Introduction

This paper details the setup scenarios for using VMware vFabric Postgres 9.1 for use with Horizon Workspace service-va to implement this service in HA mode or for production environments where external PostgreSQL database is a requirement.

Problem description

When designing Horizon Workspace for production usage an external VMware VMware vFabric Postgres 9.1 is required.
VMware vFabric Postgres is *not included as part of Horizon Workspace licenses and should therefore be bought separately* from Horizon Workspace.
VMware vFabric Postgres is a new platform for most customers so little or no information exists on install, configure, scale and backup & restore operations of this platform.

Typically customer don't want to introduce a new DB platform - PostgreSQL is for most customers a new platform - and often this comes up as a show-stopper when discussing production implementations of Horizon Workspace as MSSQL or Oracle is the typical database deployed with enterprise customers which Horizon Workspace does not support in the initial release.

Solution

VMware has a PostgreSQL as part of the vFabric family that is as easy to deploy and manage. As with Horizon Workspace it comes in the form of an appliance. This in turn means that customer does not require to skill up people on VMware vFabric Postgres as install, configure and backup/restore is easy to do and Horizon Workspace would be the only pre-defined workload running on PostgreSQL.
This guide focuses on using VMware vFabric Postgres with Horizon Workspace specifically so all information in this document is relevant primarily around that specific setup.

This document discusses the following:

1. Deployment and configuration of VMware vFabric Postgres OVF appliance with vSphere
    - Correct sizing of the appliance (Disk, RAM and CPU)
    - Enabling 'citext' module and creating a user for the database
    - Backup & restore of both the appliance and the "saas" database
2. VMware vFabric Postgres 9.1 in High Availability (replication) mode

Option 1 protects the VMware vFabric Postgres using vSphere HA. Option 2 is a deployment option where VMware vFabric Postgres is configured in cluster mode so the service itself is highly available.

Tools used for operating VMware vFabric Postgres will also be discussed.

NOTE: This document does not discuss using the open source edition of PostgreSQL or manually installing VMware vFabric Postgres on any supported operating system. For information on this refer to official PostgreSQL information or the VMware vFabric Postgres documentation.

NOTE: When using VMware vFabric Postgres be sure to buy the Standard Edition and not the vFabric Data Director edition of VMware vFabric Postgres. See download link at the Links section.

Pre-reqs

- VMware vFabric Postgres appliance v. 9.1.0
- IP information to use with the appliance(s)
- 1 or more licenses for VMware vFabric Postgres
- Admin access to vCenter where the VMware vFabric Postgres appliance(s) is to be deployed
- Username and password to be used with the VMware vFabric Postgres appliance(s)
    - Both for the default 'postgres' account and the 'horizon' account to connect with Horizon Workspace
- Downloaded and installed pgAdmin (See links section below)

Install and setup VMware vFabric Postgres 9.1 - non HA mode

Walk through deploying the VMware vFabric Postgres appliance in the OVF wizard in vCenter specifying the required information. IP information and master password supplied here can be changed in the VAMI interface later if required.
Power on the appliance and validate that you can access the PostgreSQL service using eg. pgAdmin III.

After validating you can successfully login to the web-interface of the appliance, shut down the appliance from within vCenter and make the required changes match your sizing requirements. For supporting up to 30.000 users you should allocate the following: (Changes from the default

are highlighted in bold)

Resource	Value
vCPU	**4**
Memory	**8GB**
Hard Disk 1 - Root Disk	2GB
Hard Disk 2 - Data Disk	**32GB**
Hard Disk 3 - SWAP Disk	**16GB**
Hard Disk 4 - Diag. Disk	2GB

Power the appliance back on and validate that the changes have been made. To do this login to the appliance at the console as root and run the following commands:

```
cat /proc/cpuinfo | grep -e processor
cat /proc/meminfo
```

cpuinfo will show you how many processors are now in the VM and the meminfo will show you swap and allocated memory to validate the changes against.
To validate the disk expansion run the following command and validate that the disk sizes matches the changes made:

```
df -h
```

After validating the changes stay logged into the appliance via SSH as root and move on the next section of this guide.

Edit postgresql.conf settings

Some additional changes needs to be made to the postgresql.conf before VMware vFabric Postgres can be used with Horizon Workspace.

While still logged in as root to the VMware vFabric Postgres appliance edit the following file using your favourite editor:

```
/var/vmware/vpostgres/current/pgdata/postgresql.conf
```

Locate the line that begins with *max_connections = 100* and change that to:

```
max_connections = 600
```

Add a new line to the postgresql.conf file just under the max_connections line with the

following information:

```
search_path = 'saas'
```

Change directory into */opt/vmware/vpostgres/current/bin* and change to the 'postgres' user account by running:

```
su postgres
```

Verify that you are still in the */opt/vmware/vpostgres/current/bin* path by running *pwd*

The next step requires that a file is either copied (scp) to the VMware vFabric Postgres appliance or the following content can be copy/pasted directly into vi while you have the SSH session running.

```
CREATE ROLE horizon LOGIN
  PASSWORD 'Topsecret123'
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;

ALTER ROLE horizon;
SET search_path = saas;

CREATE DATABASE saas
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1;
GRANT CONNECT, TEMPORARY ON DATABASE saas TO public;
GRANT ALL ON DATABASE saas TO postgres;
GRANT ALL ON DATABASE saas TO horizon;

\connect saas;
CREATE SCHEMA saas AUTHORIZATION horizon;
CREATE EXTENSION citext SCHEMA saas;
```

Change *PASSWORD* to whatever password you want to use with the horizon account.
Save the file as */tmp/createdb.do*

Now run the following command to execute the *createdb.do* file:

```
./psql -f /tmp/createdb.do
```

After running the above command you should receive output similar to that of below:

CREATE ROLE

ALTER ROLE
CREATE DATABASE
GRANT
GRANT
GRANT
You are now connected to database "saas2" as user "postgres".
CREATE SCHEMA
CREATE EXTENSION

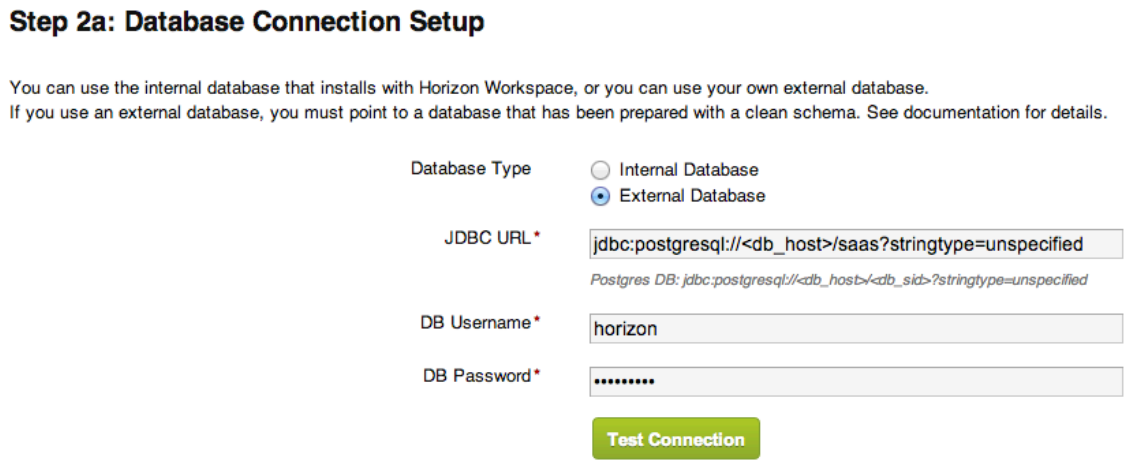
Reboot the VMware vFabric Postgres appliance.

Use pgAdmin to validate that the database, schema and user is setup correctly.

VMware vFabric Postgres appliance is now ready for use with Horizon Workspace.

When you get to step 2a in the VMware Horizon Workspace Setup Wizard chose "Database Type - External Database" and fill out the information as on the screenshot below:

NOTE: If using Postgres HA setup; when running the initial Horizon Workspace setup (Step 2a) it is important to specify the JDBC URL pointing *ONLY* to the master Postgres.
In the case of failure this URL is changed manually by an administrator to point to the slave should it become necessary.



Install and setup VMware vFabric Postgres 9.1 - HA replication mode

This document does not go into detail on how to setup and configure VMware vFabric Postgres in replication mode. Such a guide already exists on this specific topic over at the official VMware vFabric blog: Master-Slave Cluster with vFabric Postgres 9.1 on vSphere

After following the above guide you can complete the steps in this guide.
It will also be beneficial to fully understand WAL archiving, timelining and backup/restore scenarios. Refer to the official PostgreSQL documentation for material on these topics.

NOTE: After completing the HA setup and when running the initial Horizon Workspace setup using configurator it is important to specify the JDBC URL pointing *ONLY* to the master Postgres.
In the case of failure this URL is changed manually by an administrator to point to the slave should it become necessary.

Failover

Once the replication mode has been setup and is functioning use the following procedure to failover to the slave.

NOTE: Familiarise yourself with PostgreSQL best practices before performing failover operations. Failover operations typically means that the old master can never come online again and this impacts master/slave relationships as the slave effectively becomes the new master.

Promote the slave to being the new master (it will accept read and write operations) by running te following command on the slave:

```
pg_ctl promote
```

To check that the slave is out of recovery mode and functioning execute the following command from psql:

```
SELECT * FROM pg_is_in_recovery();
```

This should return "false" or f stating that the recovery is done.

Go to the Horizon Workspace Configurator URL and click on "Database Connection".

Change the JDBC to point to the newly promoted master - the node previosly functioning as slave - and click "Test Connection". A success message should be shown and after that click "Save".
Once the "Save" operation finished the tc-service has been restarted on all Service-VAs and Horizon Worskpace should now be functional again.

After the new JDBC configuration has been saved and pushed out to the Service-VA(s) it *might* be required to re-start the vApp.

Migrate database from service-va to external vFabric Postgres

If the Horizon Workspace installation was initially setup using the internally supplied vFabric Postgres you can backup the 'saas' database and restore to it to a production ready vFabric Postgres appliance.
NOTE: The procedure outlined in this section requires that the steps in *Install and setup VMware vFabric Postgres 9.1 - non HA mode* have been done so you have a fully functional vFabric Postgres appliance up and running with the saas database created as this procedure will restore against that database schema.
pgAdmin III is also required to be installed and available as this tool is used to perform backup and restore operations.

Service-VA configuration changes

Login to the service-VA with sshuser and su into root.

Change directory into */db/data*

Edit *pg_hba.conf* using vi and add a line at the very end of the file with the following content:

```
host    all             all             0.0.0.0/0               md5
```

This information should be entered as the very last line of text in the file. Exit with :wq! to commit the changes.

Edit */db/data/postgresql.conf* using vi and locate the line:

```
listen_addresses = 'localhost'
```

Change this line to:

```
listen_addresses = '*'
```

Exit with :wq! to commit the changes.

Restart the vFabric Postgres service for it to pick up on the configuration changes:

```
service vpostgres restart
```

Wait for the vFabric Postgres (vpostgres) service to come back online.

NOTE: If using Horizon Workspace 1.5 an additional step is required to allow remote access to the postgres service.

Login as root to the service-va and run the following command to allow incoming port 5432 (postgres) against the service-va. The command will open port 5432 until the service-va is rebooted:

```
iptables -A INPUT -p tcp --dport 5432 -j ACCEPT
```

To validate the change run the following command to list the local iptable rules:

```
iptables -nL -v --line-numbers | grep -i 5432
```

The output should be similar to the line shown below:

```
25 0 0 ACCEPT tcp -- *  * 0.0.0.0/0 0.0.0.0/0 tcp dpt:5432
```

Finish with validating the vpostgres service is up and running and available:

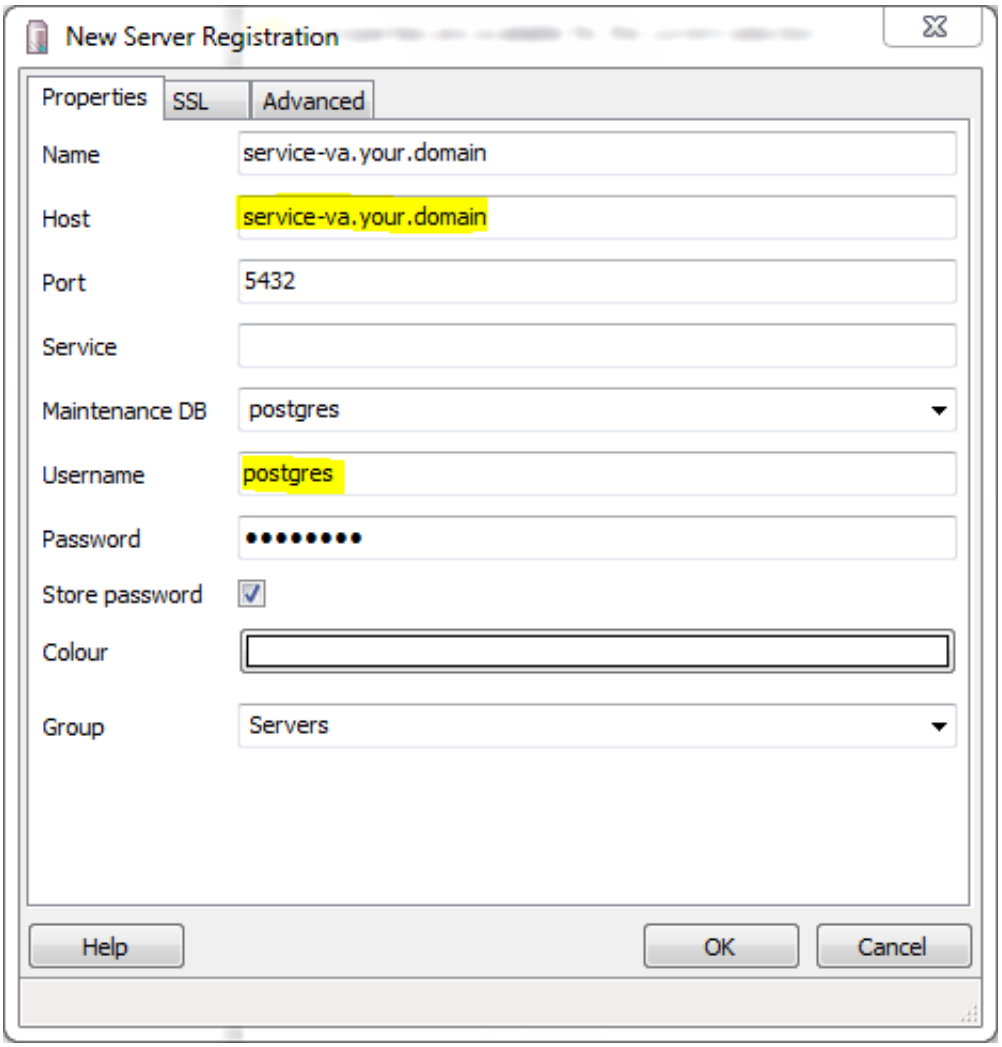
```
service vpostgres status
```

Backup 'saas' database from service-va

After the changes above are done connections can be made to the vFabric Postgres service running on the service-va to perform the backup.

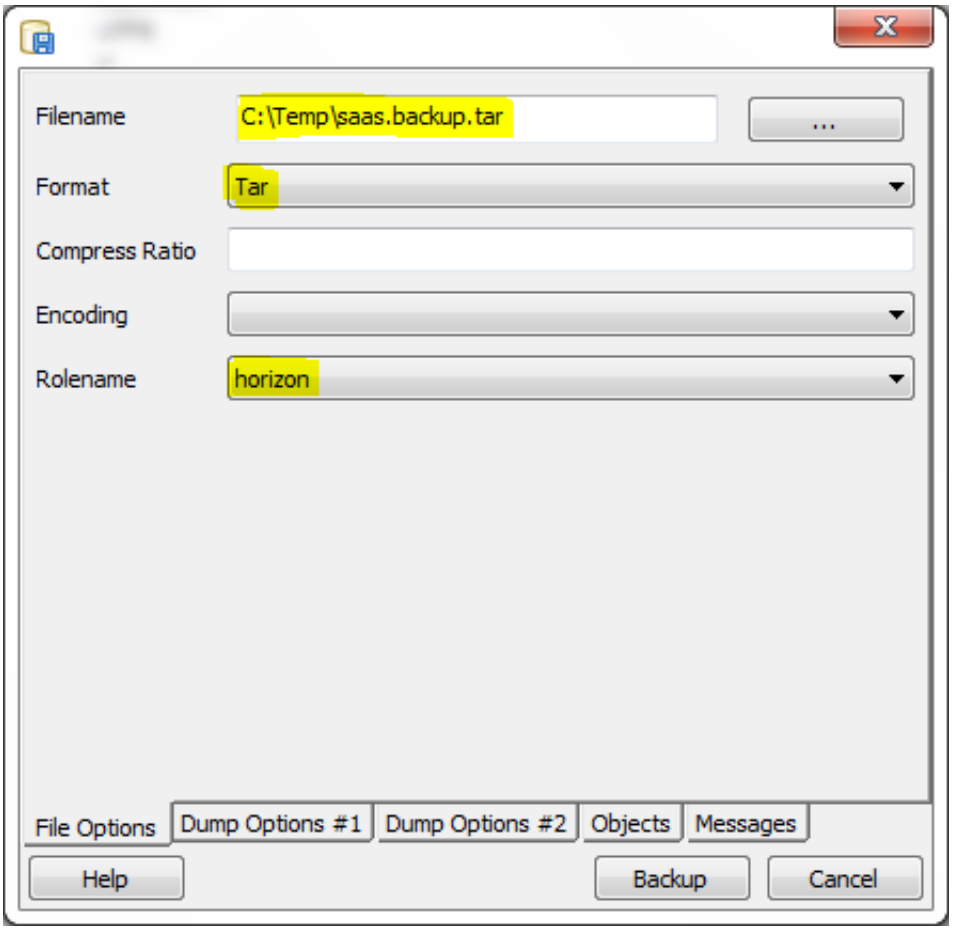
Connect to the service-VA using pgAdmin III with the following connection information:

Property	Value
Name	<service-VA-IP>
Host	<service-VA-IP>
Port	5432
Username	postgres
Password	H0rizon!



Once connected to the service-VA vFabric Postgres database navigate to Databases > saas and right click the 'saas' database and click 'Backup…'
Fill out the fields on the 'File Options' tab with the information below and click 'Backup'. No additional changes are required on any of the other tabs.

Property	Value
Filename	<path_to_backup>
Format	Tar
Rolename	horizon



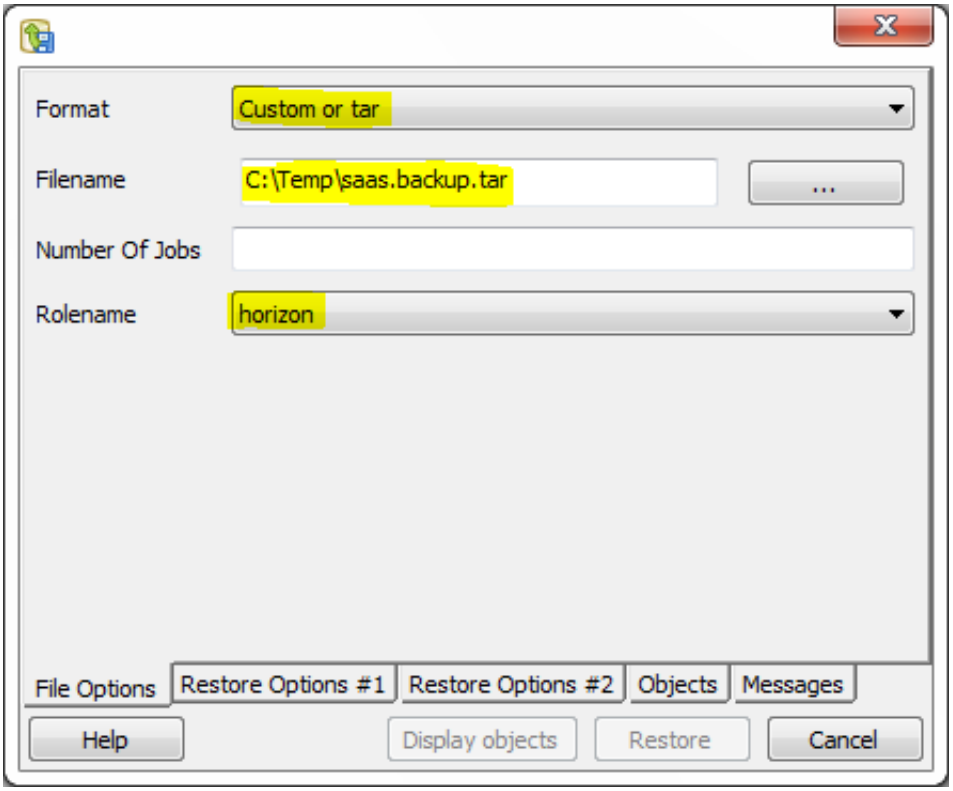
Once the backup is done go to the 'Messages' tab and verify that it states a clean exit code of '0' with no errors.

Restoring 'saas' database to production ready vFabric Postgres appliance

If not already done add your vFabric Postgres production appliance to pgAdmin III and connect using the postgres account.
Navigate to Databases > 'saas' right click the 'saas' database and click 'Restore…'
NOTE: Make sure you are performing this step on the vFabric Postgres production appliance and not against the service-va .

Fill out the fields on the 'File Options' tab with the information below and click 'Restore'. No additional changes are required on any of the other tabs.

Property	Value
Filename	<path_to_backup>
Format	Custom or tar
Rolename	horizon



Once the restore is done go to the 'Messages' tab and look for any errors. Most likely you will see the following at the beginning of the log:

```
pg_restore: [archiver (db)] Error while PROCESSING TOC:
pg_restore: [archiver (db)] Error from TOC entry 7; 2615 16386 SCHEMA saas
pg_restore: [archiver (db)] could not execute query: ERROR:  schema "saas"
    Command was: CREATE SCHEMA saas;

pg_restore: creating EXTENSION plpgsql
pg_restore: creating COMMENT EXTENSION plpgsql
pg_restore: [archiver (db)] Error from TOC entry 3818; 0 0 COMMENT EXTENSIO
pg_restore: [archiver (db)] could not execute query: ERROR:  must be owner
    Command was: COMMENT ON EXTENSION plpgsql IS 'PL/pgSQL procedural langu

pg_restore: creating EXTENSION citext
pg_restore: creating COMMENT EXTENSION citext
pg_restore: [archiver (db)] Error from TOC entry 3819; 0 0 COMMENT EXTENSIO
pg_restore: [archiver (db)] could not execute query: ERROR:  must be owner
    Command was: COMMENT ON EXTENSION citext IS 'data type for case-insensi
```

And the following at the end of the log:

```
WARNING: errors ignored on restore: 3
```

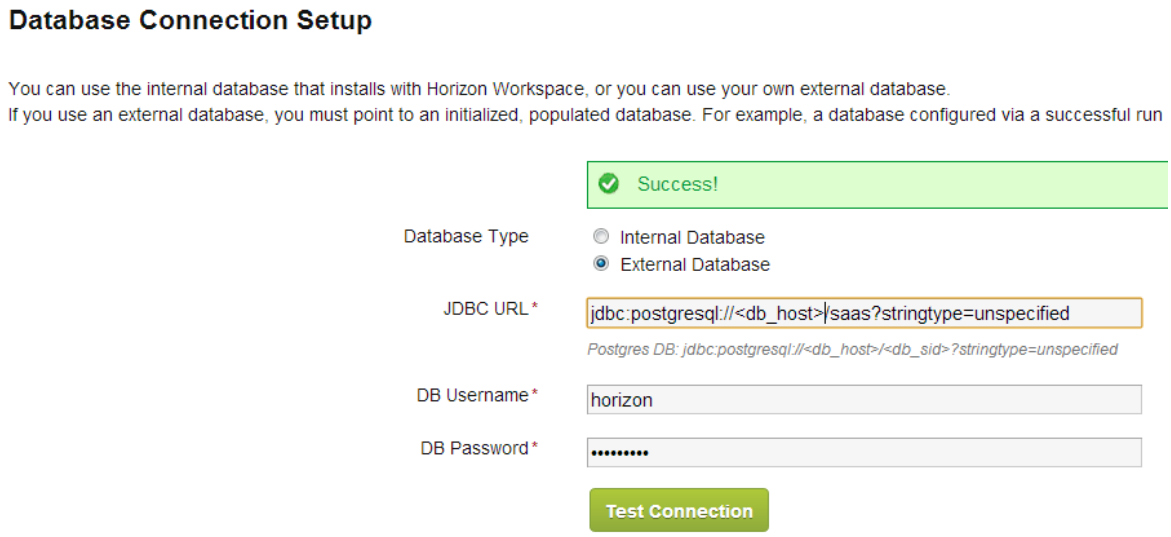
This is expected behavior as citext extension and saas schema has already been created priviously.

Change JDBC connection information

After successfully restoring the database the JDBC connection information should be changed to point to the new external vFabric Postgres appliance.

Login to the configurator-va and click on 'Database Connection' on the left side of the page.

Click on 'External database' and fill out the information as on the screenshot below.



Click 'Test Connection' and wait for the 'Success!' message to appear before clicking 'Save'. This might take a few minutes as everything is updated and the internal vFabric Postgres on the service-va appliance is stopped.

Backup scenarios

This section details the different options available for taking a backup of the Horizon Workspace database. For restore operations see the Restore scenarios section.
NOTE: When performing backup of the vFabric Postgres database try to avoid taking this backup at the same time as the connector-VAs sync from Active Directory.

Backup to a remote location using pg_dump

This approach is recommended only if your database size is 10GB or less due to the time it can take to do the backup over the LAN.
Before beginning this step download the VMware vFabric Postgres Standard Edition 9.1 Client Tools for the operating system you wish to perform the remote backup on.

Once downloaded change into the installation directory of the Client Tools and run the following command: (The below example uses the Windows Client Tools)

```
pg_dump -h <your-postgres-server> -U postgres -W --clean -f C:\<backup-path
```

Enter the password for 'postgres' when pg_dump prompts for it.

This will dump the entire 'saas' database with the 'clean' option meaning when performing a restore any existing 'saas' database will first be dropped.

Full VM backup of the VMware vFabric Postgres appliance

This approach is recommended if your database size os 10GB or more in size as the backup typically happens over the storage network.
NOTE: This requires additional backup software not included with VMware Horizon Workspace.

If using this approach verify with your backup software vendor how to perform a full consistent VM backup.

Restore scenarios

This section details the different failure scenarios you can experience and how you use the backup options to get the Horizon Workspace solution back up and running.
None of the restore scenarios described will discuss detailed restore operations like restoring a separate table or content in tables.

NOTE: VMware vFabric Postgres needs to be restored together with all the other components that make up VMware Horizon Workspace. For more information please see Horizon Workspace Backup Data Best Practices
This means that if your VMware vFabric Postgres appliance fails you need to restore all the components as mentioned in the order in the Horizon Workspace Backup Data Best Practices guide.

Rollback the database to a previous version

Use this restore option of the Horizon Workspace database is corrupt or information in the database was somehow deleted.

For this procedure to work you need to download and install the VMware vFabric Postgres Standard Edition 9.1 Client Tools on the operating system where you will perform the restore from.
Before continuing with the restore step verify that you can successfully connect to the VMware vFabric Postgres appliance using pgAdmin III or any other PostgreSQL client.

```
psql -h <your-postgres-server> -U postgres -W -d saas -f C:\<backup-path>\s
```

Enter the password for 'postgres' when psql prompts for it.

The screen output will show DROP and ALTER commands before the restore process begins. After running the restore verify that VMware Horizon Workspace can successfully connect to the database. This can be done from *https://<configurator-va>* and once logged in click "Database Connection" on the left side of the page and then click "Test Connection" to verify that database connectivity is working again.

Restore the entire database to a new VMware vFabric Postgres appliance

Use this restore option in the case you have either lost or corrupted the VMware vFabric Postgres appliance at a VM or OS level.

After following the steps in *Full VM backup of the VMware vFabric Postgres appliance* use your backup software to restore the VMware vFabric Postgres appliance and simply power it back online for VMware Horizon Workspace to connect to.
After restoring the VMware vFabric Postgres appliance verify that VMware Horizon Workspace can successfully connect to the database. This can be done from *https://<configurator-va>* and once logged in click "Database Connection" on the left side of the page and then click "Test Connection" to verify that database connectivity is working again.

Miscellaneous configuration options

This section details optional configuration options that can be performed to eg. tighten security or integrate with existing log tools etc.

Enable syslog to a remote syslog server

Default loggin option for VMware vFabric Postgres is 'csvlog' to the local disk at /var/vmware/vpostgres/current/pgdata/pg_log. In addition to 'csvlog' options locally you can enable syslog to a remote syslog server as well. This will allow both local 'csvlog' as well as the 'syslog' option in VMware vFabric Postgres.
This is a 2 step process where first the postgresql.conf file enables syslog and second the local syslog-ng services passes these logs to a remote syslog server

1 - Enable syslog in postgresql.conf

SSH into your VMware vFabric Postgres appliance as root and edit the following file using your favourite editor:

```
/var/vmware/vpostgres/current/pgdata/postgresql.conf
```

At the bottom of this file locate the following properties and change to look like the below:

```
log_destination = 'syslog,csvlog'
log_directory = 'pg_log'
log_filename = 'postgresql-%a'
log_rotation_age = 1440
log_truncate_on_rotation = on

syslog_facility = 'LOCAL0'
syslog_ident = 'postgres'
```

The *log_destination* is already present and needs to be changed to also have syslog option enabled.
syslog_facility and *syslog_ident* needs to be added manually.

2 - Configure syslog-ng to forward postgresql logs

Configuration option 1 will only enable VMware vFabric Postgres related logging to a remote syslog server. If you would also like to enable general OS (SLES) logging to the remote syslog server use configuration option 2

Configuration option #1

```
destination logserver   { udp("<your-syslog-server>" port(514)); };
filter f_postgres       { facility(local0); };
log { source(src); filter(f_postgres); destination(logserver); };
```

Configuration option #2

```
destination logserver   { udp("<your-syslog-server>" port(514)); };
log { source(src); destination(logserver); };

filter f_postgres       { facility(local0); };
log { source(src); filter(f_postgres); destination(logserver); };
```

Reboot the VMware vFabric Postgres appliance and go to your syslog server to validate successfull setup by looking for LOG entries from postgres.

External resources / links

- VMware VMware vFabric Postgres Documentation
- Download VMware vFabric Postgres Standard Edition 9.1
- Download VMware vFabric Postgres Standard Edition 9.1 - Client Tools
- Open Source PostgreSQL Documentation
- PGAdmin Download
- Master-Slave Cluster with vFabric Postgres 9.1 on vSphere
- Horizon Workspace Backup Data Best Practices

Changelog

Version 0.1

- Initial version including overview and general install procedure

Version 0.2

- Changed from DB-gui approach to TSQL script for creating users, database, schema and extension
- Added remote syslog configuration option
- Added how to change DB-gui certificate
- Added section on required changes to postgresql.conf file to work with Horizon Workspace

Version 0.3

- Changed to comply with naming guidelines
- Added references to "Master-Slave Cluster with vFabric Postgres 9.1 on vSphere" on vFabric blog
- Added backup/restore descriptions for db and full VM options
- Added reference to "Horizon Workspace Backup Data Best Practices" to comply with those requirements
- Removed "Order of installation" section
- Added JDBC example and screenshot of the configurator webUI where its setup

Version 0.4

- Added recommendation on when to perform the backup to avoid clashing with connector-VA sync operations

- Added instructions on how to migrate from internal service-VA Postgres to external vFabric Postgres

Version 0.5

- Updated section names and changed wording

Version 0.6

- Changed the location of the postgresql.conf file to the valid path (/db/data/postgresql.conf)

Version 0.7

- Removed db-GUI references on HTTPS 8443
- Changed psql script

Version 0.8

- Added instructions on Workspace 1.5 firewall changes to allow remote access to service-va postgres service (Fix by: Frank Pascher, VMware)
- Changed typos/wording

Version 0.9

- Added instructions on vFabric Postgres failover procedure and initial JDBC setup.
- Changed *search_path* option for DB user.