

Title: Issue with spring cloud dataflow and Remote Repository: Apps are installed but i can't deploy streams

Post Body:

i'm facing an issue using spring cloud dataflow connected to a remote repository.

I think i managed to connect the dataflow server to the repository correctly because at first i couldn't import apps and now i can

The problem is that when i try to deploy a stream the dataflow server doesn't see the remote repository.

Here's an example to make myself clear

When i try to import a jar that does not exist the import is successful but if i try to open the details from the UI i get:

Failed to resolve MavenResource: [JAR-NAME] Configured remote repositories: : [repo1],[springRepo]

So i guess that the system sees 'repo1'

But then when i deploy a stream (with all valid apps) i get:

Error Message = [Failed to resolve MavenResource: [JAR-NAME] Configured remote repository: : [springRepo]]

I followed this: <https://github.com/spring-cloud/spring-cloud-dataflow/issues/982>

And this: <https://docs.spring.io/spring-cloud-dataflow/docs/1.1.0.BUILD-SNAPSHOT/reference/html/getting-started-deploying-spring-cloud-dataflow.html>

This is my docker-compose.yml:

```
version: '3'
services:
  kafka:
    image: wurstmeister/kafka:2.11-0.11.0.3
    expose:
      - '9092'
    environment:
```

Accepted Answer: None

Highest Rated Answer:

The alternative way to set maven repository is SPRING_APPLICATION_JSON environment. See [here](#) for the official documentation. Docker compose line should be like this

```
- SPRING_APPLICATION_JSON={"maven":{"local-repository":"local","remote-repositories":{"repo1":{"url":"https://URL/repository/maven-releases/","username":"springRepo","password":"springRepo"},"springRepo":{"url":"https://spring.io/repository/maven-releases/","username":"springRepo","password":"springRepo"}}
```

To prevent access to default spring repository just replace repo1 to springRepo .