



Progress on Error Reporting Issue

Background

Abiquo Engineering team analyzed some historical open issues on our support portal and other related issues open in our internal bug tracking system, and then started a period of deep investigation in order to find the root cause generating these problems.

The aim of this document is to try to be transparent, present the current status of the investigation results and explain the actions that we are taking.

Three technical details to help understand the problem

We want to explain 3 details of Abiquo implementation that will help you to understand the problem and our investigations:

- When a user deploys or reconfigures a VM, this starts a process that begins on the Management server (to allocate resources, validate business rules, resource status, etc.), then continues in Virtual Factory, which performs the actual deploy/ reconfigure on the hypervisor. When this is completed, it returns to the Management server, and finalizes in an internal component called HANDLERS where Abiquo consolidates data and does some final steps.
- When you reconfigure, Abiquo creates a “fake tmp_vm” in the DB so that it can roll back if something goes wrong during the deploy. This temporary VM is only available during the reconfigure progress and after that, the handler component is responsible for deleting it after the reconfigure.
- If something happens with the handlers, then you will see an Alert Icon near the VM



The Symptoms

What symptoms have brought this matter to our attention and led us to open an internal investigation to understand the root cause?

- For deploys, our support has detected a number of tickets with VM correctly deployed or reconfigured but with the status LOCKED.
- For reconfigures, several tickets explain that the reconfigure has completed but the error icon appears near the VM, warning that something went wrong in the handler. Apart from that, the tmp_vm was not deleted in DB (Generating tmp_vm's in the DB that would never be eliminated)



- Apart from that, our QA team had some similar errors in our internal bug tracking that they were investigating

In the past (1.8.5 or previous version) these errors used to be for a transaction error raised by our ORM (hibernate) component, but this was addressed in 2.0 and the logs analyzed evidenced that the root cause was another type of error.

Because of all these symptoms and because there wasn't a formal pattern for generating these situations, we decided to dedicate resources, time and effort to investigate.

The error

After some investigations, stress tests and other actions, we could determine some situations to examine in more detail:

On deploy

When the deploy process is completed and the handler starts to consolidate some information in the DB, when we get the VM info from the DB, an unexpected NULL appears in the Hypervisor object. Obviously this situation should not be possible (a VM MUST have an hypervisor object to start the deploy process, and this hypervisor information is correctly populated in the DB). But again, it's not a consistent error. When executing a script that starts to do concurrent deploys/reconfigures and other actions in the platform, this appears maybe once in all the testing.

Some logs that shows the problem:

```
03:29:19.576 ERROR c.a.a.i.t.DatacenterNotificationsCallback - Error processing task 'DEPLOY' in
handler 'com.abiquo.api.amqp.listeners.t
arantino.handlers.impl.DeployEventHandler$$EnhancerByCGLIB$$d46492d3'
java.lang.NullPointerException
    at
com.abiquo.api.amqp.listeners.tarantino.handlers.impl.DeployEventHandler.onTaskFinishedSuccessfully(D
eployEventHandler.java:100)
[DeployEventHandler.class:na]
    at
com.abiquo.api.amqp.listeners.tarantino.handlers.impl.DeployEventHandler$$FastClassByCGLIB$$bc37d
0a6.invoke(<generated>) [cglib-
2.2.jar:na]
    at net.sf.cglib.proxy.MethodProxy.invoke(MethodProxy.java:191) [cglib-2.2.jar:na]
    at
org.springframework.aop.framework.Cglib2AopProxy$DynamicAdvisedInterceptor.intercept(Cglib2AopPro
xy.java:617) [spring-aop-3.0.5.
RELEASE.jar:3.0.5.RELEASE]
    at
com.abiquo.api.amqp.listeners.tarantino.handlers.impl.DeployEventHandler$$EnhancerByCGLIB$$d4649
2d3.onTaskFinishedSuccessfully(<g
enerated>) [cglib-2.2.jar:na]
```



On Reconfigure

The situation is similar, with the same stress tests and situations. But at this point, the problem is trying to delete the vm_tmp. We can see that the object is correctly stored in the DB and when Abiquo tries to get the tmp object in the handler, we receive a null object. This situation shouldn't be possible because it MUST create a vm_tmp during reconfigure. Again it's a "sometimes" error that appears very intermittently.

Some related logs:

```
javax.persistence.NoResultException: No entity found for query at
com.softwarementors.bzngine.engines.hibernate.CriteriaJpaHelper.getSingleResult(CriteriaJpaHelper.java:197) [bzngine-0.9.2-SNAPSHOT.jar:na]
at
com.softwarementors.bzngine.dao.hibernate.JpaHibernateDaoBase.getSingleResult(JpaHibernateDaoBase.java:175) [bzngine-0.9.2-SNAPSHOT.jar:na]
at
com.abiquo.server.core.cloud.VirtualMachineDAO.findBackup(VirtualMachineDAO.java:303) [model-2.0-HF5.jar:na]
at com.abiquo.server.core.cloud.VirtualMachineRep.findBackup(VirtualMachineRep.java:269) [model-2.0-HF5.jar:na]
at
com.abiquo.api.services.cloud.VirtualMachineService.getBackupVirtualMachine(VirtualMachineService.java:2880) [api-2.0-HF5-classes.jar:na]
```

Undeploy

We have not done any specific testing with undeploy yet, but based on the support tickets, it seems to be the same/similar situation as deploy.

How Abiquo responds in this situation and possible workarounds

On deploy

When the errors appears

- The task in Redis is marked as ABORTED (Because it's a handler error and it means that the deploy task is not 100% complete)
- VM remains LOCKED, because the last task in the handler is to subscribe to the Monitor system, Abiquo can't see that the real state is ON

Workaround:

Our Virtual Infrastructure check is responsible for checking if LOCKED VMs are really locked. If it detects that the deploy is correctly completed, it will subscribe the VM to monitor system and publish the correct ON, then the VM will update its status to ON and the vApp won't be locked.



On reconfigure

When errors appears

- The task in redis is marked as aborted.
- The error icon appears on VM (Not locked) because the last known task on the VM (the reconfigure task) has an aborted status.
- In the DB, there is a vm_tmp that has not been deleted after reconfigure.
- After that, any reconfigure on this VM is going to be done correctly on the hypervisor but the error icon is always going to appear because abiquo is going to generate a new vm_tmp. This causes inconsistency because the handler is waiting for only 1 vm_tmp but the DB has more than one, so Abiquo can't determine what vm_tmp to use to finalize handler tasks (Remember, the reconfigure task works fine but the error icon appears)

Workaround:

- First of all delete the tmp_vm related to the VM

First of all, determine what idVM is affected

```
#> select idVM from virtualmachine where temporal is not null and name = XXX;  
(Here can be more than one register, then, follow Second and final step for each idVM)
```

Secondly, delete related rasd information for this temporal vm

```
#> delete ipm, rm from ip_pool_management ipm, rasd_management rm where  
rm.idManagement = ipm.idManagement and rm.temporal is not null and rm.idVM = IDVM;  
#> delete dm, rm from disk_management dm, rasd_management rm where rm.idManagement =  
dm.idManagement and rm.temporal is not null and rm.idVM = IDVM;  
#> delete rm from rasd_management rm where rm.temporal is not null and rm.idVM = IDVM;
```

Finally, delete the tmp vm

```
#> delete virtualmachine where temporal is not null and name = XXXX;
```

This is going to solve the tmp problem and then, to delete the error icon we have to perform a successful task on the VM. To do that, for example, we can power off and then power on a VM, or pause and then power on a VM. In short, we need to do a task on the VM to move the last task to the SUCCESS status in Redis.

Next Steps

We have a group of engineers working on this problem. You have to understand that it's a complex situation because moving the platform to debug mode is not generating these errors because when the developers add a breakpoint, the error never appears.

But we are working hard to definitively solve it.

Our plan is to inform you when the problem seems to be solved and find the best solution for our customers. This makes it highly likely that a new HF for 2.0 will be released.



We are sorry for the inconvenience that this problem can cause. We are working hard to find a solution as soon as possible.