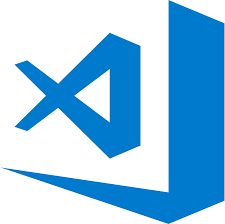
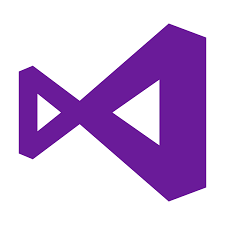
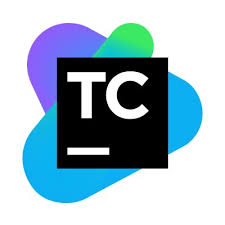
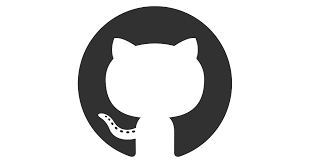
**Compass Deployments**



**Visual Studio**

.Net/C# Dev

**VS Code**

JS/HTML Dev

**GitHub**

Source Code Repo

**TeamCity**

Build/Test Automation

**Compass.ZIP**

UI/API artifacts

**J Frog**

Artifact Repo

**CA RA**

Deployment Automation

**IIS 8.0**

App Servers

**SonarQube**

Code Quality Scan



**Deployment**

**Test Triggers**

**1. Dev:** Developers check-in code into GitHub. UI developers use VS code /WebStorm IDE for web development. API/.Net dev use Visual Studio IDE. Both the environments support sonar lint plugin. This plugin helps in identifying the code quality issues even before pushing the code to GitHub which reduces rework.

**2. GitHub:**

URL: <https://github.ual.com/Compass>

Contacts: David, Shams; Tools Team

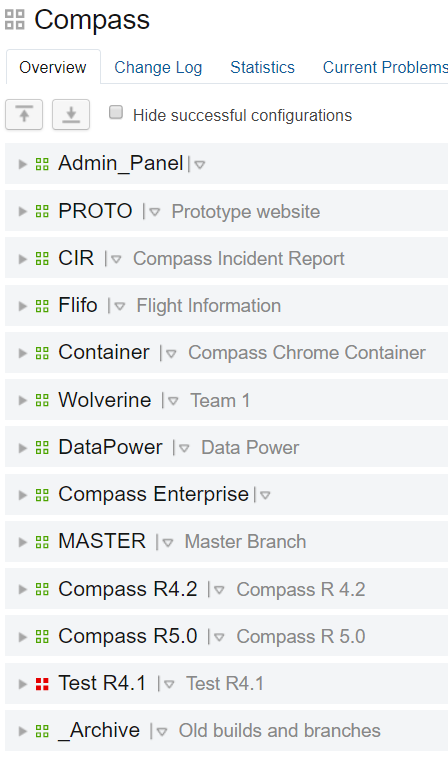
There are multiple repositories like Adminpanel, Compass, Flifo, Container, proto, style-guide in Compass GitHub organization. Developers can’t push their code to master branch directly. A new pull request is created to check-in the code to git and then an approver merges the code to master branch after pull request review. Master branch has the last dev code. A release branch is cut from master before the release and final package to deploy on prod is created from this release candidate.

**3. TeamCity:**

URL: <https://teamcity.ual.com/project.html?projectId=Customer_ContactCenter_Compass>

Contact: Vijay, Cherukuvada (Tools Team)

The above link points to the compass projects in TeamCity.

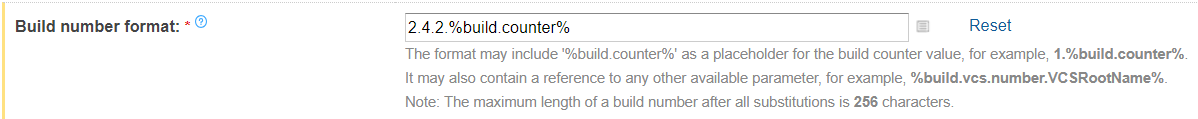


1. **TC Master Build:**

There are dedicated builds for UI and Services which pulls the specified branch in the parameters section.

These builds can be duplicated and reconfigured instead of creating one from scratch.

Build numbers are appended to the version numbers of the application packages. This version number and git commit number are then embedded into the package into about.json/about.txt file.



1. **Artifact paths:**

Generated artifacts on the agent machine are then packaged together into CompassMainUI.zip or CompassMainServices.zip. These directories may change from time to time.

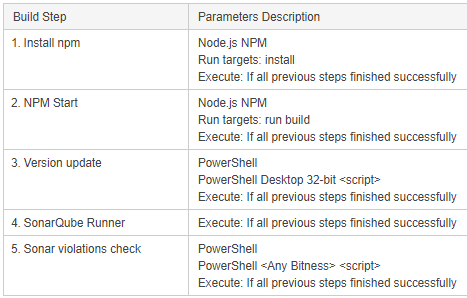
1. **VCS Roots:** Each build has a VCS root connection to GitHub. Each connection can point to multiple branches with the help of parameters instead of creating one for every build.

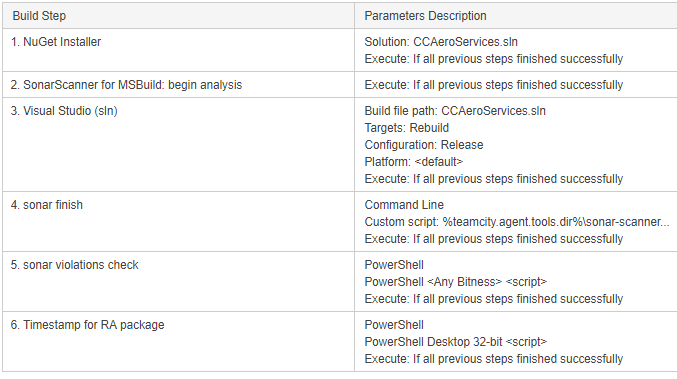
***Note:***  *Always use service account for any kind of authentication on TeamCity. Do not use personal file IDs. Service Account for compass is svc-Compass.*

1. **Build steps:**

Compass, AdminPanel, container, proto have different build steps for both UI and .Net. Important config settings xml file will be attached to this document for future references. Builds and deployments are chained together.

A new build is added to the queue spontaneously when there is a git commit. Depending on the agent availability the build fires off. At this point, artifact will be generated and stored on the TC server only. It is not deployed to artifactory as we don’t want to store all the build generated artifacts. Only the dev deployed artifacts will be saved. Core build steps maybe be defined by dev teams but sonar, version update scripts, custom mail and stuff like that must be defined in proper order. If the steps included are independent of build/deployment, they can be chained in another build with snapshot/artifact dependencies. This will reduce the runtime of the build.







1. **Deployment Steps:** Chron job is set to trigger dev deployment every hour with the latest build package, if any. Continuous deployments to dev can be achieved but as we deploy to only one server in lower environments at a time, the environments will be down for 2-3 mins per deployment which may cause too much downtime if there are more commits. This deployment steps include
2. Pushing the build generated package to JFrog Artifactory repo
3. Generating timestamp to define the Deployment Plan in RA
4. Trigger the RA deployment based on the type of deployment, environment to deploy, deployment template, etc.

**Tip:** .Net builds are comparatively faster than UI.

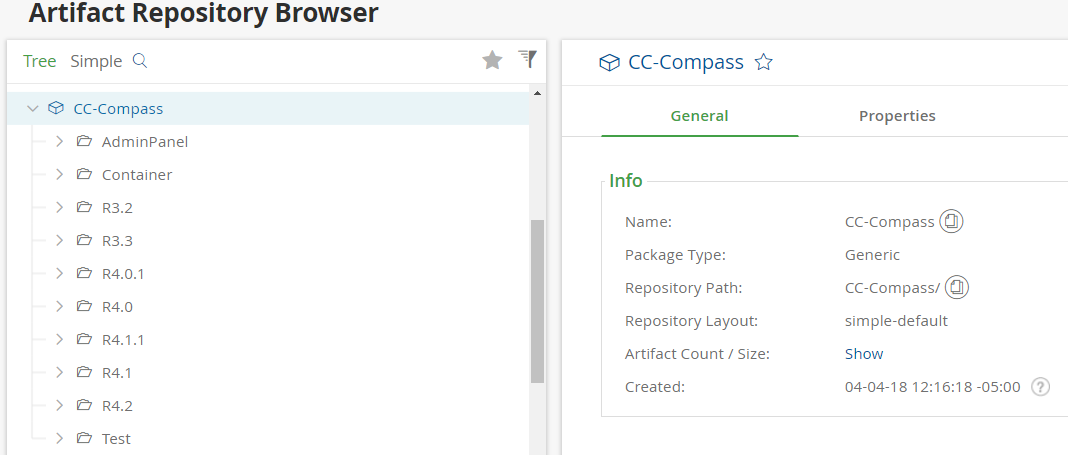
**JFrog Artifactory:**

Link: <https://artifactory.ual.com/artifactory/webapp/#/artifacts/browse/tree/General/CC-Compass>

Contact: Vijay, Cherukuvada (Enterprise Tools Team)

Compass Uses Artifactory as the package repository. As mentioned, not all build packages are not saved in Jfrog. Only the artifacts which are built/deployed to RA are stored here. Couple of notable points here are:

1. The name of the repository is **CC-Compass**
2. Only the compass service account has write access to this repository.
3. As of now, there is no retention policy applied to this repo.
4. Compass UI, Services, Container, Adminpanel, Flifo, etc. packages are saved here.



**CA Release Automation (CA RA):**

Link: <http://10.162.2.162>

Contact: Srinivas, Pattela (CARA Team)

Release automation is deployment automation tool used by multiple projects across United Airlines. The mechanism of this tools is similar to Octopus Deploy tool. It has actions and flows to define the deployment template. It has all the plugins to support compass IIS deployments. Please find the attached CARA Runbook, Custom Artifact Upload to CARA, Nolio agent setup docs below:

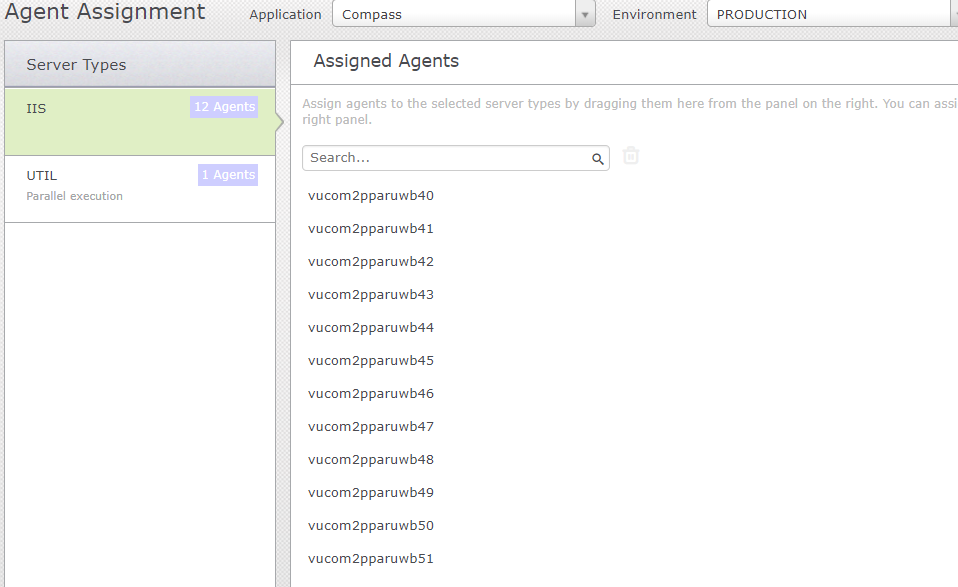




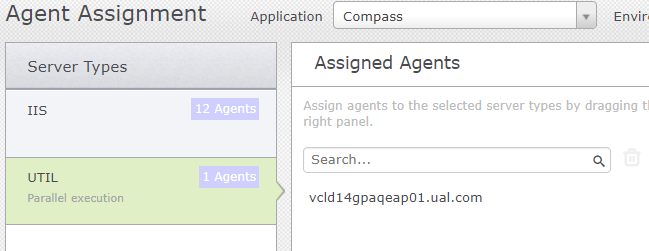


Steps to create a deployment environment:

1. **Create environment:** Environments > Environment and Tags >New > Name the environment > Select the process tags in the Process Tags pane (select all to avoid confusion, there’s no harm)
2. **Assign Agents to the environment created:** This agent is the server where we want to deploy. Steps: Environment > Select Environment created from the dropdown > Click on IIS on Server types pane > search for all the agents you want to deploy on the Agents pane and click Assign. Once you have assigned the IIS agents click UTIL to add the Utility agent which performs the execution of the deployment. UTIL agents can be any machines as these are not the actual deployment servers, but it’s better to assign one of the deployment server from the IIS server types assigned previously.

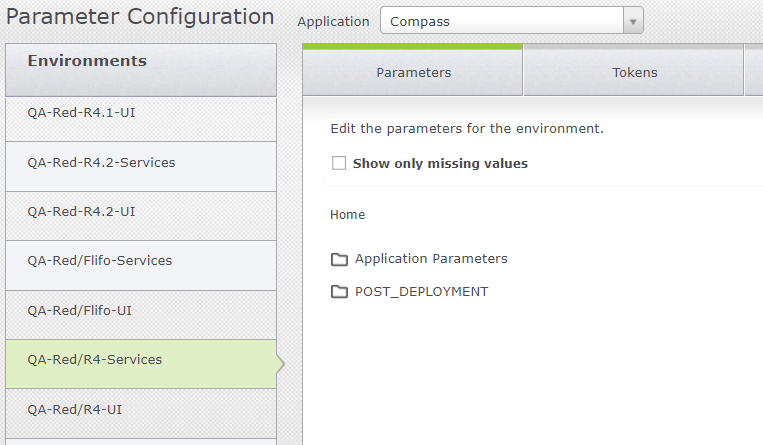


IIS Server Type: 12 agents (image)



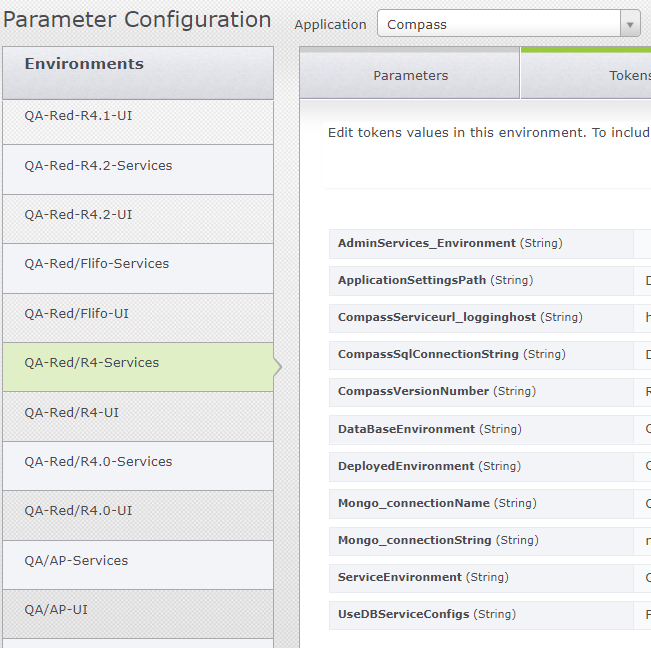
UTIL Server Type: 1 agent (image)

1. **Parameter and Configuration:** Environments > Parameter Configuration > Choose the Environment > Application Parameters/Post\_deployment/ F5 parameters > enter the parameter specific to the the deployment on the selected environments.



Parameter Configurations (image)

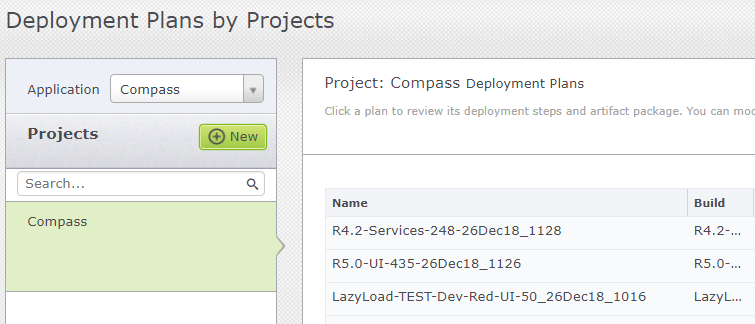
1. **Tokens:** Environments > Parameter Configuration > Choose the Environment > Tokens in the middle tab > enter the environment/ deployment specific parameter configuration which will change replace the place holder in the code during the deployment. This is very helpful feature; this feature can hide the account/password strings from the code.



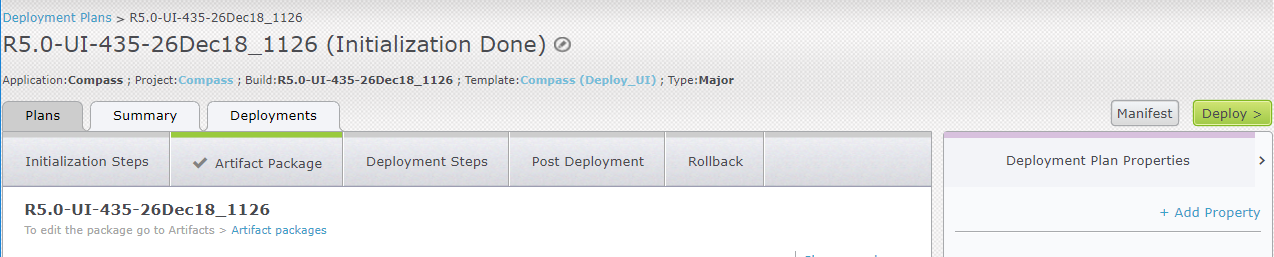
Tokenization in RA (image)

**Steps to perform a deployment:**

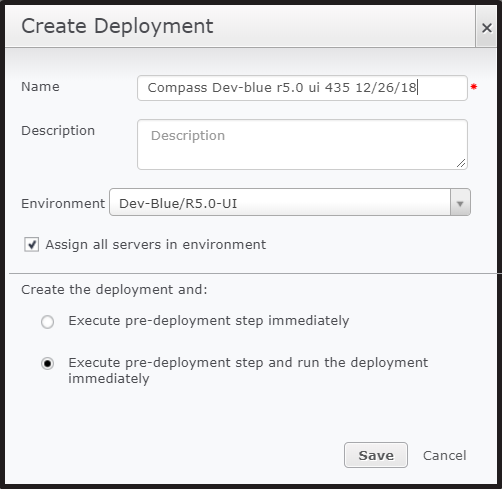
1. **Deployment from plans:** Click on Release > Deployment Plans by Projects > Select the deployment plane created by TC (make sure of the timestamp & build number before deployment) > Click on Deploy > in the Create Deployment pop-up box > Enter the name of the deployment > Select the environment to deploy > Click save and the deployment will begin.



Select Deployment Plan (image)



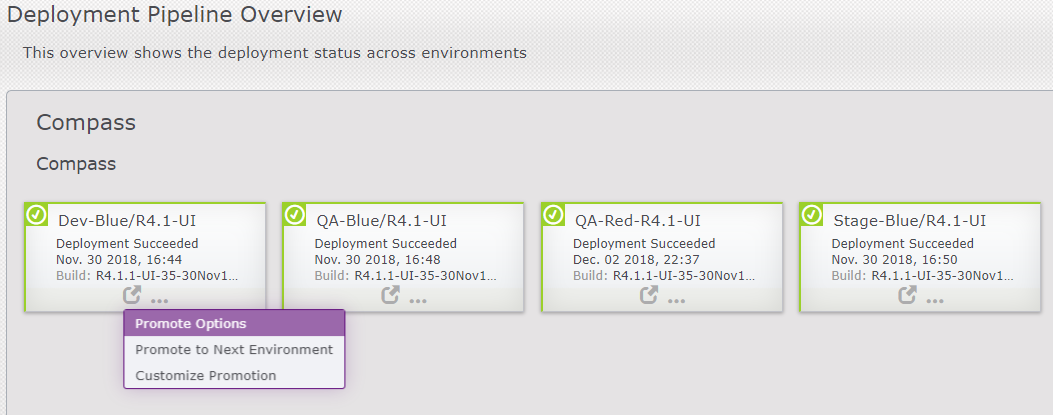
Trigger deployment (image)



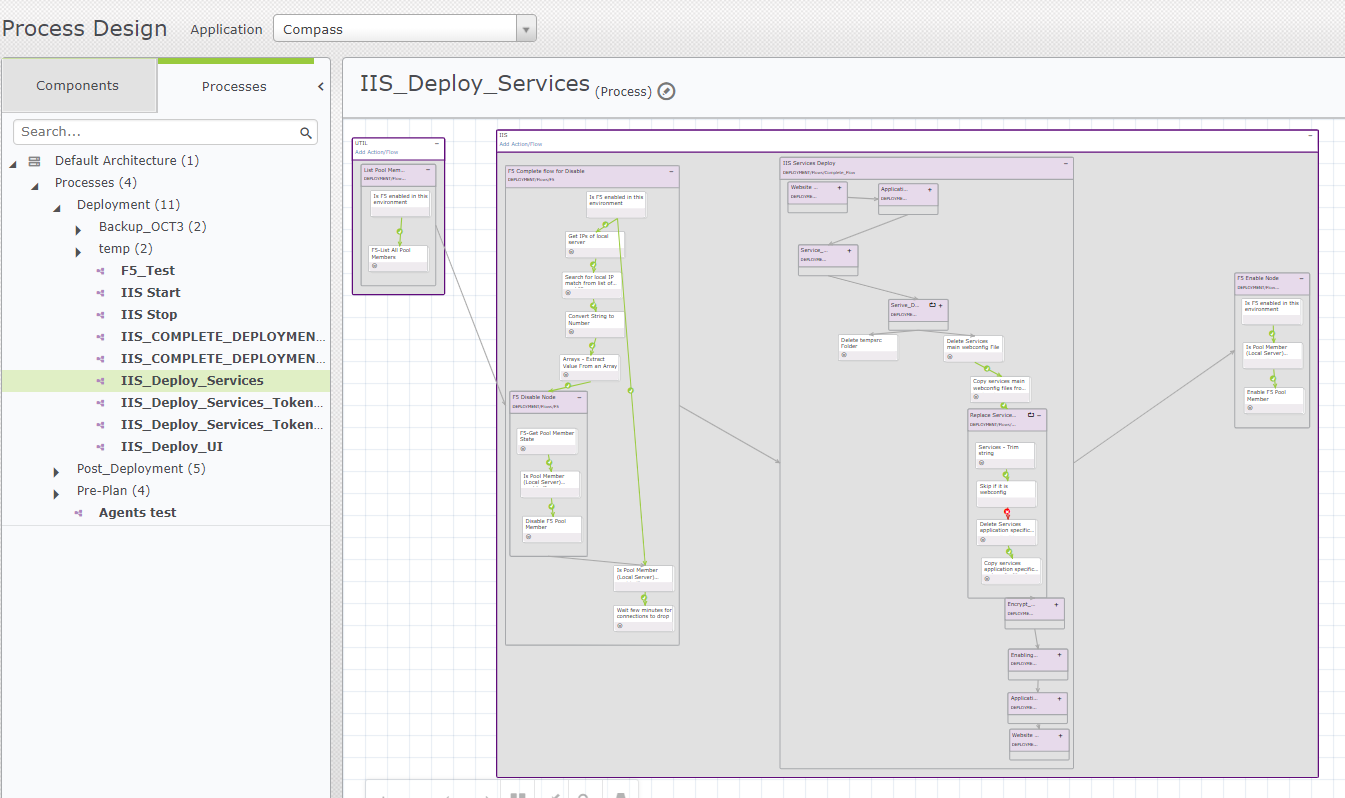
Run the deployment (image)

1. **Promote from Deployment Pipeline Overview:**  Here the same Deployed package to dev can be promoted to QA, stage, pre-prod without confusion. Click on Release > Deployment Pipeline Overview > Select the environment > click on option icon in the bottom of that environment > Promote to next environment/Customize promotion> Confirm by clicking Yes.

Promotion from environment to environment in the order of DEV > INT > QA > STAGE > PRE-PROD.



**CARA Process Design:** Actions and Flows are put together to create deployment process. There are multiple processes in compass application. Initially there were only a few but they were enhanced as the compass evolved. To view the Process/Components of the process: Click on Design > Process Design > Choose Components / Process Tabs from the right pane



Process Design (image)

**IIS Servers:**

Compass uses IIS 8 application servers on Windows server 2012 R2 virtual machines. Compass application is virtual directory compatible which means multiple applications can be hosted on a single site. But each application should have a dedicated application pool for better performance and security.

**SonarQube:**

Link: <http://sonar.ual.com:9000>

Contact: Enterprise Tools Team

1. SonarQube scans are triggered up on every checkin on Master /Release branches.
2. If the scan finds more than one issue, the Teamcity build is set to fail. This is already being setup but turned off.
3. In future No Pass-No Build can be turned on when there are no more Issue in Compass Sonar projects. Currently there are 400+ UI and tens of API code issue.
4. Implementing Code Coverage tools will be helpful.
5. **Citrix Deployments**



**IIS 8.0**

App Servers

Citrix VDIs managed by **VMware**

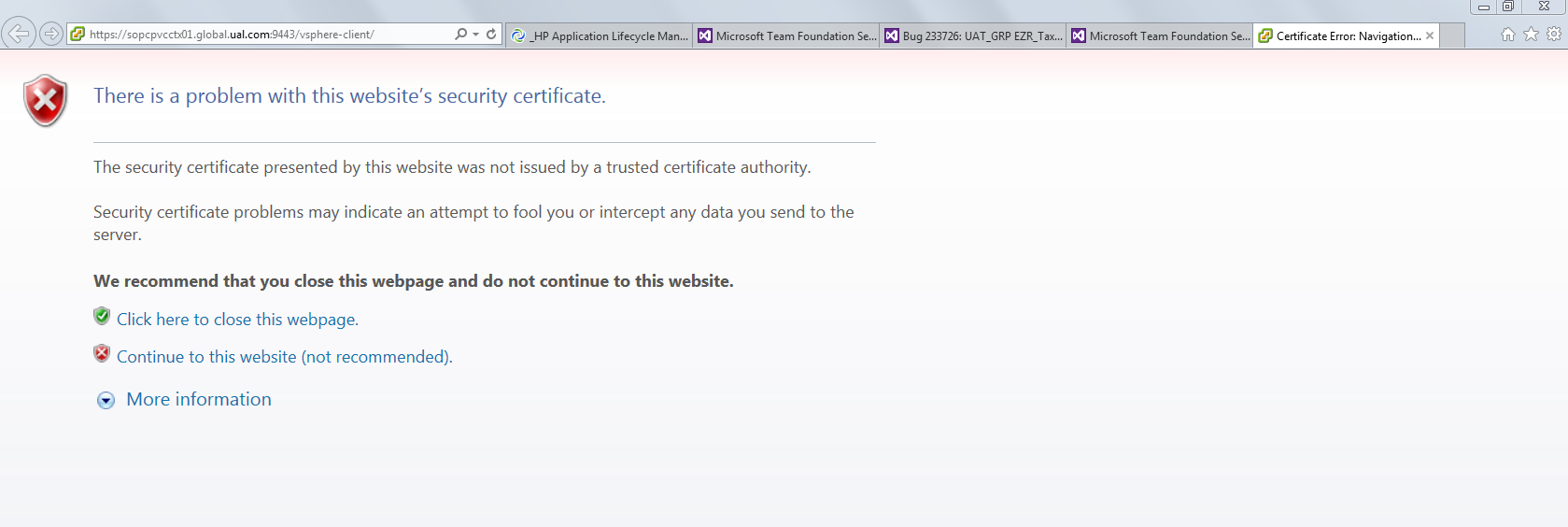
**Citrix Studio** to push Updates to VDIs

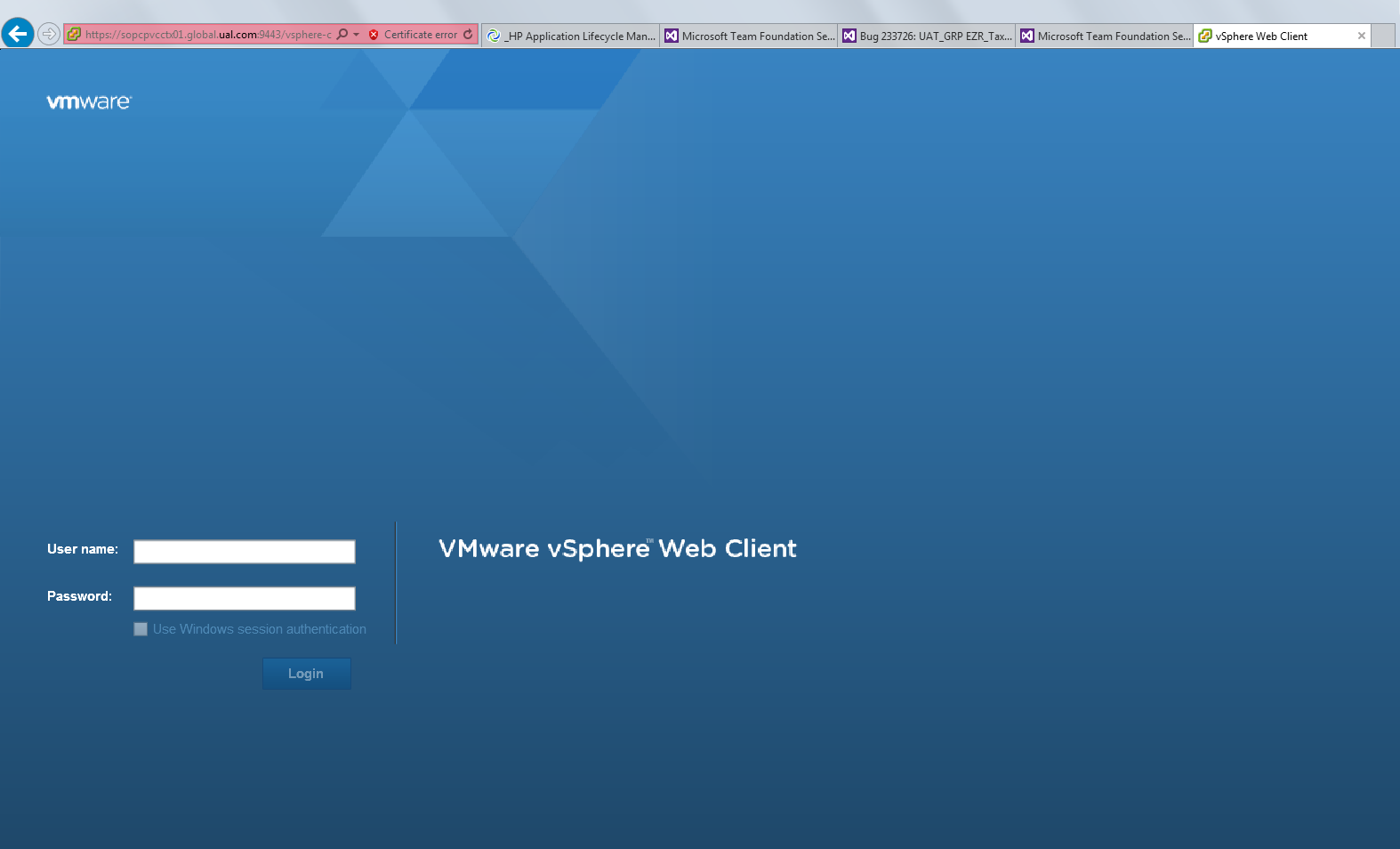
**Citrix Receiver**

(Agents/QA/PO)

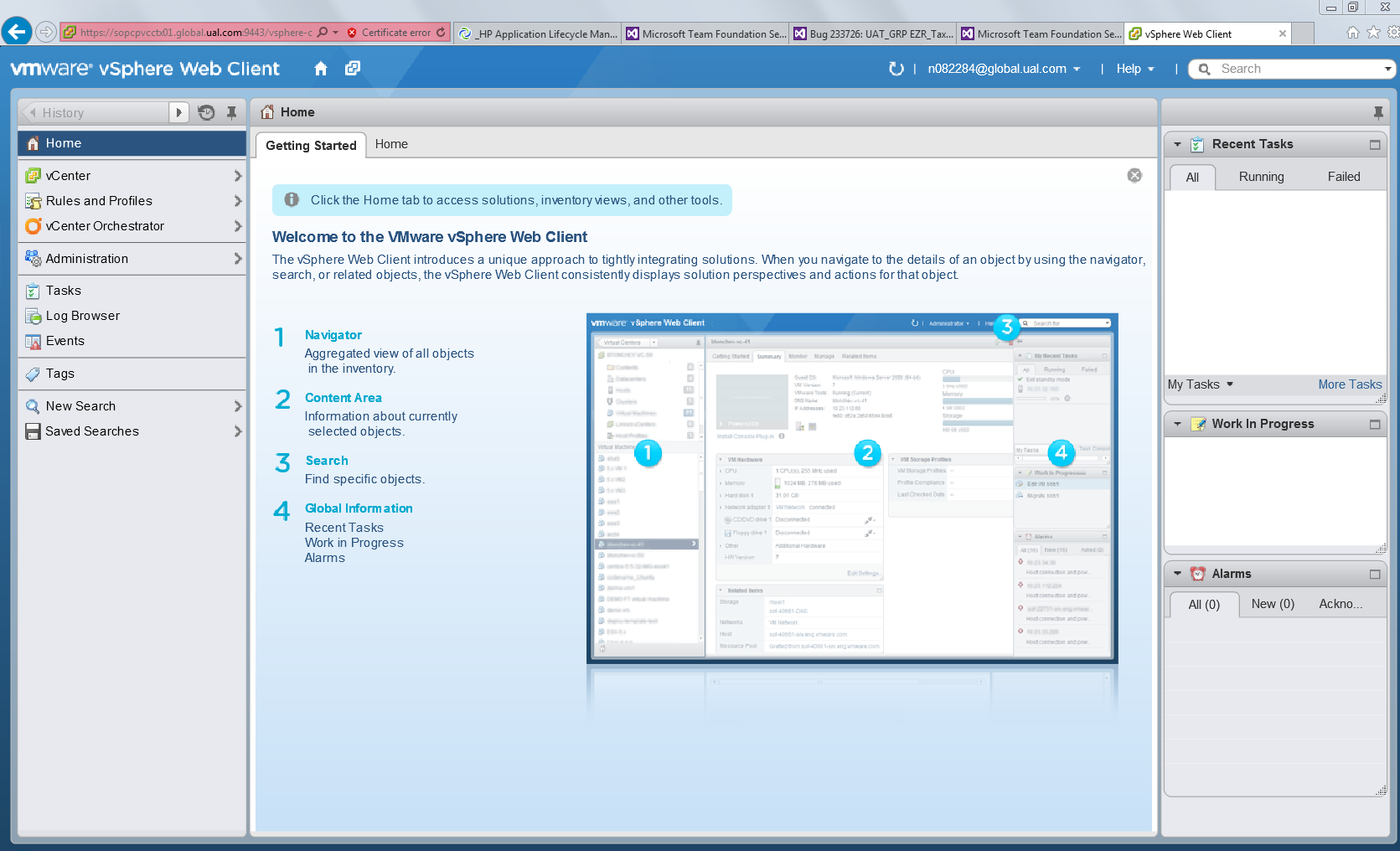
**Browser**

**(**DEV/QA)

1. Go to the path <https://sopcpvcctx01.global.ual.com:9443/vsphere-client/>. The below screen appears. Click on Continue to this website (not recommended)
2. The below screen appears. Enter Global\your file ID for Username and password and Login



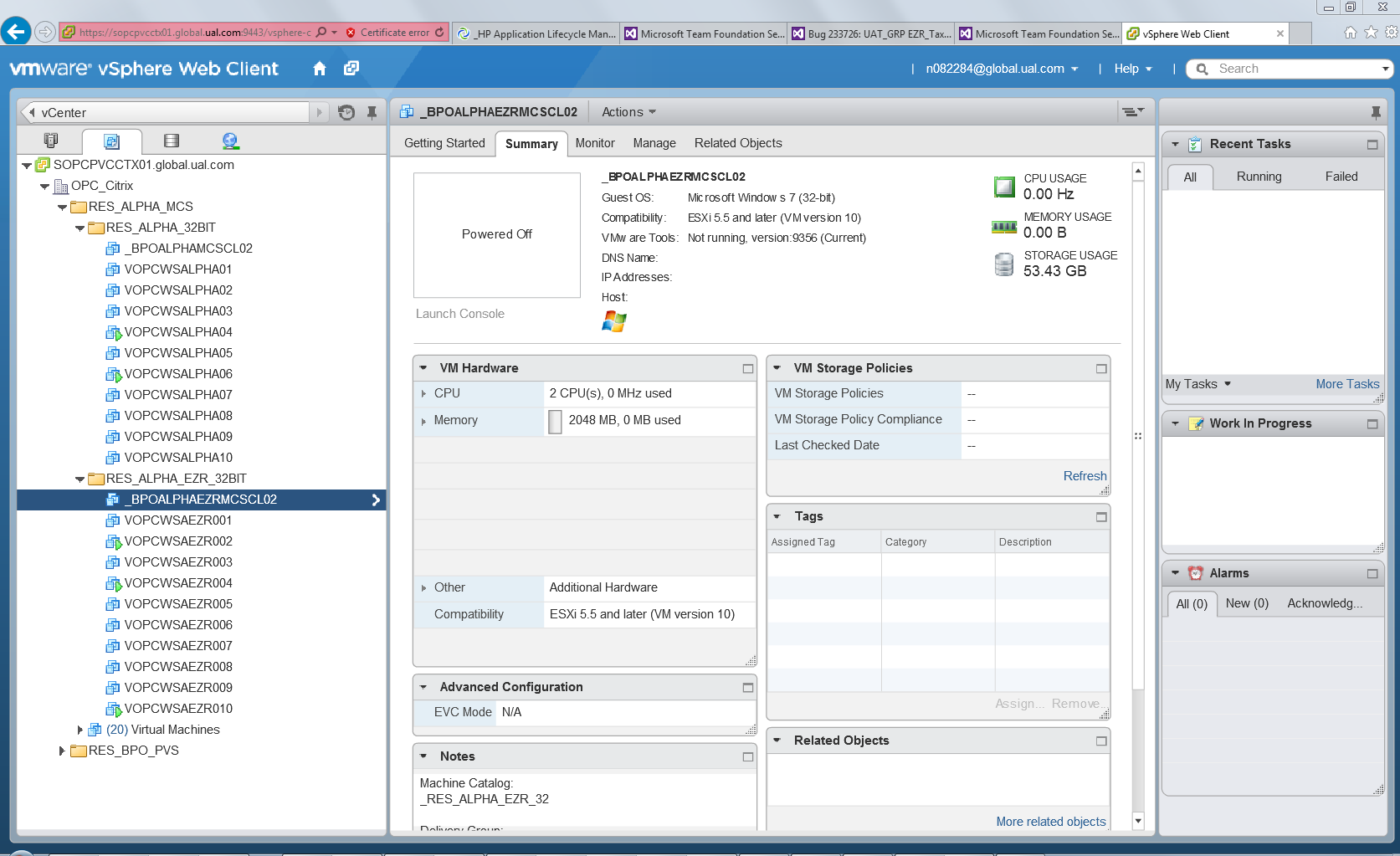
1. The below screen appears.

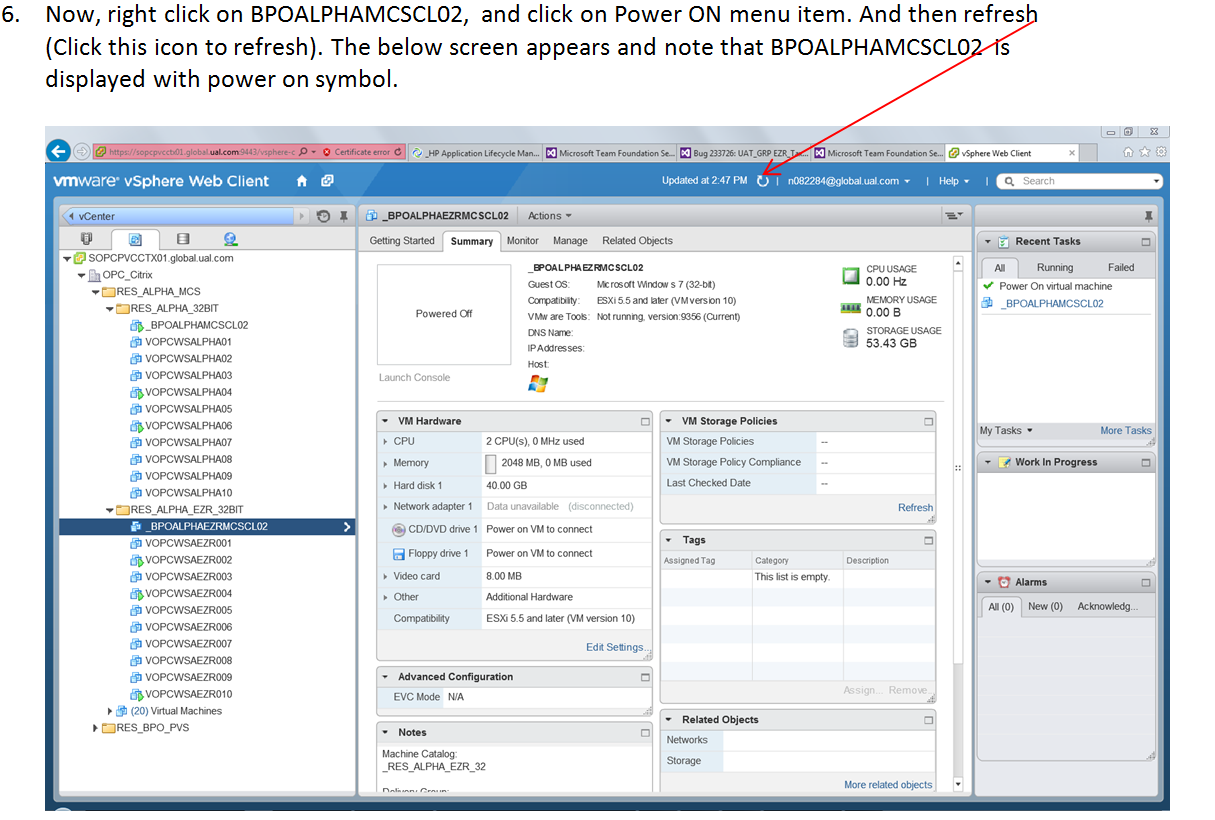


1. Click on vCenter on left side. The below screen appears

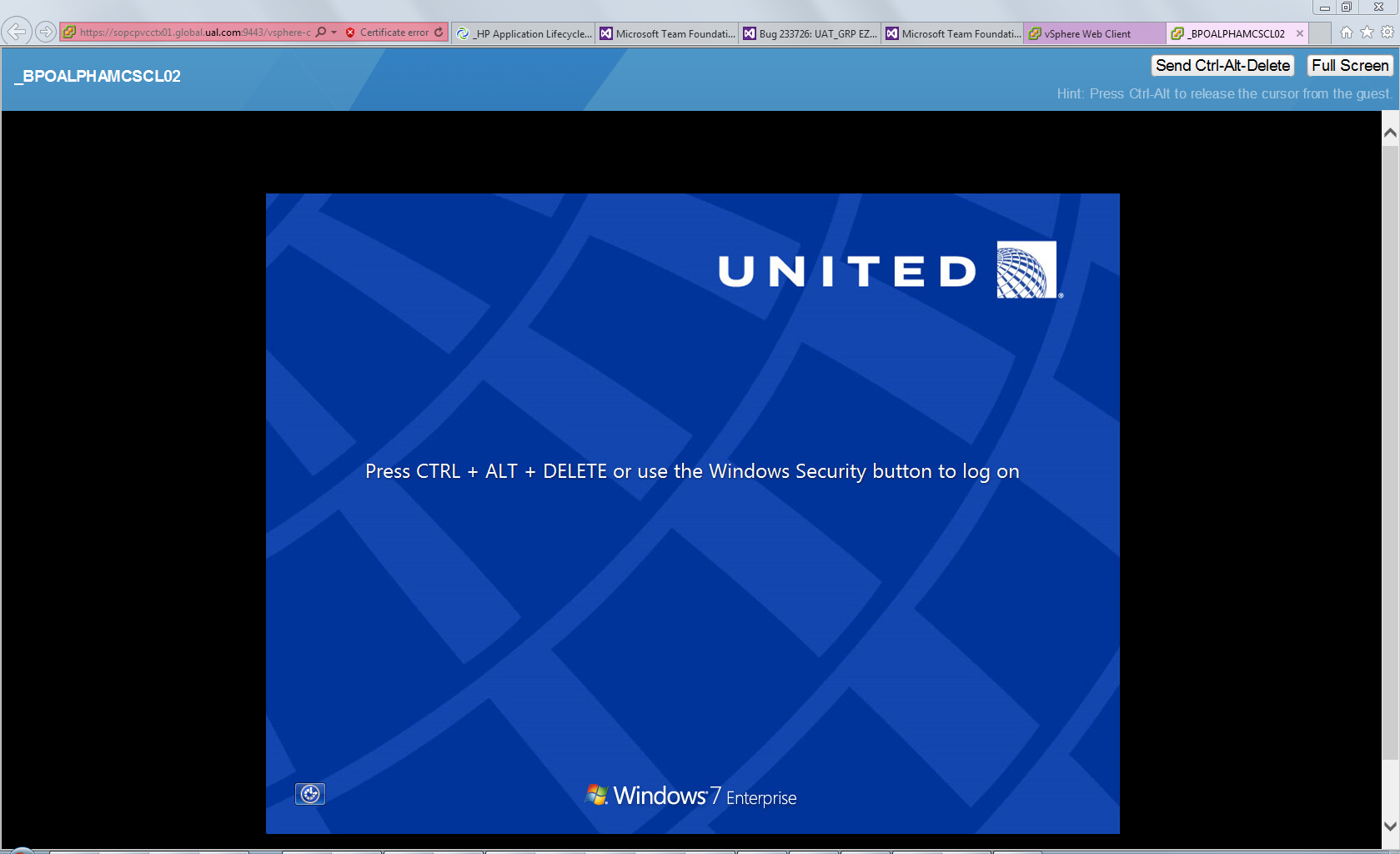


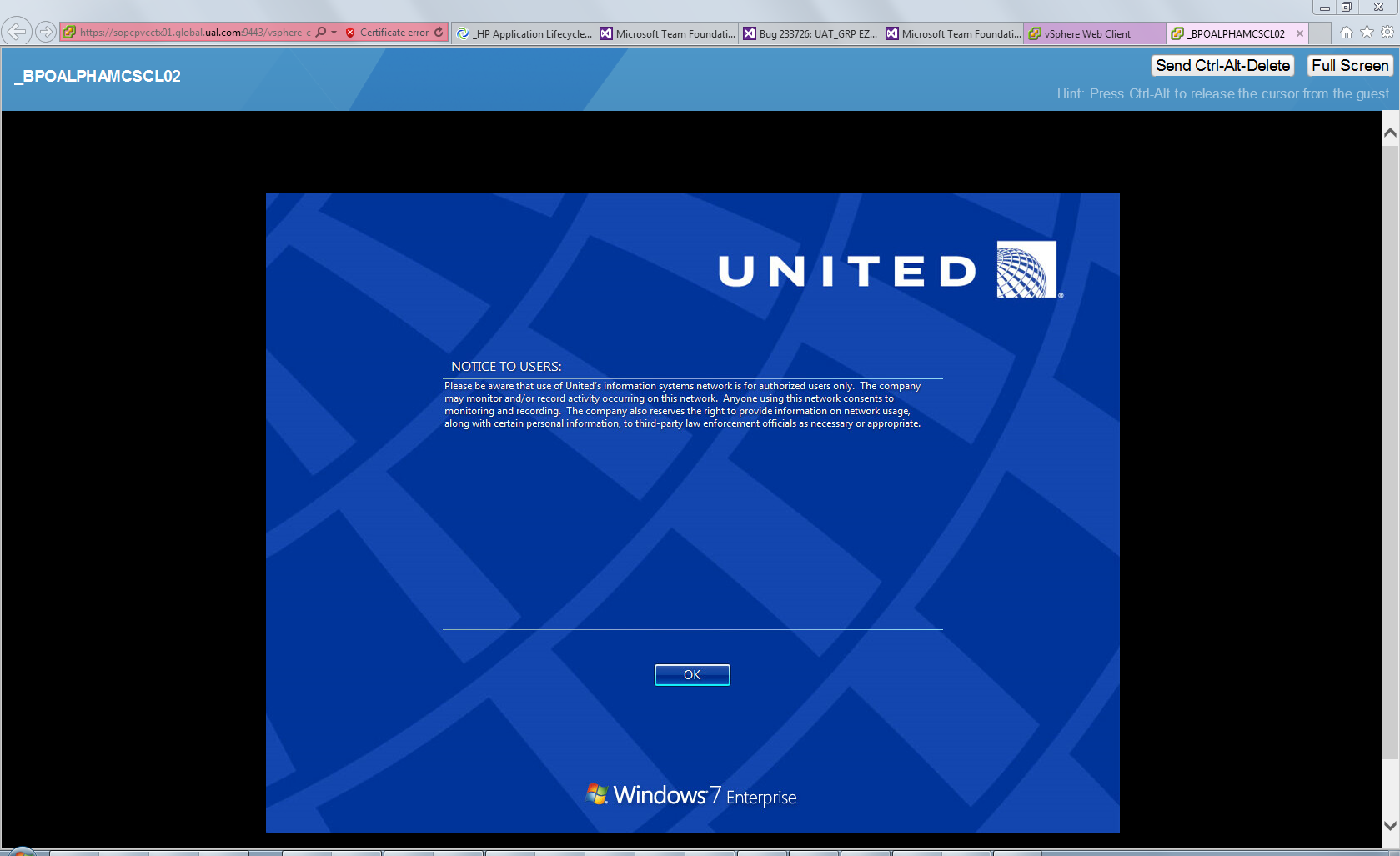
1. Click on VMs and Templates Menu on left side



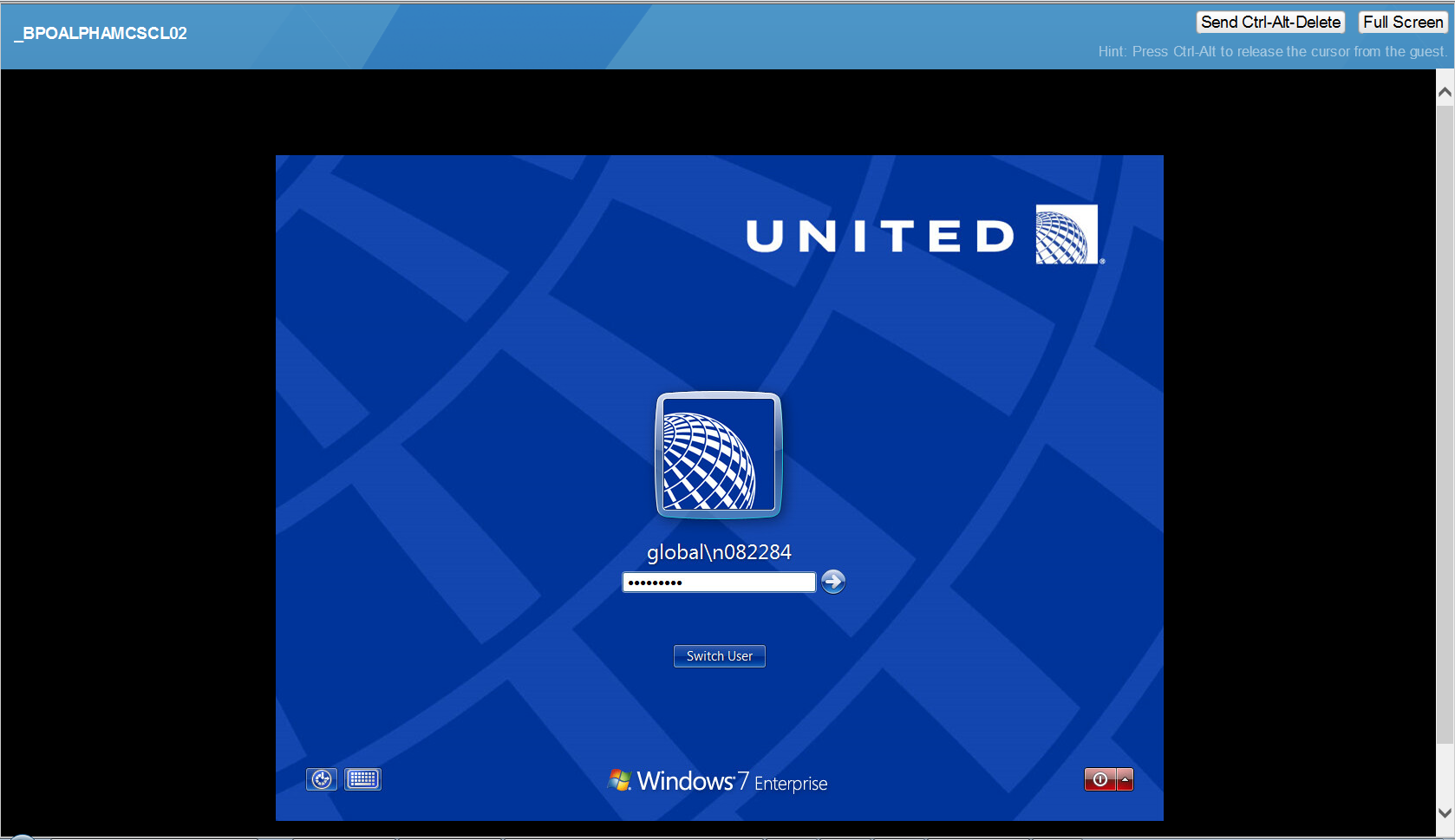


1. Now right click on BPOALPHAMCSCL02 and choose Open Console Menu item. Below screen appears. Click on Send Ctrl-Alt-Delete button.

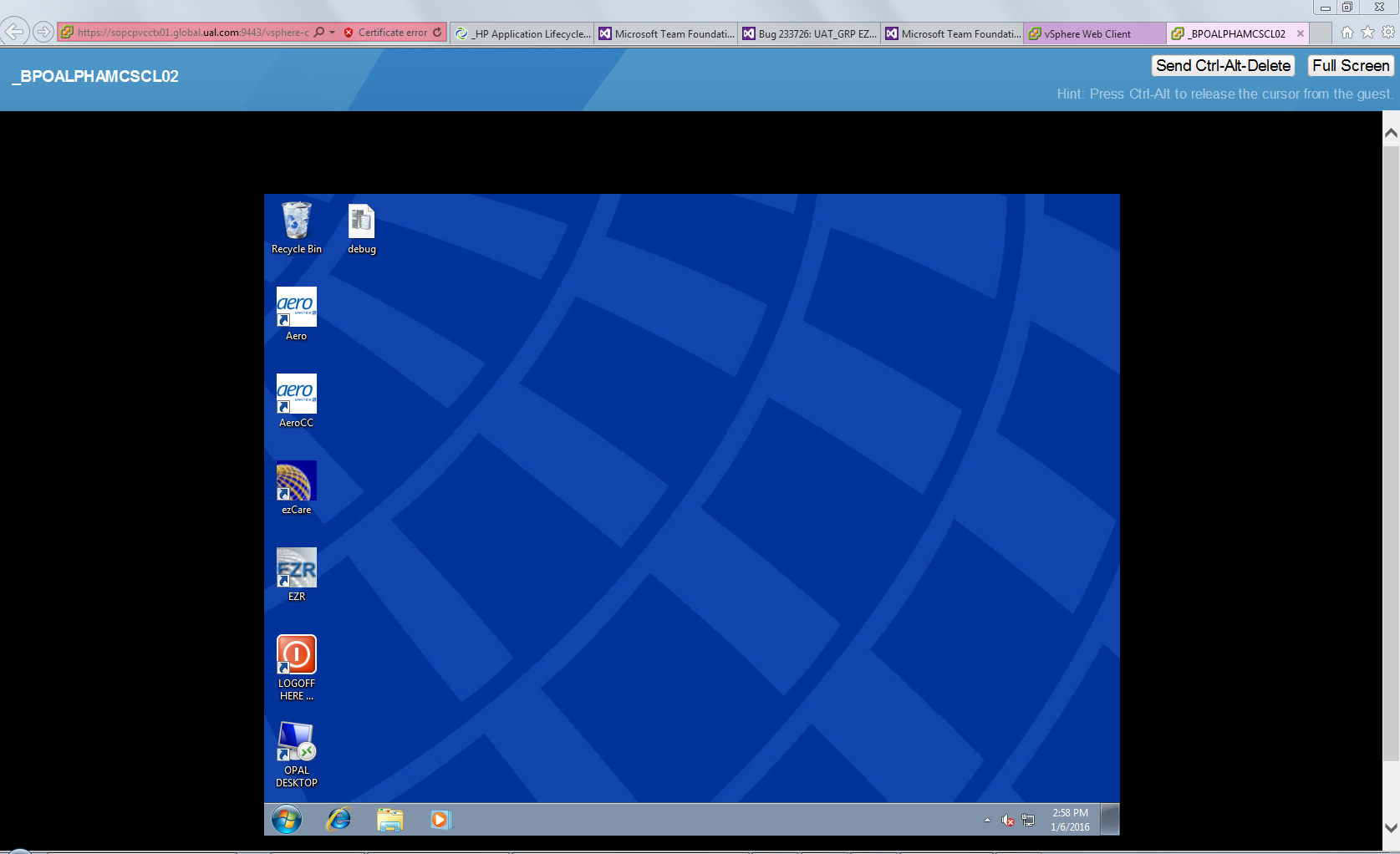


1. The below screen appears. Click OK. 

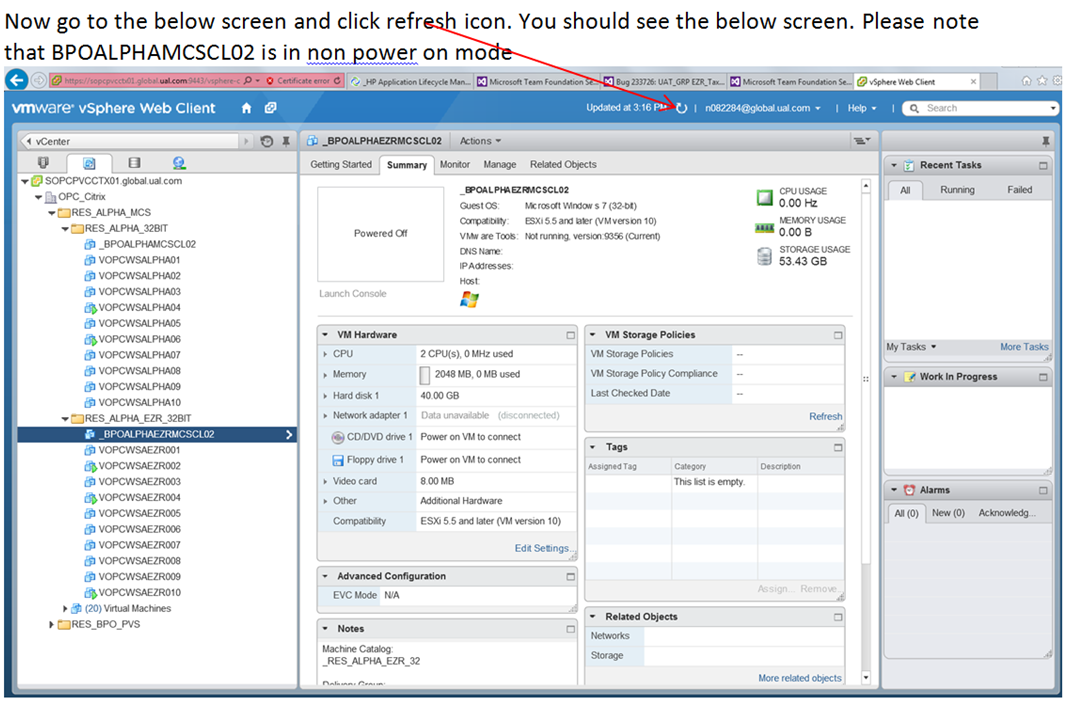
Below screen appears and will ask you to login. Login with your credentials.



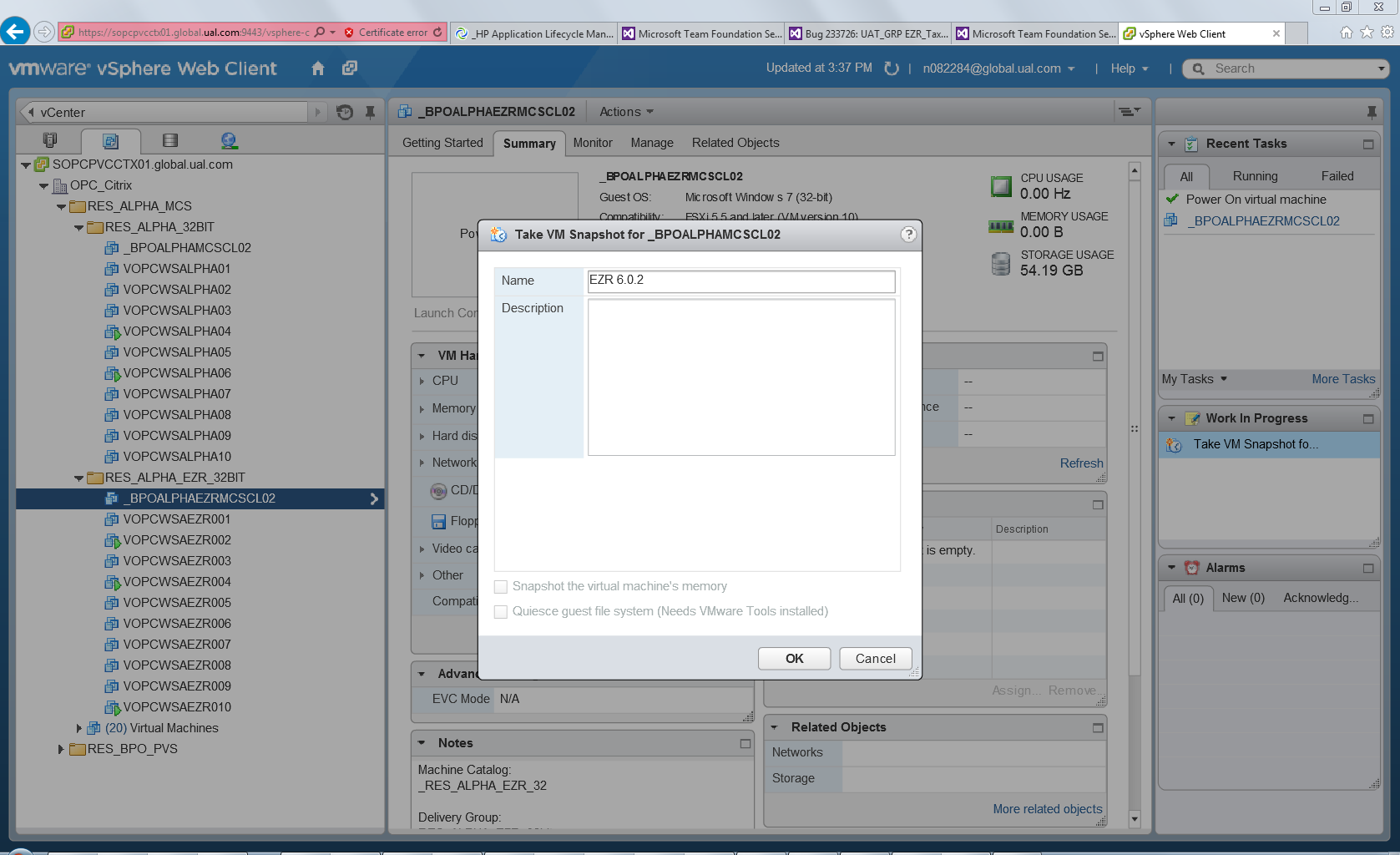
1. You will see the below screen



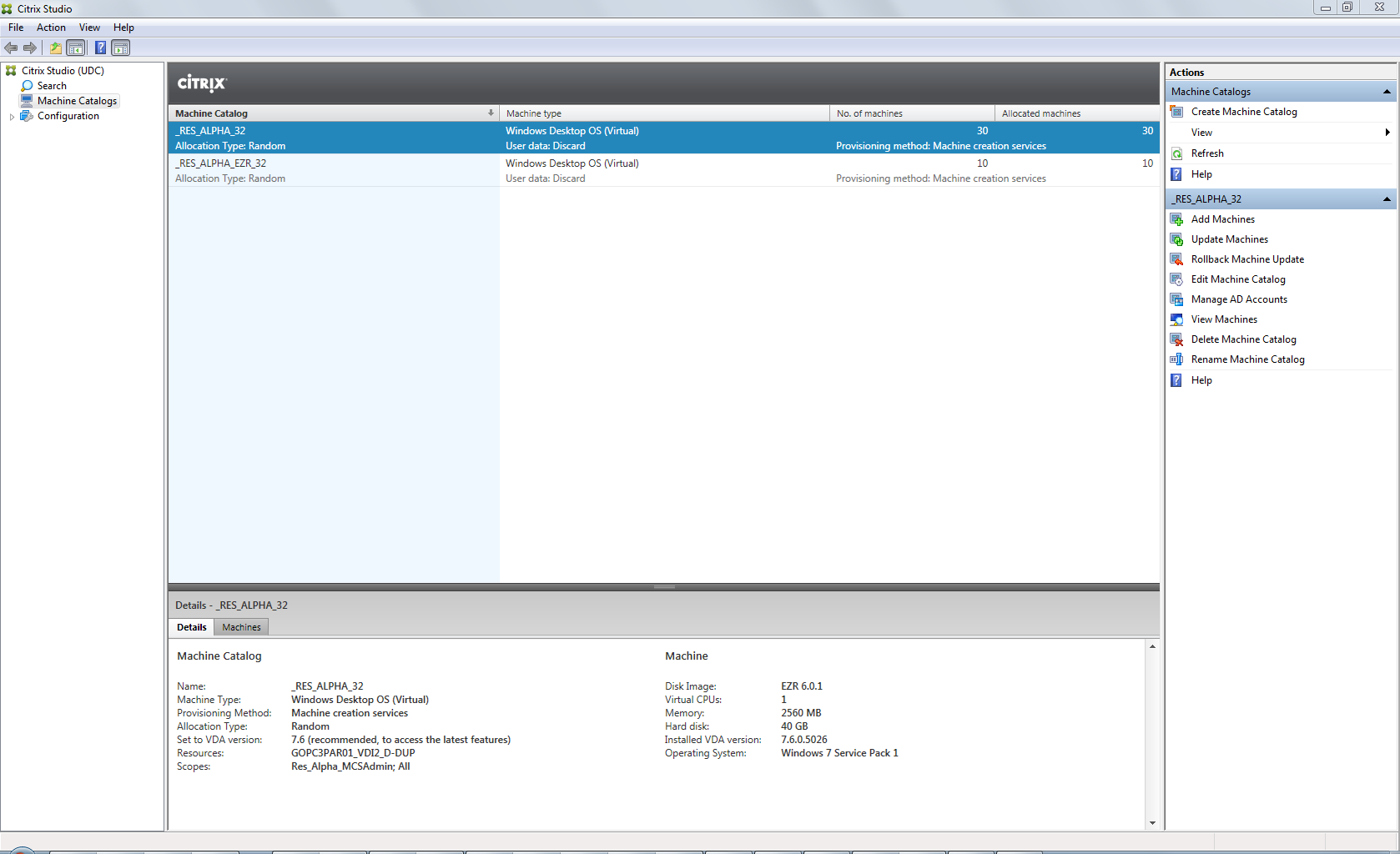
1. Now go to control panel, uninstall EZR. Open build machine path, copy the latest version of EZR folder and paste it to this machine’s C drive(to save space on this machine, please delete the previous version EZR folder in C drive if any), open the folder, install the setup.EXE (Run as admin). Open EZR once to make sure latest version has been installed.
2. Then Shut down the machine.



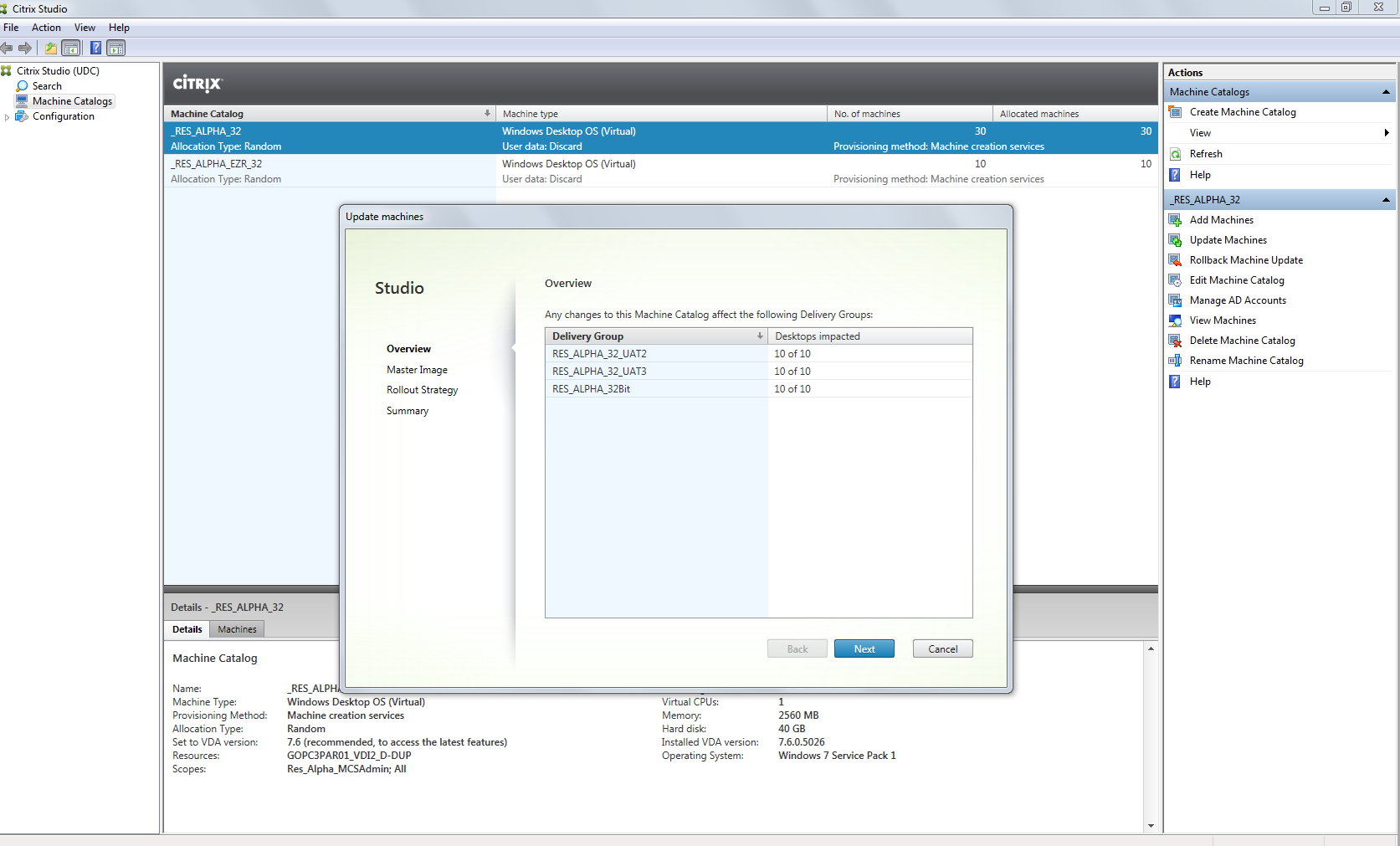
1. Now, go back to Step 6 and Repeat Steps 6-12 for BPOALPHAEZRMCSCL02 .
2. Now, click refresh icon(as shown in the above screen) on vCenter site. Now you will see both BPOALPHAMCSCL02 and BPOALPHAEZRMCSCL02 VDIs diplaying in non power on mode.
3. Right click on BPOALPHAMCSCL02 and choose Take Snapshot menu item. You will see below screen. Enter name for Name field(I usually enter latest version number). And click OK.



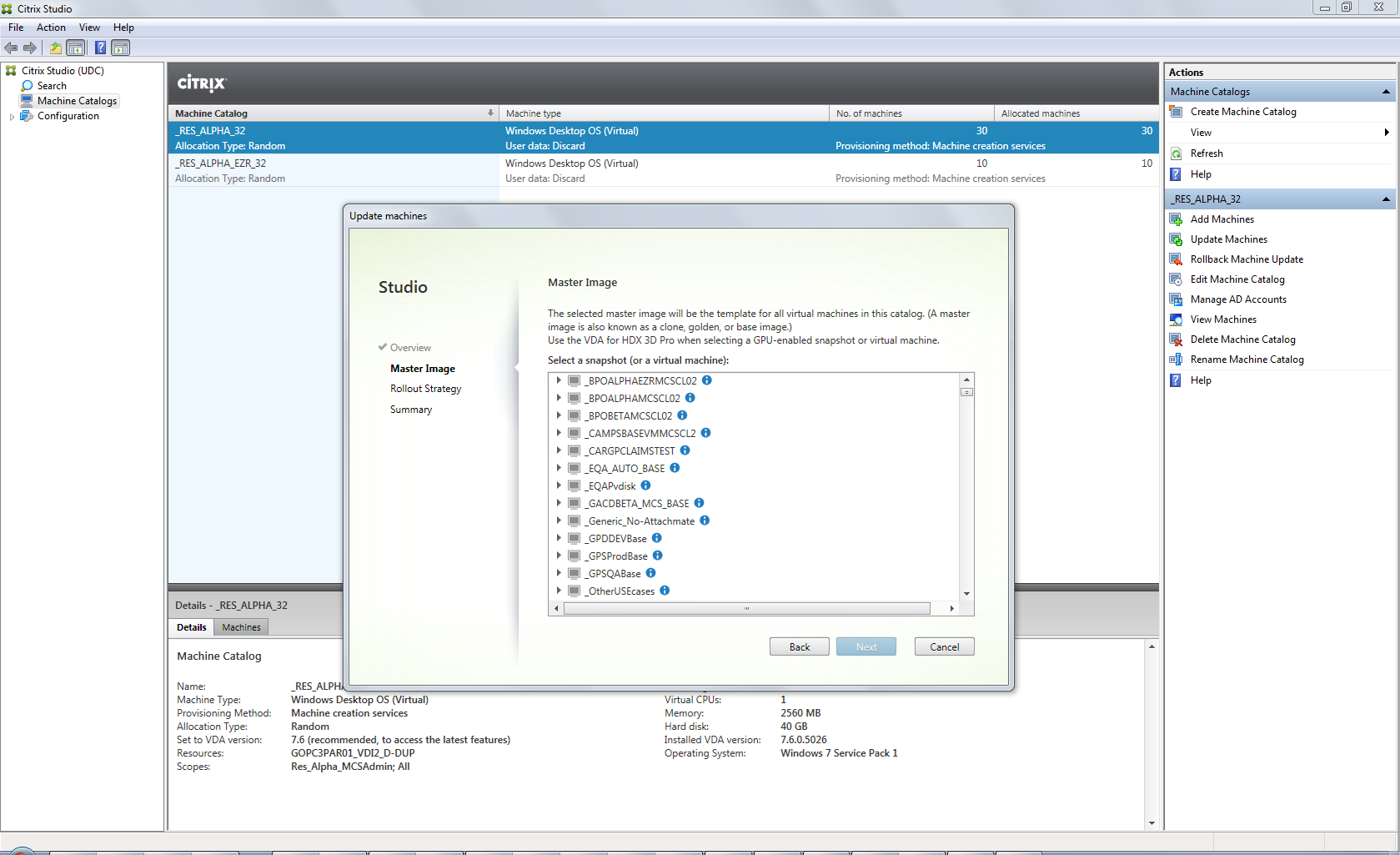
1. Repeat step 15 for BPOALPHAEZRMCSCL02
2. After this step you can log out from vCenter(vSphere) site.
3. Now, go to start menu of your computer and type Citrix Studio (I have asked Rajesh to install this software on your system) and open Citrix Studio. You will see the below screen.



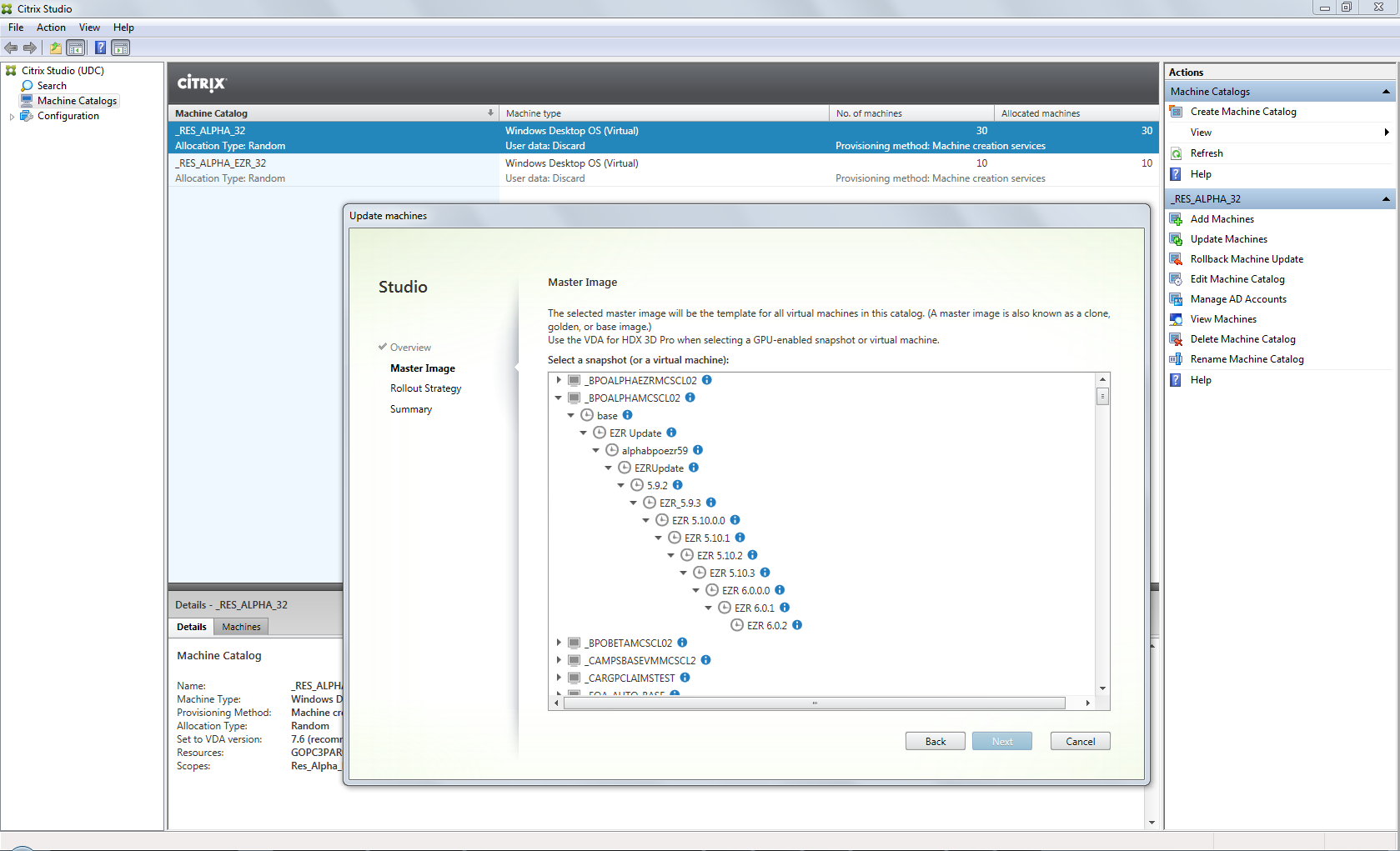
1. Right click on RES\_ALPHA\_32 and choose update machines menu item. You will see below screen. Click Next.



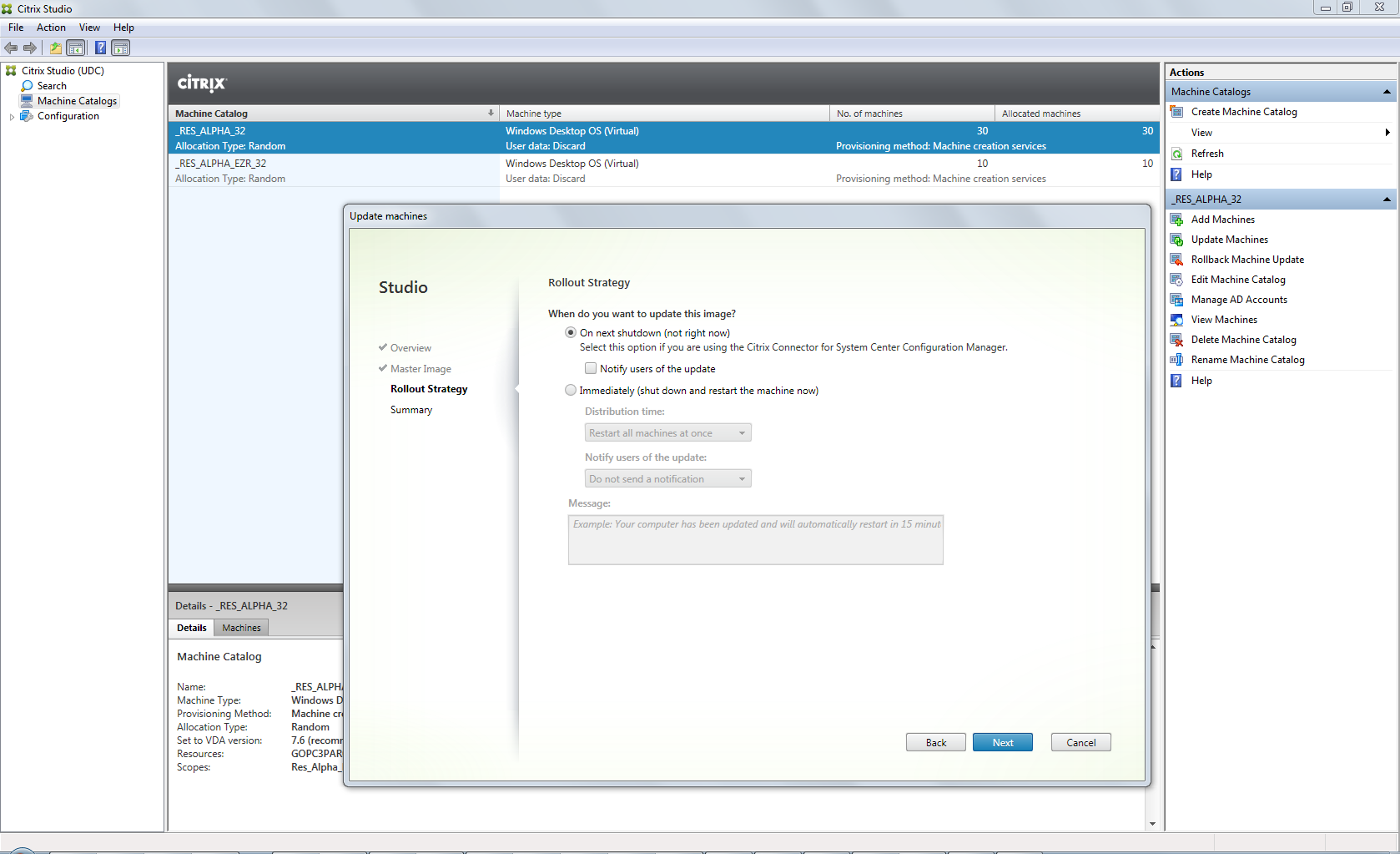
1. You will see below screen. Expand BPOALPHAMCSCL02 (since we are updating RES\_ALPHA\_32) till you find the name you gave while taking snap shot)

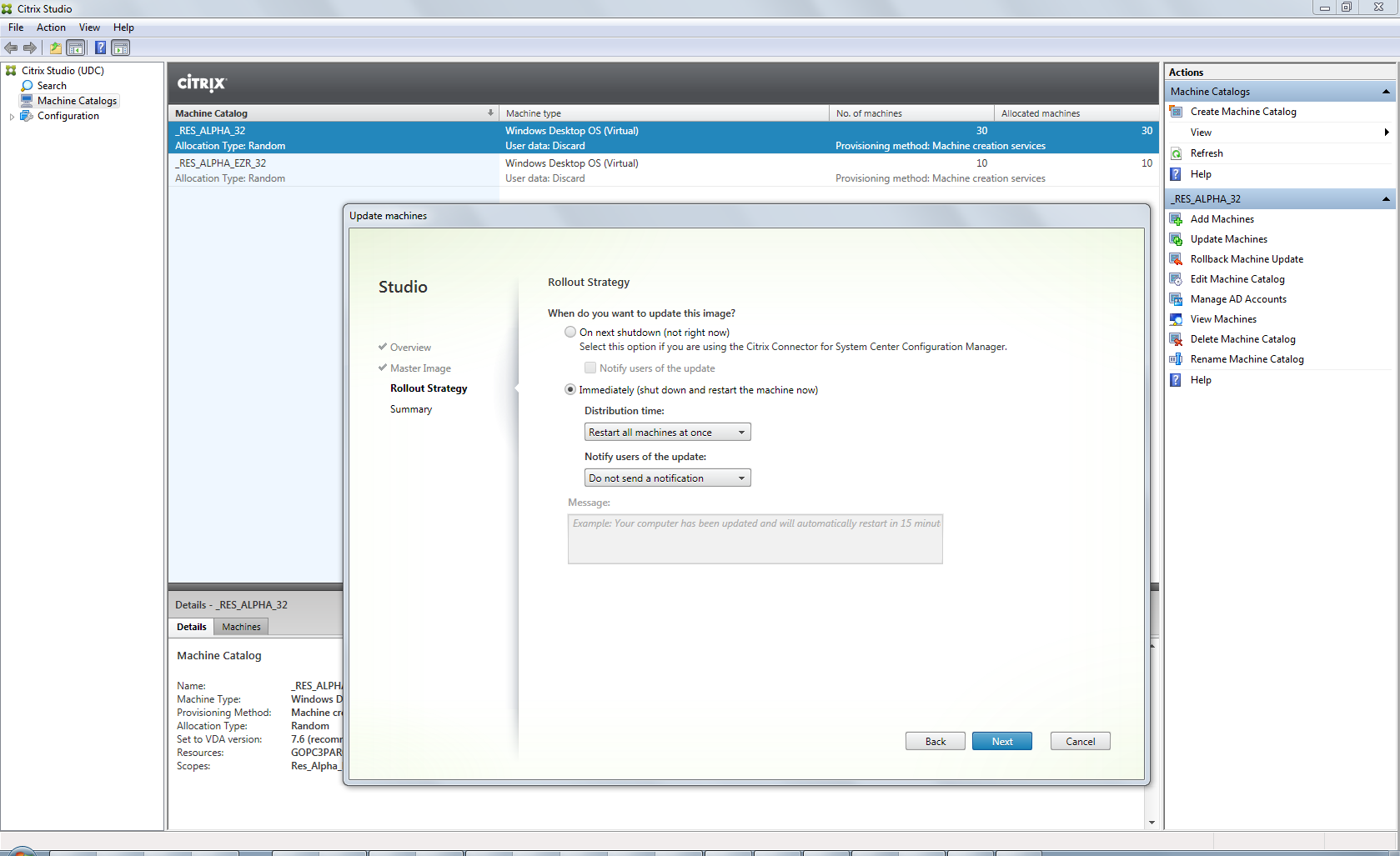


1. You should see a screen like below after expanding BPOALPHAMCSCL02. Click on EZR 6.0.2 (the name I gave while taking snap shot) and click Next.

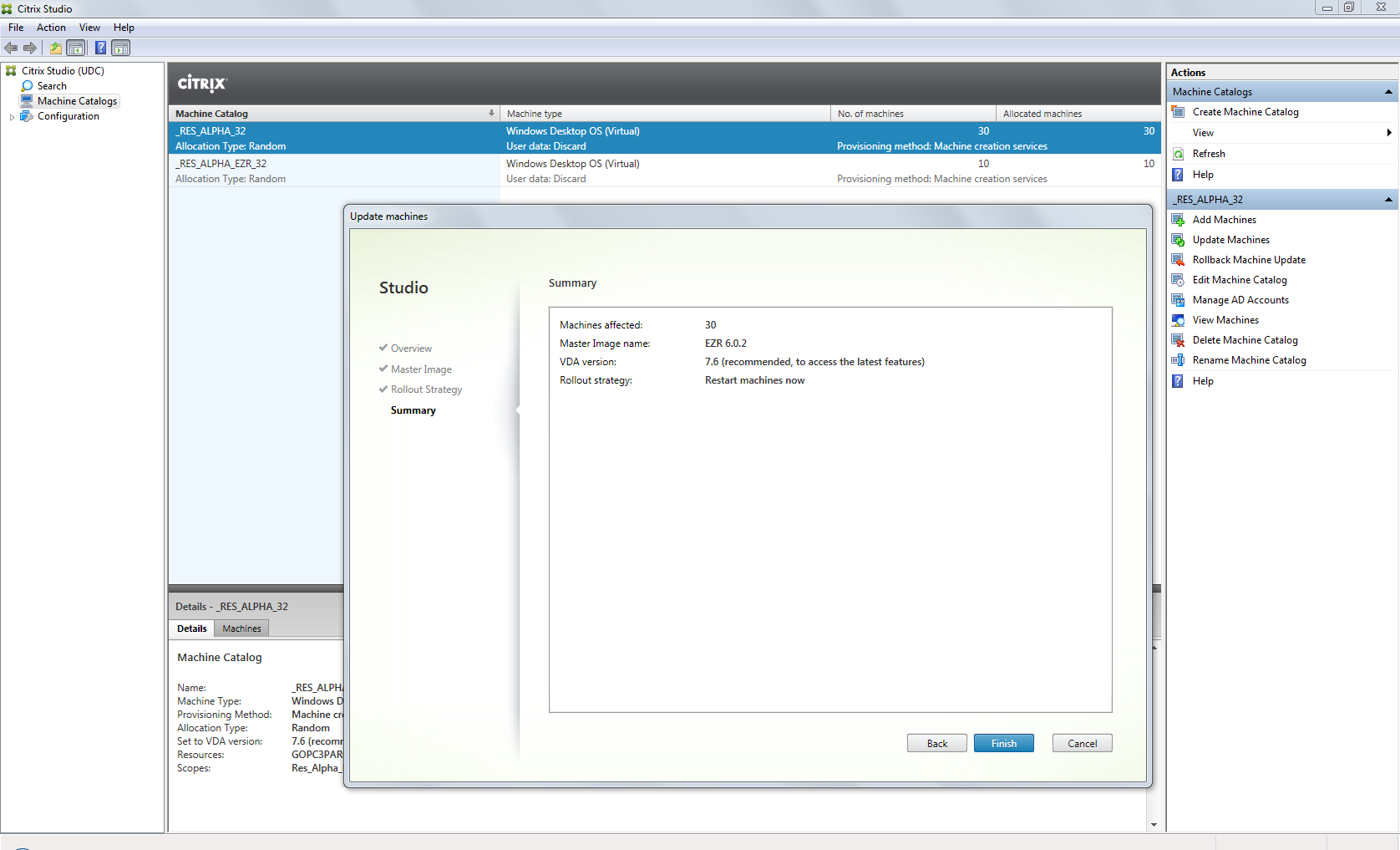


1. You should see the below screen now. Select immediately option and click Next.

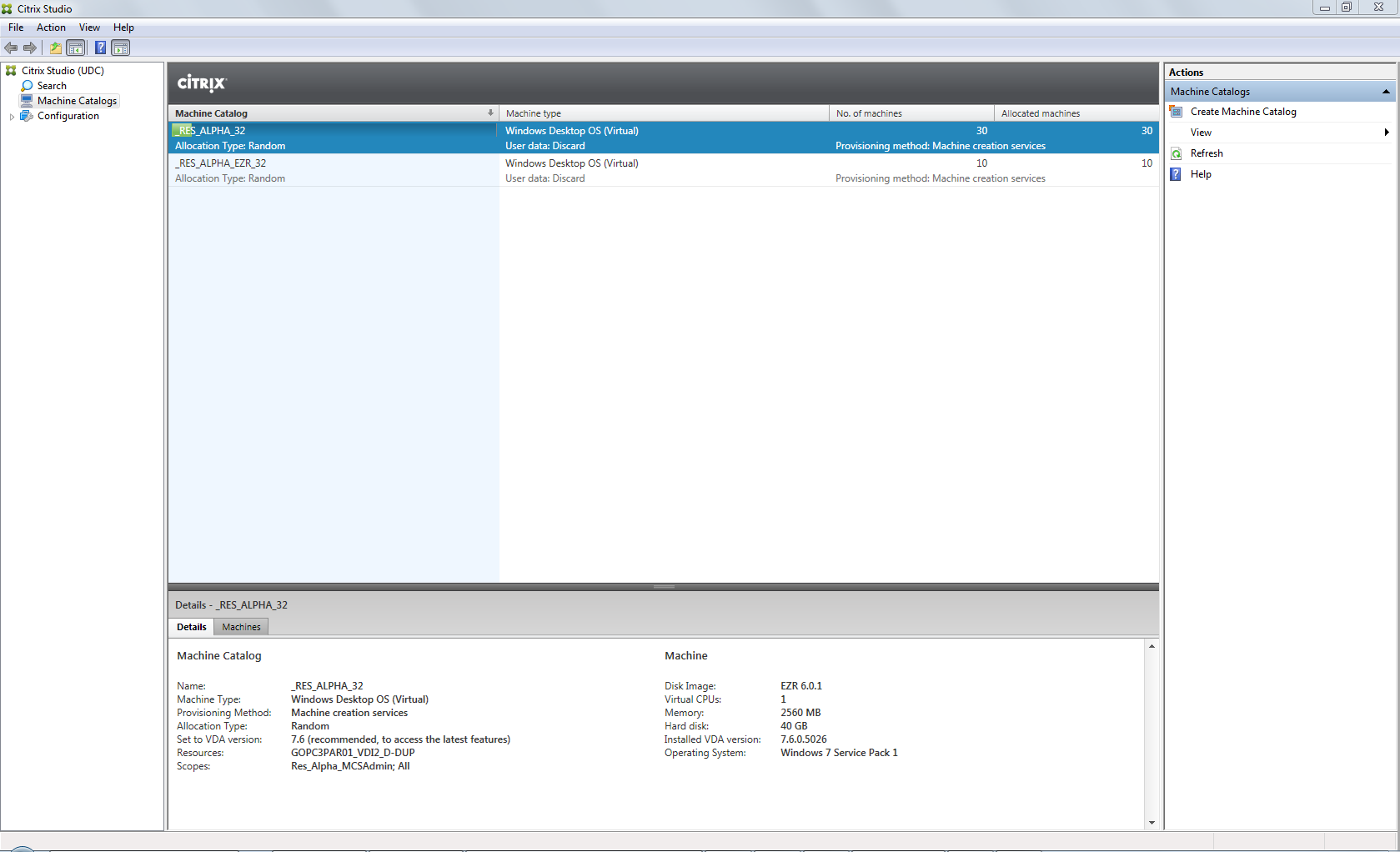




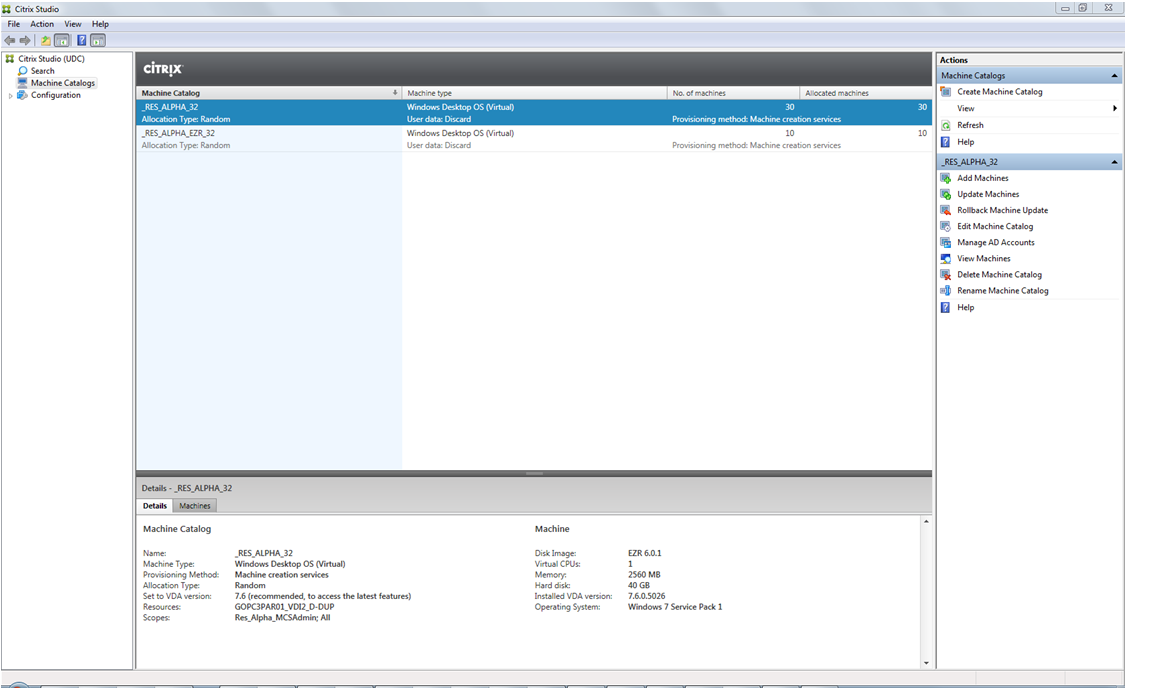
1. Click Finish



1. You will see a screen like below

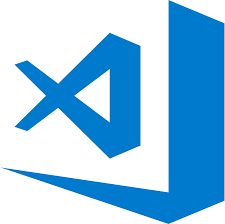
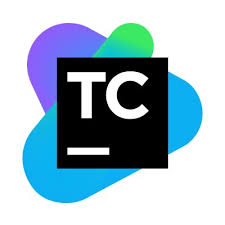
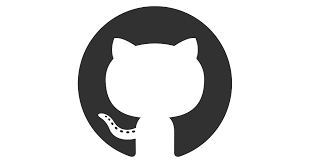


1. Now, repeat steps 19-24 for RES\_ALPHA\_EZR\_32( this time you need to expand BPOALPHAEZRMCSCL02 till you find your snap shot name in Step 20)
2. After some time (may be 40 t0 60 min) you will see a screen like below.



1. Now, go to https://remotectx.ual.com, login, click on Res\_alpha\_32 icon, login and open EZR and verify are you seeing latest EZR, log off. Do the same thing for RES\_ALPHA\_EZR\_32 also.
2. When you are waiting for the step 26 to complete, please RDP to VOPCPTS07.global.ual.com (this is for Alpha TS update) and uninstall EZR and install latest EZR by copying exe from build machine. And then log off.

**Proto/Style-Guide Deployments**



**VS Code**

JS/HTML Dev

**GitHub**

Source Code Repo

**TeamCity**

Build/Test Automation



**IIS 8.0**

App Servers

**Proto/StyleGuide:**

1. Porto/Style Guide CI/CD doesn’t include artifact storage or CARA deployment. Teamcity directly deploys the artifact to IIS server.
2. Samba Deploy is used for CD.
3. There are no backups.
4. Automated deployments on every check-in.
5. Configurations files for both Build and Deploy are attached below.



**Deployment Strategy**

**Dev**



**Integration**



**PRE-PROD**



**QA**



**STAGE**



**BETA**



**PRODUCTION**



QA/Stage CSL

Pre-prod/Prod-perf CSL

Beta/Prod CSL

**Application Pipeline**

