

BlueBRIDGE - Project Activity #5823

Project WP # 641 (Closed): WP5 - Supporting Blue Assessment: VREs Development [Months: 1-30]

Project Task # 642 (Closed): T5.1 Stock Assessment VRE [Months: 1-30]

Project Activity # 1674 (Closed): VRE Stock Assessment Workplan

Project Activity # 1679 (Closed): ICES Stock Assessment VRE

Task # 1778 (Closed): Toolbox for Data Limited Stocks

Task # 5822 (Closed): Simple minimal example of Shiny App with ShinyProxy

Able to deploy docker containers

Nov 21, 2016 02:08 PM - Scott Large

Status: Closed	Start date: Mar 15, 2017
Priority: Normal	Due date:
Assignee: _InfraScience Systems Engineer	% Done: 100%
Sprint: WP04	
Participants:	
Description https://docs.docker.com/ I am unable to initiate docker containers on my local machine. Would it be possible to have a docker instance on the e-infrastructure, such that users (i.e., @julien.barde@ird.fr and myself) can dump ui.r and server.r files and have an operational shinyproxy server?	
Subtasks: Project Activity # 7493: Linking work - space with VREs via R Closed	
Related issues: Blocks BlueBRIDGE - Task # 7492: Use LBI application as shinyproxy use-case Closed Mar 15, 2017	

History

#1 - Nov 21, 2016 04:06 PM - Pasquale Pagano

- Sprint changed from UnSprintable to WP04
- Assignee set to _InfraScience Systems Engineer
- Project changed from D4Science Infrastructure to BlueBRIDGE
- Tracker changed from Task to Project Activity

We currently do not provide docker support. However, It is planned in the BlueBRIDGE activities in WP4.

So I have two questions for my colleagues:

- InfraScience Managers: can we provide a machine with docker? The only solution I can imagine is that
 - we provide the server with SG that is responsible of the Authentication and Authorisation
 - the user uploads the ui.r and server.r files on the workspace
 - the user triggers a data transfer job to that server by using the user token
 - the job has a post transfer job to run locally to that server that configure docker

This pattern is generic and if feasible then it can be reused also for other software running in docker.

- W4 leader, @gabriele.giammatteo@eng.it, to report on the support ENG can provide to support it and by when.

#2 - Nov 21, 2016 06:25 PM - Gabriele Giammatteo

From a preliminary internal discussion I had with @daniele.pavia@eng.it (who has some knowledge and experience with Docker) it seems feasible to

use Docker in the way you are proposing. We can provide support in building the Docker images and the jobs to make it run with the custom R files provided by the users. This should be achievable before the Christmas holidays however, to have a more clear planning, I think a call with InfraScience Managers to understand better the work to do is needed.

ENG can also provide support designing a more generic solution (I'm thinking to a service) that wraps this initial solution to allow users to use "any" containerized software with custom configurations, but this is for sure a more long-term activity that needs much more effort. I do not think we will be able to work on this before the review.

#3 - Dec 12, 2016 11:07 AM - Daniele Pavia

- % Done changed from 0 to 20

Dear @scott.large@ices.dk , @julien.barde@ird.fr , I'm working on an ansible role to deploy docker, shinyproxy and custom shinyapps. I'd like to have one of your shinyapps as a sample to test my work out. I need an application.yml, dump ui.r and server.r sample, can you guys help me out?

#4 - Dec 12, 2016 12:17 PM - Scott Large

Thanks, @daniele.pavia@eng.it!

Maybe it will be best to start with a very simple and minimal example. The ShinyProxy [Getting Started](#) guide provides a [demo](#) that might be a good thing to try and reproduce. I'm not sure how you like to work, but maybe we can clone that template to github or something, make the necessary changes for the Docker container and shinyproxy application.yml and then @julien.barde@ird.fr and I could try to clone the working example with our more complex Shiny apps. Let me know if this is useful? Thanks in advance, I'm excited to see how this will work.

SL

#5 - Dec 12, 2016 02:26 PM - Daniele Pavia

I was actually using "docker pull openanalytics/shinyproxy-demo" to test up to now. It looks like it's working - the server is up and running - but I can't login as it tries to authenticate against an external ldap for some reason. I've changed the application.yml accordingly (to enable "simple" auth) but there's something wrong in there, as the server itself can't parse the application.yml after I modified it.

TLDR; If you can give me a custom docker image or dockerfile plus the relative application.yml (I'm fine if it's based on the openanalytics/shinyproxy-template) I should be good to go.

#6 - Dec 14, 2016 11:26 AM - Scott Large

- % Done changed from 20 to 40

Thanks Daniele,

I think the default username (tesla) and password (password) should work for the authentication on the demo.

Please see the .zip file: <https://goo.gl/s6A8C7>. The app works on my local machine, so hopefully the settings are easy to dump onto the BB machine. I have removed authentication (auth: none) just to make things easier.

My workflow:

- 1) Changed the tcp to 0.0.0.0:2375 (per [getting started](#))
- 2) Modified the Dockerfile according to the R package dependencies for each app
- 3) Modified the application.yml file with the name, description, image, and docker-cmd information for each shiny app.
- 4) Built a docker image for each shiny app (d1m, StockTrends, and euler) docker build -t <shinyapp> .
- 5) Test that the app is working locally docker run -it -p 3838:3838 <shinyapp> at [localhost:3838](#)
- 6) Deploy locally with shinyproxy java -jar "shinyproxy-0.7.8.jar"
- 7) Test apps locally with shinyproxy at [localhost:8080](#)

#7 - Jan 19, 2017 12:03 PM - Daniele Pavia

- % Done changed from 40 to 70

I've developed a couple of ansible roles that are capable of deploying shinyproxy and shinyapps such as those provided by [@scott.large@ices.dk](mailto:scott.large@ices.dk) .

Notice:

- 1) the ansible role that deploys shinyproxy requires an application.yml that's already configured for all the shinyapps that are/need to be deployed. The applications.yml has not been reduced to a template.
- 2) the shinyapps need to be provided as an archive (one per app) that contains both the dockerfile and all the files that are going into the container image.

These ansible roles have been developed for CentOS/RHEL and Ubuntu 14.04, no extensive testing has taken place for Ubuntu though.

I need a way to provide these roles to CNR, either by having write access to the gitlab repo or by sharing them via ftp/mail/file hosting cloud services and I need someone @ CNR to test them out on their infrastructure.

No smartgears service has been developed (yet).

#8 - Jan 19, 2017 01:27 PM - Andrea Dell'Amico

Daniele Pavia wrote:

Notice:

- 1) the ansible role that deploys shinyproxy requires an application.yml that's already configured for all the shinyapps that are/need to be deployed. The applications.yml has not been reduced to a template.
- 2) the shinyapps need to be provided as an archive (one per app) that contains both the dockerfile and all the files that are going into the container image.

These ansible roles have been developed for CentOS/RHEL and Ubuntu 14.04, no extensive testing has taken place for Ubuntu though.

I need a way to provide these roles to CNR, either by having write access to the gitlab repo or by sharing them via ftp/mail/file hosting cloud services and I need someone @ CNR to test them out on their infrastructure.

No smartgears service has been developed (yet).

[@daniele.pavia@eng.it](mailto:daniele.pavia@eng.it) you now have now write access to the ansible-playbooks repository. You'll find a library/centos/roles directory, you can add yours there. We have a docker role for Ubuntu, I used it to test sharelatex. Can we build the shiny* roles on top of it?

#9 - Jan 19, 2017 02:40 PM - Daniele Pavia

@andrea.dellamico@isti.cnr.it I've included a very basic docker role for both Centos 7 and Ubuntu 14.04 - they only real configuration needed for shinyproxy is to enable the tcp socket for the docker daemon/service, which is done in different ways between Centos 7 and Ubuntu 14.04. We may merge our docker roles if it makes sense, or we may just rename my role to shinyproxy-docker is those roles are going to run on purposed machines (not running other docker instances). I'd rather go for the first option though.

#10 - Jan 19, 2017 03:22 PM - Andrea Dell'Amico

Daniele Pavia wrote:

I've included a very basic docker role for both Centos 7 and Ubuntu 14.04 - they only real configuration needed for shinyproxy is to enable the tcp socket for the docker daemon/service, which is done in different ways between Centos 7 and Ubuntu 14.04. We may merge our docker roles if it makes sense, or we may just rename my role to shinyproxy-docker is those roles are going to run on purposed machines (not running other docker instances). I'd rather go for the first option though.

The docker role I created only installs and activate docker (running as the docker user and not directly as root).

The tcp socket enabling can be part of the docker role and we can use a variable to enable it (default as False?).

If there are more services specific parts, they should go on dedicated roles that will be executed after the docker role.

#11 - Feb 14, 2017 08:36 AM - Leonardo Candela

@scott.large@ices.dk replied to this ticket by the following email ...

Hi all,
What is the status of the ticket? I am a bit lost on the jargon. Is there a service available where I can start initiating docker instances?

Thanks in advance,

SL

I kindly ask @daniele.pavia@eng.it and @andrea.dellamico@isti.cnr.it to update the ticket thus to report on the current status of this activity.

#12 - Feb 16, 2017 12:00 PM - Daniele Pavia

@leonardo.candela@isti.cnr.it @scott.large@ices.dk @andrea.dellamico@isti.cnr.it Sorry for all the jargon, I tend to get carried away easily ;)

So, here's the state of the art: I pushed the ansible roles I wrote to CNR's gitlab repo. As of now those haven't been tested against their environment yet (as far as i know). Once that's done and shinyproxy actually gets deployed it's only a matter of giving [@scott.large@ices.dk](mailto:scott.large@ices.dk) access to shinyproxy's web interface.

I'm still not sure whether developing a restful service makes sense here, so I've got to ask [@scott.large@ices.dk](mailto:scott.large@ices.dk): do you expect to update and (re)deploy the shinyapps frequently or is it just a one shot deployment and you only need to access that specific version of the euler/DLM/stocktrends? If those services need to get update every once in a while I'd say that redeploying them thru ansible seems to me the most sensible choice. On the other hand, if you expect/need to update the shinyapps fequently then developing a smartgears service make sense.

[@andrea.dellamico@isti.cnr.it](mailto:andrea.dellamico@isti.cnr.it) maybe we can try to run the shinyapps ansible roles one one of your VMs in the meantime?

#13 - Feb 17, 2017 08:14 AM - Scott Large

Thanks [@daniele.pavia@eng.it](mailto:daniele.pavia@eng.it) for the feedback. I think the goal will be to have operational shiny apps that will be deployed for a relatively long period of time before they are replaced with newer versions. I have no experience with ansible, but if you can provide a clear workflow for how to deploy, I can try to figure it out. If a shiny app is built on data that can change from time to time, would we need to redeploy every time the data is changed, or is there a clever way to link the data with the docker instance.

Thanks!

SL

#14 - Apr 06, 2017 04:13 PM - Andrea Dell'Amico

- Blocks Task #7492: Use LBI application as shinyproxy use-case added

#15 - Apr 06, 2017 04:14 PM - Andrea Dell'Amico

- Status changed from New to In Progress

We now have the infrastructure in place and the demo apps run.

#16 - May 26, 2017 12:22 PM - Daniele Pavia

just a reminder to [@scott.large@ices.dk](mailto:scott.large@ices.dk) that [@andrea.dellamico@isti.cnr.it](mailto:andrea.dellamico@isti.cnr.it) is waiting for his feedback about the shinyproxy instance deployed at CNR. We're stuck until we can actually get some feedback.

#17 - Jun 05, 2017 06:05 PM - Andrea Dell'Amico

- Status changed from In Progress to Feedback

In #7492 [@nathan.vaughan1@gmail.com](mailto:nathan.vaughan1@gmail.com) successfully built and pushed a docker image into the docker hub. It seems the best solution to me if we do not have a need for more privacy.

#18 - Jun 05, 2017 06:59 PM - Nathan Vaughan

[@andrea.dellamico@isti.cnr.it](mailto:andrea.dellamico@isti.cnr.it) Great, I'm glad it's up and running. I checked it out and all 3 apps are working for me now. I agree now that I've figured it out that the docker hub route seems the best way to get these apps published. The automated builds are good with a full error log as it took a few tweeks over the weekend to get everything loading correctly in the image. Will the image hosted on the VRE automatically update if it is rebuilt on docker hub? That would make it easy to update apps and even include new apps, without any overhead to CNR, if we can build more than one into a single image. Cheers, Nathan.

#19 - Jun 05, 2017 07:04 PM - Andrea Dell'Amico

Nathan Vaughan wrote:

The automated builds are good with a full error log as it took a few tweeks over the weekend to get everything loading correctly in the image. Will the image hosted on the VRE automatically update if it is rebuilt on docker hub?

I have no mechanism in place (yet?) but I think it's doable.

That would make it easy to update apps and even include new apps, without any overhead to CNR, if we can build more than one into a single image. Cheers, Nathan.

That part still requires a manual intervention: every application inside a container image must be explicitly listed into the shinyproxy configuration. Not a big deal, it's four lines of data, but without them the application does not show up

#20 - Jun 05, 2017 07:15 PM - Nathan Vaughan

Thanks Andrea, Makes sense that's the 4 lines you referenced before correct? name, displayname, CMD, and image location.

#21 - Jun 05, 2017 07:20 PM - Andrea Dell'Amico

Nathan Vaughan wrote:

Thanks Andrea, Makes sense that's the 4 lines you referenced before correct? name, displayname, CMD, and image location.

The ones for LBI are

```
- name: shinyproxy-lbi
display-name: LBI shiny app
description: LBI shiny app
docker-cmd: ["R", "-e shiny::runApp('/root/LBI_shiny')"]
docker-image: nathanvaughan/shinyproxy-lbi
```

An additional field called groups: can be added if an application execution must be restricted to a group of users (from ldap, so think about VRE members).

#22 - Jun 23, 2017 01:32 PM - Andrea Dell'Amico

Closing this one.

#23 - Jun 23, 2017 01:32 PM - Andrea Dell'Amico

- *Status changed from Feedback to Closed*