



Cloud render services for design decisions

Creating images and movies from 3D VRED scenes

Christoph Frandrup
Technical Consultant

Class summary

Photorealistic digital imagery of virtual prototypes helps to accelerate the design and engineering process and speeds project approval.

This class shows how to use cloud render services to accelerate the image and movie creation process.

Class summary

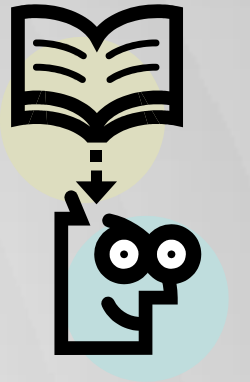
Real life example:

Render 27.000 frames on a single machine takes ~ 1200 days but using 320 nodes just takes about 4 days



Vehicle designed and built by Marc Mainville

Key learning objectives



At the end of this class, you will know:

- How to use and setup a VRED scene for large render Jobs
- How to setup a cloud/cluster environment to use with VRED
- Best practices on compute time, price and performance
- How a Major German Automotive OEM uses VRED for design decisions

Setup a scene for large render Jobs

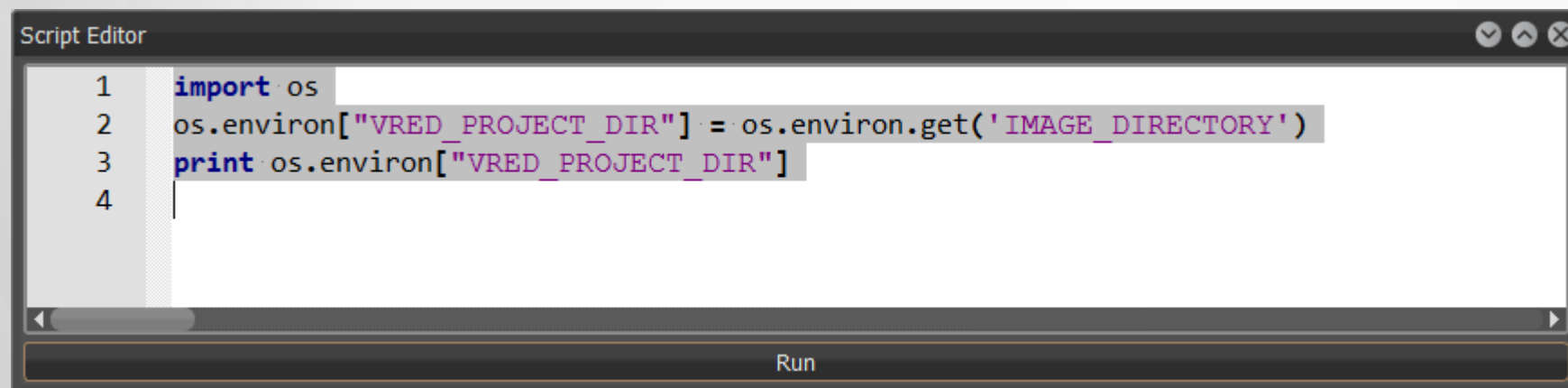
The following steps are needed to setup your VRED scene for render jobs:

- Set views or animations to be rendered
- Choose the relevant variants
- Define the level of quality that needs to be achieved
- Select the resolution and auxiliary, illumination channels
- Make use of the VRED sequencer for these tasks

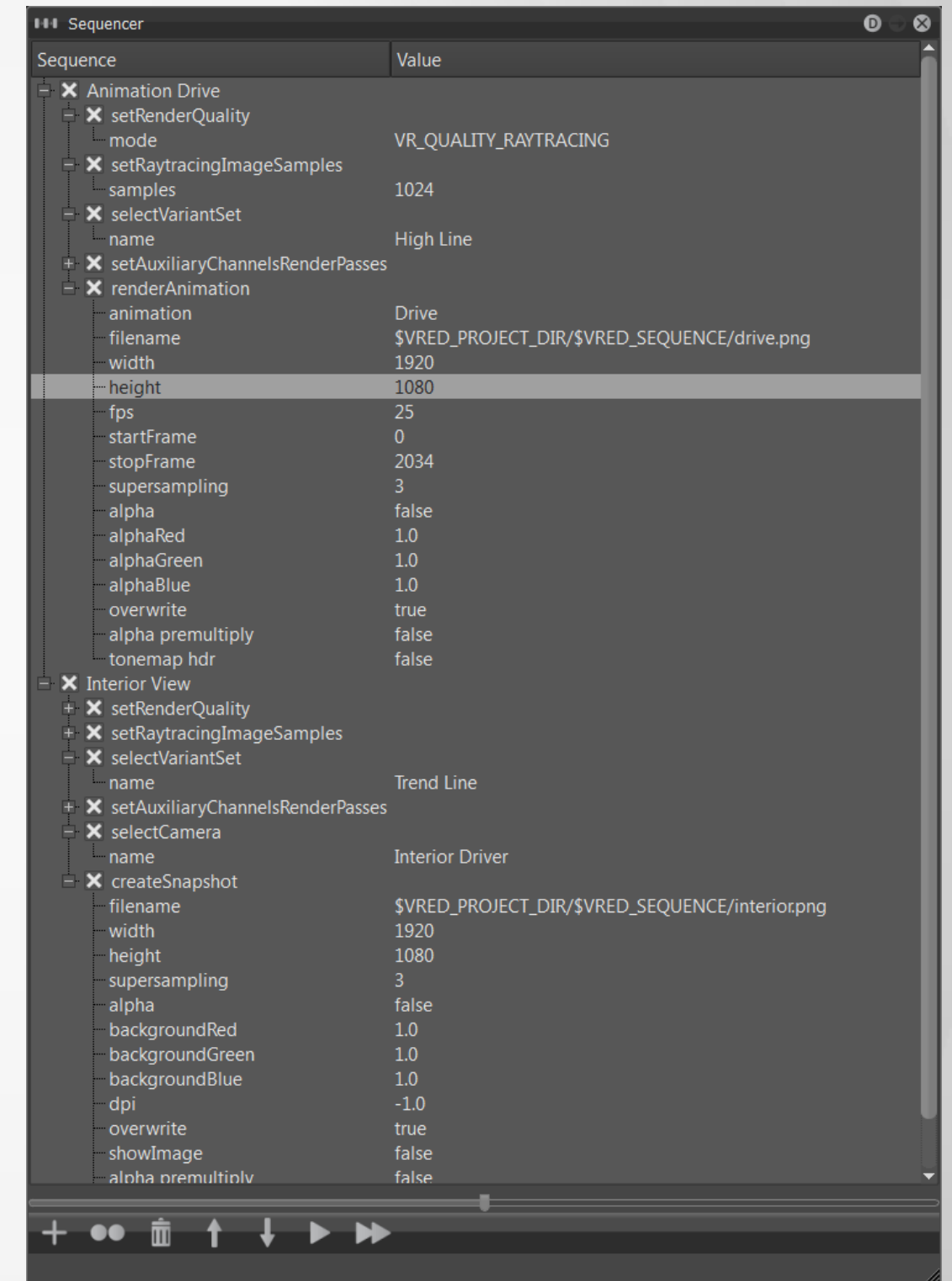
Setup a scene for large render Jobs

The screenshot example shows two sequences:

- Render a predefined animation
- Render a snapshot of a defined camera
- Define the target directory



```
Script Editor
1 import os
2 os.environ["VRED_PROJECT_DIR"] = os.environ.get('IMAGE_DIRECTORY')
3 print os.environ["VRED_PROJECT_DIR"]
4
Run
```



Setup a cloud/cluster environment

General cluster architecture

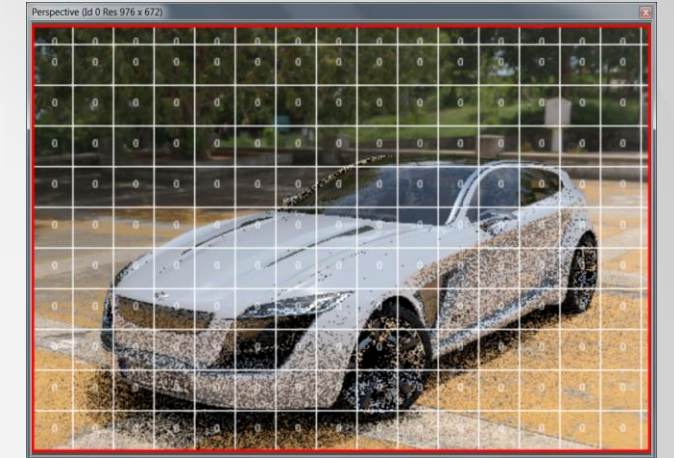
- Cluster Master:

- Loading the VRED scene
- Distributing the scene and controlling the Cluster Clients
- Saving the results



- Cluster Clients:

- Computing the image tiles and sending those back to the Cluster Master

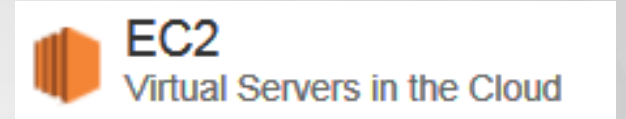


Setup a cloud/cluster environment

Two ways of using a cloud/server environment

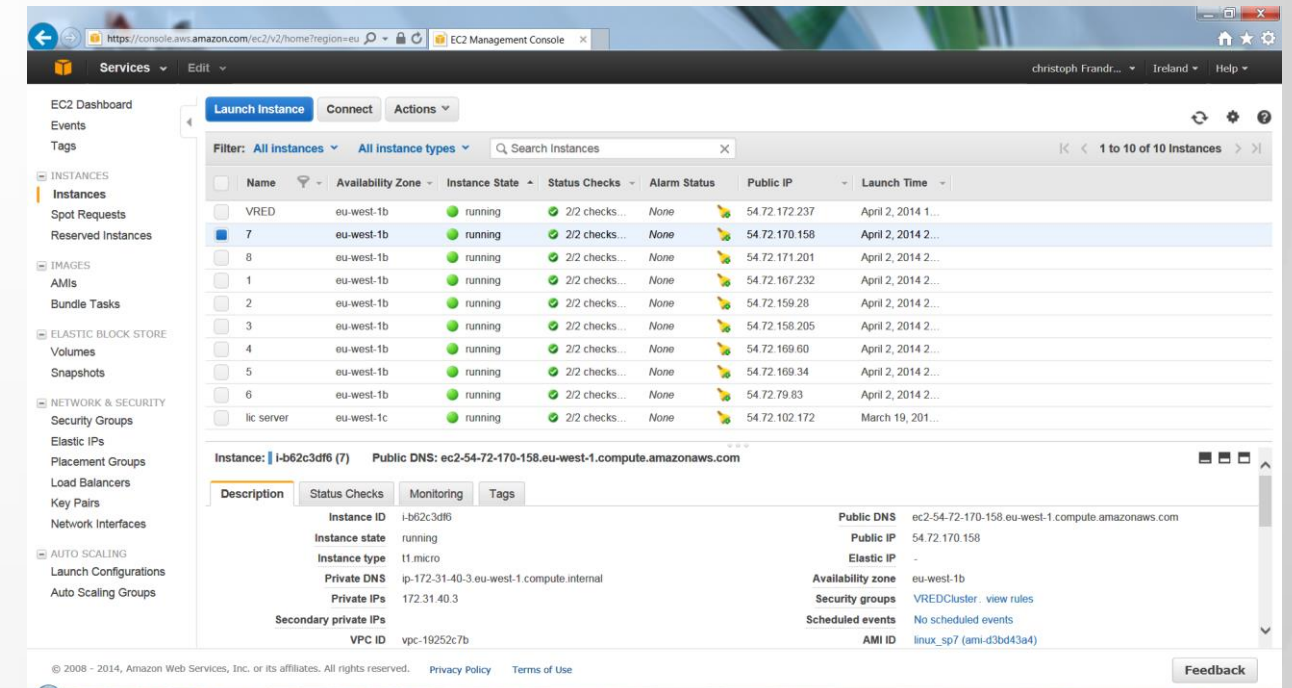
- Use the amazon cloud
 - Easy to access and easy to reserve multiple nodes
 - Full control of the whole environment and the timing
- Use of a private cloud/compute center
 - Feels more secure for some customers with sensitive data
 - Needs to be integrated into scheduling systems as no full time and exclusive access is possible
 - Unattended start and stop

Amazon cloud setup



Those steps are needed to use the Amazon cloud

- Start your instance
- Create a security group
- Install Software on Master and Clients
- Create images
- Restart using powerful instances



Amazon cloud setup

Start your instance

- To start an instance go to My Account / Console -> AWS Management Console and select EC2
- Select Instances and press **Launch Instance**
- Select Microsoft Windows Server 2012 Base
- The micro instance is chosen by default
- Start the instance
- Do the same for Linux

<input checked="" type="checkbox"/>	linux	t1.micro	eu-west-1b	● running	Initializing	None
<input type="checkbox"/>	windows	t1.micro	eu-west-1b	● running	Initializing	None

INSTANCES

Instances

Spot Requests

Reserved Instances

Launch Instance

Connect

Actions ▼

Amazon cloud setup

Create a security group

- When starting the real cluster we need a security group to be sure our instances are allowed to communicate
- We need three ports, those are the default ports:
 - 27000 for the license server
 - 8889 for the vred cluster service
 - 1234 for the vred cluster itself
- In the management console go to Security Groups and create one

Create Security Group

Security group name *i* cluster

Description *i* vred cluster

VPC *i* vpc-19252c7b (172.31.0.0/16) *
* denotes default VPC

Security group rules:

Inbound Outbound

Type <i>i</i>	Protocol <i>i</i>	Port Range <i>i</i>	Source <i>i</i>
Custom TCP Rule ▾	TCP	1234	Anywhere ▾ 0.0.0.0/0 ✕
Custom TCP Rule ▾	TCP	8889	Anywhere ▾ 0.0.0.0/0 ✕
Custom UDP Rule ▾	UDP	27000	Anywhere ▾ 0.0.0.0/0 ✕

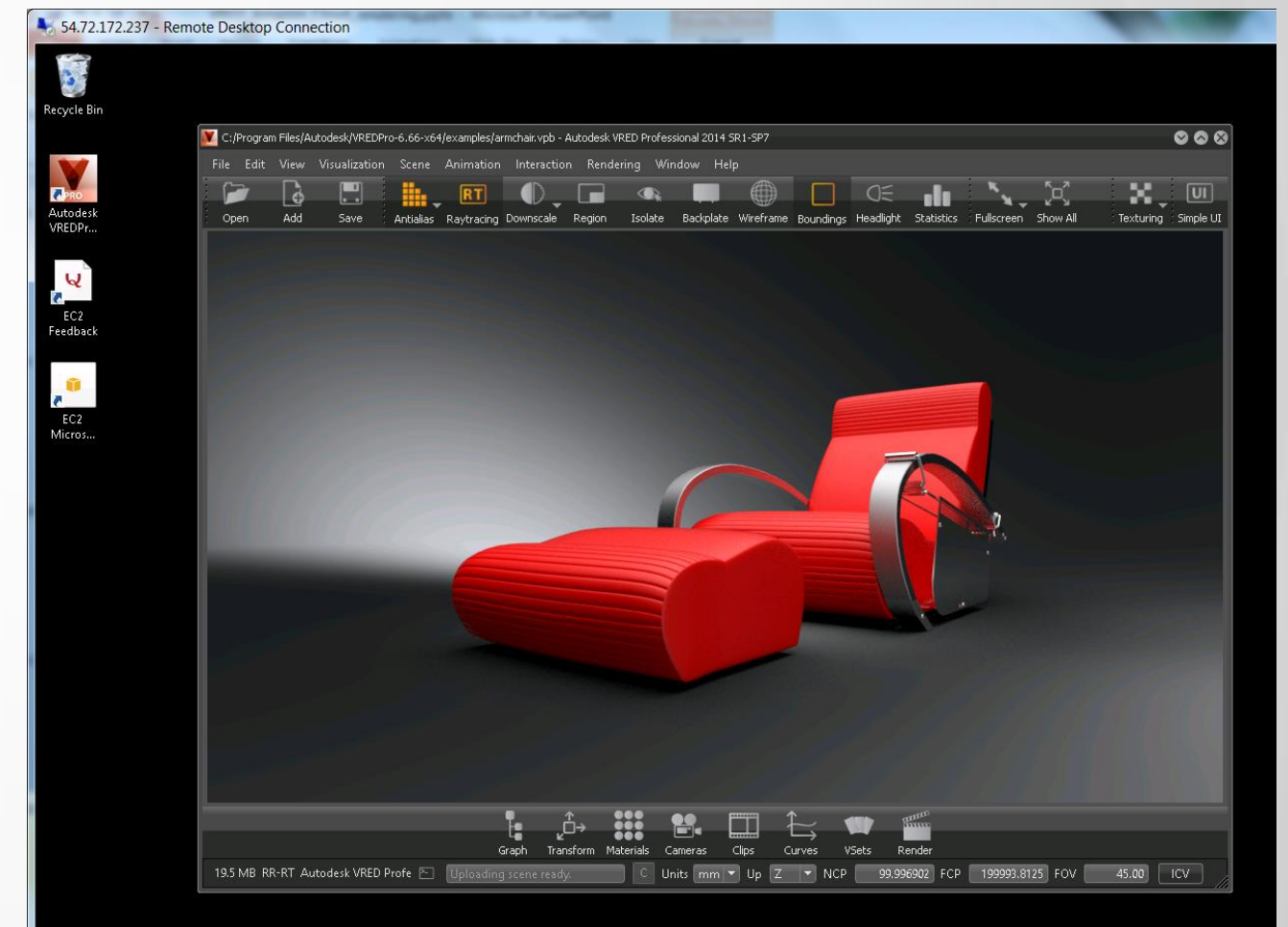
Add Rule

Cancel Create

Amazon cloud setup

Install Software on Master and Clients

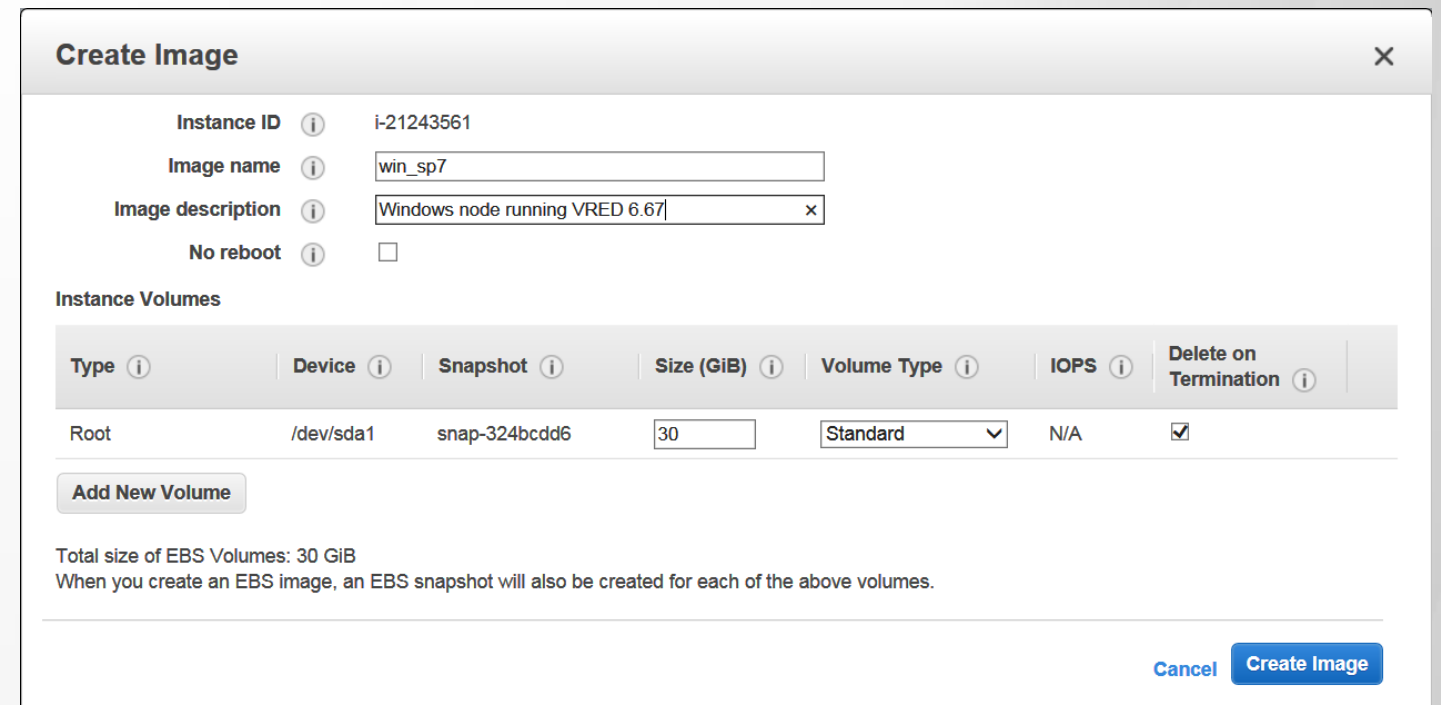
- Install
 - VRED Pro on your Windows instance
 - VRED cluster binaries on your Linux instance
- Make sure the clusterService starts at boot time



Amazon cloud setup

Create images

- Create images to use more powerful machines
- Select your Windows instance and from the actions choose **create image**
- Do the same for the Linux instance
- Next step is to boot more powerful machines using those images



The screenshot shows the 'Create Image' window in the Amazon Management Console. It includes fields for Instance ID (i-21243561), Image name (win_sp7), Image description (Windows node running VRED 6.67), and a checkbox for 'No reboot'. Below these is a table for 'Instance Volumes' with columns for Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, and Delete on Termination. A single volume is listed: Root, /dev/sda1, snap-324bcdd6, 30, Standard, N/A, with the 'Delete on Termination' checkbox checked. At the bottom, there is a summary of the total EBS volume size (30 GiB) and a note about EBS snapshots, along with 'Cancel' and 'Create Image' buttons.

Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Delete on Termination
Root	/dev/sda1	snap-324bcdd6	30	Standard	N/A	<input checked="" type="checkbox"/>

Amazon cloud setup

Restart using real instances

- Go to the Instances menu and choose **Launch instance**
 - Choose My AMIs
 - Launch a Windows instance
 - Launch x Linux instances
-
- Start rendering
 - Transfer images by e.g. BOX

The screenshot shows the AWS Management Console interface for launching an EC2 instance. The page is titled 'Step 3: Configure Instance Details' and includes a progress bar at the top with steps: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance Details (active), 4. Add Storage, 5. Tag Instance, 6. Configure Security Group, and 7. Review. The main content area contains several configuration options:

- Number of instances:** A text input field set to '10'.
- Purchasing option:** A checkbox for 'Request Spot Instances' which is currently unchecked.
- Network:** A dropdown menu showing 'vpc-19252c7b (172.31.0.0/16) (default)' with a 'Create new VPC' link.
- Subnet:** A dropdown menu showing 'subnet-98b79fec (172.31.32.0/20) | Default in eu-west-1' with a 'Create new subnet' link. Below the dropdown, it states '4082 IP Addresses available'.
- Public IP:** A checkbox for 'Automatically assign a public IP address to your instances' which is checked.
- Placement group:** A dropdown menu showing 'No placement group'.
- IAM role:** A dropdown menu showing 'None'.
- Shutdown behavior:** A dropdown menu showing 'Stop'.
- Enable termination protection:** A checkbox for 'Protect against accidental termination' which is unchecked.
- Monitoring:** A checkbox for 'Enable CloudWatch detailed monitoring' which is unchecked, with a link 'Additional charges apply.' below it.
- EBS-optimized instance:** A checkbox for 'Launch as EBS-optimized instance' which is unchecked, with a link 'Additional charges apply.' below it.
- Tenancy:** A dropdown menu showing 'Shared tenancy (multi-tenant hardware)' with a link 'Additional charges will apply for dedicated tenancy.' below it.

At the bottom right of the configuration area, there are buttons: 'Cancel', 'Previous', 'Review and Launch' (highlighted in blue), and 'Next: Add Storage'. The footer of the console shows copyright information: '© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved.' and links for 'Privacy Policy' and 'Terms of Use'. A 'Feedback' button is also present in the bottom right corner.

Private Cloud / Compute center

Those steps are needed to use a private cloud or a compute center

- Setup an unattended start and stop for the rendering process
 - The scheduling systems decides when to start on which resources
 - The complete start and stop mechanism has to work unattended
- Integrate the start scripts into the scheduling system
 - VRED start and Cluster start

Private Cloud / Compute center

Setup an unattended start and stop of the rendering process

- Set the target directory for the rendered images:

```
export IMGDIR=/scratch/project/images
```

- Start the rendering process

```
VREDPro -nogui -postpython "  
setRenderQuality(VR_QUALITY_RAYTRACING);  
startRaytracingCluster('/scratch/project/nodes.txt',0,$nodenum,8,32,0,true);  
runAllSequences();  
stopCluster();  
terminateVred()" $scene
```


Private Cloud / Compute center

Setup an unattended start and stop for the rendering process

```
VREDPro -nogui -postpython “
```

```
setRenderQuality(VR_QUALITY_RAYTRACING);  
startRaytracingCluster('/scratch/project/nodes.txt',  
    0,$nodenum,8,32,0,true);  
runAllSequences();  
stopCluster();  
terminateVred()" $scene
```

Start VRED and execute the following commands after \$scene is loaded

Switch to raytracing mode
Start the cluster

Run the predefined sequences
Stop the cluster
End VRED

Private Cloud / Compute center

Integrate the start scripts into the scheduling system

A typical job runs on ~40 nodes.

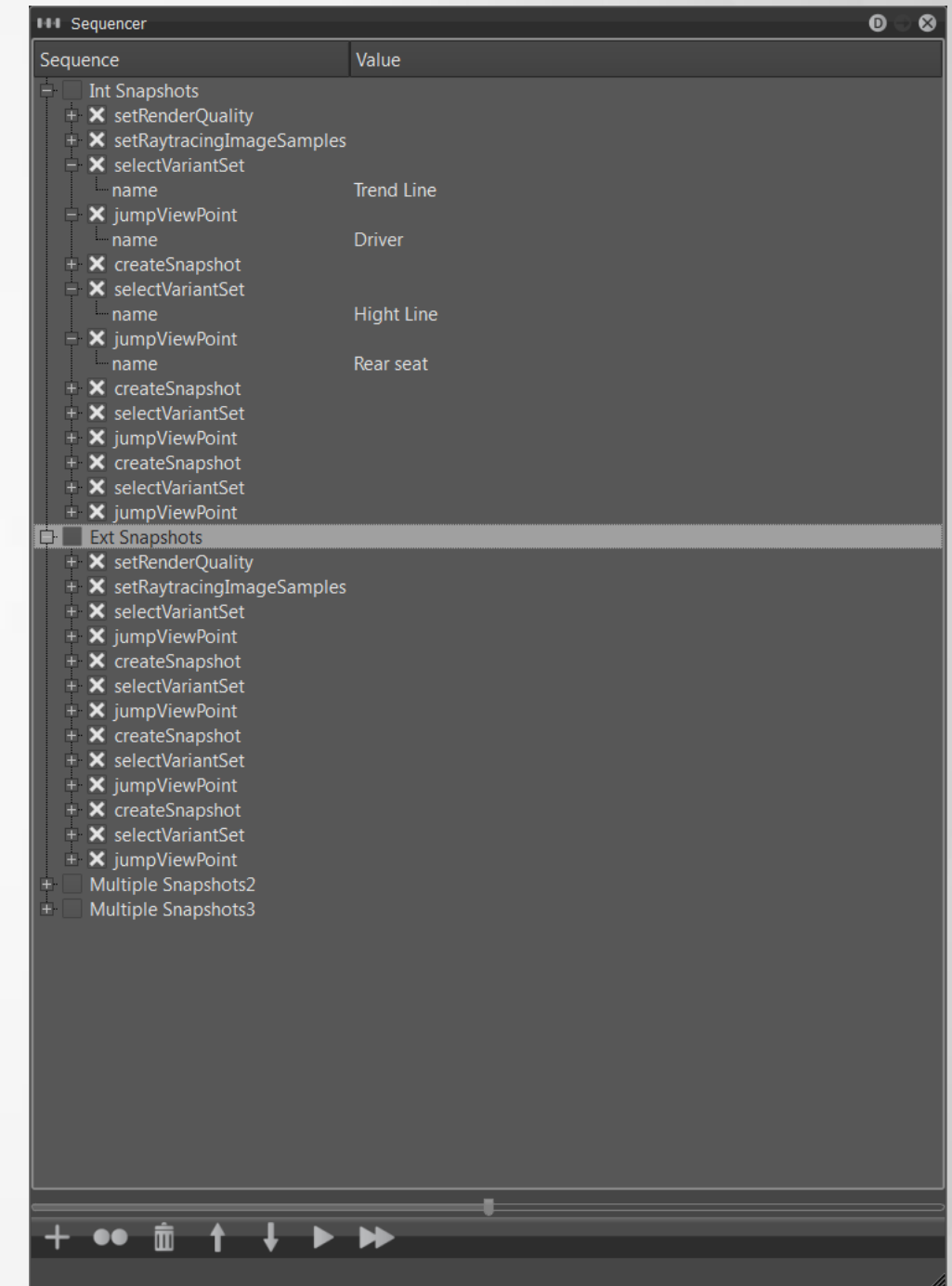
The scheduling system has to:

- Reserve the needed nodes e.g. 40
- Create a text file containing all cluster clients (nodes.txt)
- Start the clusterService on all cluster clients
- Copy the input scene into the project directory
- Start the VRED script from the chosen master (one out of the reserved)
- Notify the users when the Job is done



Best practices on compute time, price and performance

- The VRED scenes are setup to render multiple animations or images
- Sequences can be organized to contain
 - one image or animation
 - multiple images and/or animations
- By selecting the different sequences the work can be shared between more small size clusters



Best practices on compute time, price and performance



Advantages of splitting up the jobs:

- Makes scheduling easier and the chances are higher that the job starts
- Jobs can run in parallel on big clusters and sequential on smaller cluster

Split into several smaller jobs

- Risk of errors is smaller, one small Job can be restarted
- Lots of small jobs scale better than one large Job -> Saves compute hours



Example usage of VRED for design decisions

Present design alternative in known environments

Use real-time VRED scenes for movie creations

- A) Create the “environment”
 - Film and scan a real vehicle by the Russian Arm car
 - Model the environment, floor and street
 - Use the filmed content as a background for reflections and lighting
 - Create an animation fitting to the real drive



www.vimeo.com



www.vimeo.com

Example usage of VRED for design decisions

Present design alternative in known environments

- B) Render and Combine
 - Render the virtual car animation by using sequences
 - One image per frame
 - Use render passes for the different channels
 - Combine the rendered images with the real movie



Session Feedback

- Via the Survey Stations, email or mobile device
- AU 2015 passes given out each day!
- Best to do it right after the session
- Instructors see results in real-time







Students, educators, and schools now have

FREE access to Autodesk design software & apps.

Download at www.autodesk.com/education



Earn your professional Autodesk Certification at AU

Visit the [AU Certification Lab](#)