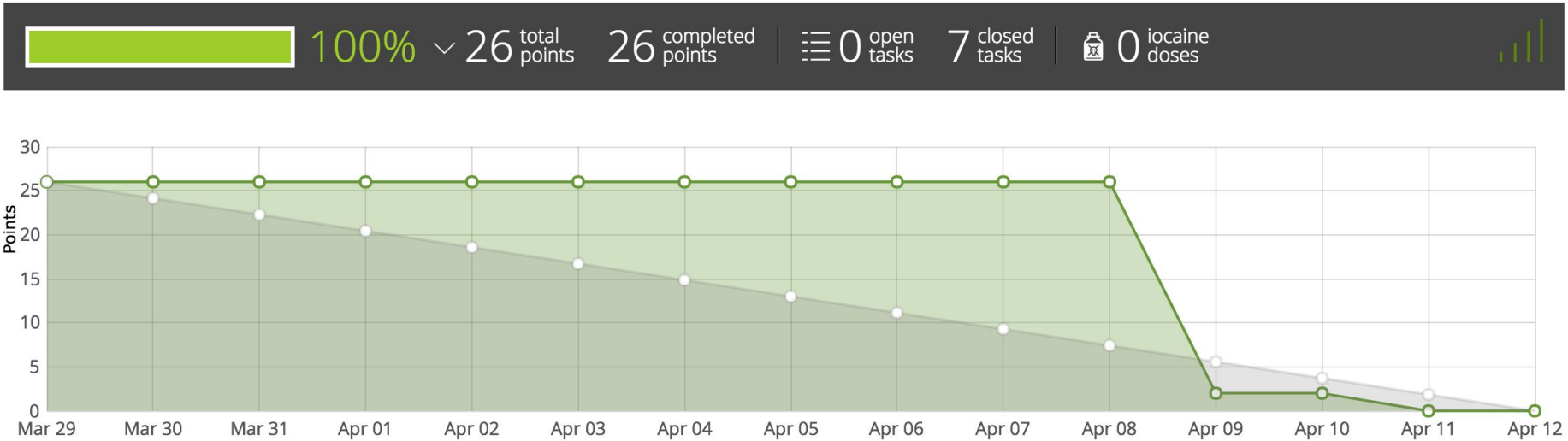
Load testing Dataverse

- What is load testing ?
- Tool - Jmeter

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
968	00:53:45.422	Users of Datav...	User on login p...	211		3260	177	211	106
969	00:53:45.393	Users of Datav...	User on login p...	242		3260	177	242	135
970	00:53:45.409	Users of Datav...	User on login p...	228		3260	177	226	119
971	00:53:45.434	Users of Datav...	User on login p...	204		3260	177	204	95
972	00:53:45.444	Users of Datav...	User on login p...	196		3260	177	196	85
973	00:53:45.455	Users of Datav...	User on login p...	188		3260	177	188	84
974	00:53:45.465	Users of Datav...	User on login p...	180		3260	177	179	74
975	00:53:45.482	Users of Datav...	User on login p...	224		3260	177	224	124
976	00:53:45.501	Users of Datav...	User on login p...	205		3260	177	205	105
977	00:53:45.473	Users of Datav...	User on login p...	234		3260	177	234	130
978	00:53:45.494	Users of Datav...	User on login p...	214		3260	177	214	112
979	00:53:45.514	Users of Datav...	User on login p...	203		3260	177	202	98
980	00:53:45.525	Users of Datav...	User on login p...	195		3260	177	195	100
981	00:53:45.533	Users of Datav...	User on login p...	223		3260	177	220	109
982	00:53:45.553	Users of Datav...	User on login p...	205		3260	177	205	90
983	00:53:45.541	Users of Datav...	User on login p...	231		3260	177	217	102
984	00:53:45.573	Users of Datav...	User on login p...	200		3260	177	200	70
985	00:53:45.562	Users of Datav...	User on login p...	212		3260	177	212	82
986	00:53:45.605	Users of Datav...	User on login p...	172		3260	177	171	101
987	00:53:45.592	Users of Datav...	User on login p...	185		3260	177	185	114
988	00:53:45.582	Users of Datav...	User on login p...	197		3260	177	196	124
989	00:53:45.615	Users of Datav...	User on login p...	213		3260	177	212	120
990	00:53:45.637	Users of Datav...	User on login p...	193		3260	177	193	116
991	00:53:45.629	Users of Datav...	User on login p...	204		3260	177	203	124
992	00:53:45.647	Users of Datav...	User on login p...	228		3260	177	228	129
993	00:53:45.662	Users of Datav...	User on login p...	213		3260	177	213	114
994	00:53:43.877	Users of Datav...	User on login p...	2668		3260	177	2666	1184
995	00:53:43.871	Users of Datav...	User on login p...	2682		3260	177	2679	1190
996	00:53:43.857	Users of Datav...	User on login p...	2712		3260	177	2711	1204
997	00:53:43.847	Users of Datav...	User on login p...	2755		3260	177	2751	1224
998	00:53:43.838	Users of Datav...	User on login p...	2775		3260	177	2773	1234
999	00:53:43.814	Users of Datav...	User on login p...	2807		3260	177	2807	1247
1000	00:53:43.804	Users of Datav...	User on login p...	2852		3260	177	2850	1267

Sprint 6 Burndown

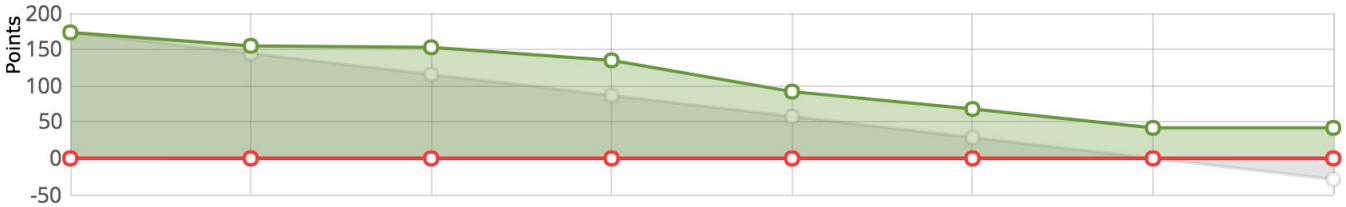
2018 BUCS528 DATAVERSE SCALE BU CS 528 CLOUD COMPUTING - DEMO 5 29 MAR 2018-12 APR 20



BOSTON UNIVERSITY

Project Burndown & next sprint

2018 BUCS528 DATAVERSE SCALE BACKLOG



- Create separate pull requests for Postgres & Glassfish
  - Glassfish is production ready
  - Postgres will need more work
- Merge Postgres & Glassfish then deploy on MOC
- Test a reasonable load using JMeter

Release Planning

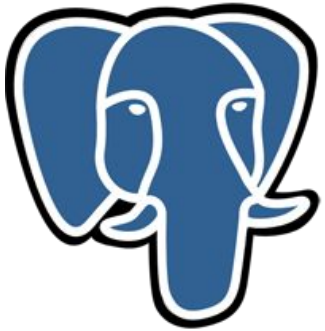
https://tree.taiga.io/project/msdisme-2018-bucs528-template-6/

THANKS!!

The Dataverse Project

Solr

redhat



PostgreSQL