



12th Summer
School on
SCIENTIFIC
VISUALIZATION

Case History

Visualization in HPC environment

Paolo Ramieri - [p.ramieri@cineca.it](mailto:p.ramieri@ Cineca.it)
SuperComputing Applications and Innovation Department





Index

Remote visualization tools

- X server
- VNC (Virtual Network Computing)

Remote visualization facilities at CINECA

- RCM (Remote Connection Manager)
- EnginFrame VIEWS



Remote Visualization Tools



X Window System

The X Window System (commonly known as as X Window, X11 or simply as X, and sometimes informally X-Windows) is a widely used graphics manager, de facto standard for many Unix-like systems. Released under a free software license, since 2004 is maintained by the X.Org Foundation.

X provides the environment and the basic components for graphical interfaces or design, moving windows on the screen and the interaction with input devices such as mouse and keyboard.



X server

An X server is a server of connections to X terminals in a distributed network that uses the X Window System.

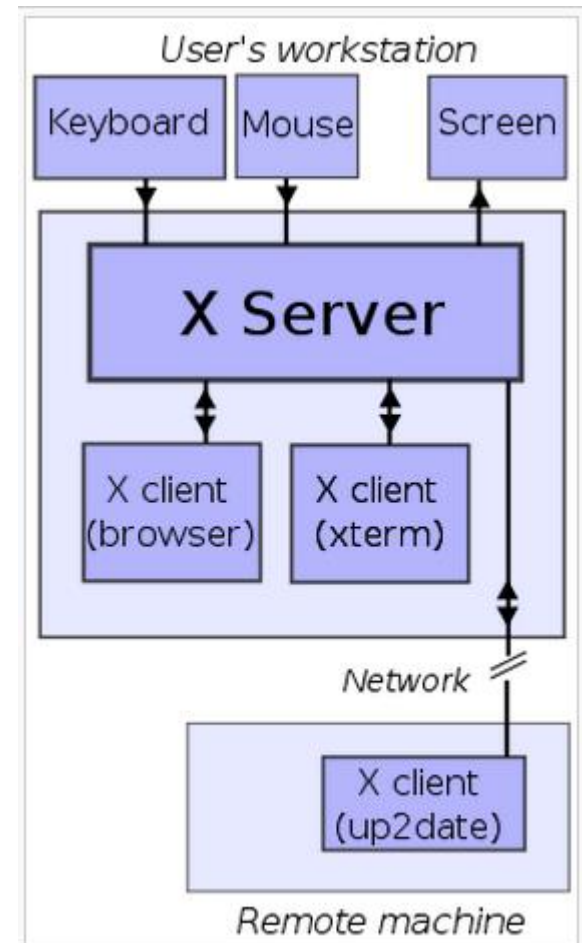
The terms client and server are often confused: for X, server is the local display of the user, not a remote machine.

From the terminal user's point-of-view, the X server may seem like a server of applications in multiple windows.



Example deployment of X server

The X server receives input from a local keyboard and mouse and displays to a screen. A web browser and a terminal emulator run on the user's workstation and a software update application runs on a remote computer but is controlled and monitored from the user's machine.





X server

- ✓ X-ming (freeware)
<http://sourceforge.net/projects/xming/>
- ✓ FreeXer - free X-server
<http://sourceforge.net/projects/freexer/>
- ✓ X-Win32 (commercial)
<http://www.starnet.com/xwin32/>

- ✓ xterm (Unix / Linux)

xterm is the standard terminal emulator for the X Window System. A user can have many different invocations of xterm running at once on the same display, each of which provides independent input/output for the process running in it (normally the process is a Unix shell).



VNC (Virtual Network Computing)

In computing, Virtual Network Computing (VNC) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to remotely control another computer. It transmits the keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network.

VNC is platform-independent – a VNC viewer on one operating system may connect to a VNC server on the same or any other operating system. There are clients and servers for many GUI-based operating systems and for Java. Multiple clients may connect to a VNC server at the same time. Popular uses for this technology include remote technical support and accessing files on one's work computer from one's home computer, or vice versa.

VNC was originally developed at the Olivetti & Oracle Research Lab in Cambridge, United Kingdom. The original VNC source code and many modern derivatives are open source under the GNU General Public License.



VNC system

A VNC system consists of a client, a server, and a communication protocol

- The VNC server is the program on the machine that shares its screen. The server passively allows the client to take control of it.
- The VNC client (or viewer) is the program that watches, controls, and interacts with the server. The client controls the server.
- The VNC protocol (RFB) is very simple, based on one graphic primitive from server to client ("Put a rectangle of pixel data at the specified X,Y position") and event messages from client to server.



VNC: how it works

The server sends small rectangles of the framebuffer to the client. In its simplest form, the VNC protocol can use a lot of bandwidth, so various methods have been devised to reduce the communication overhead.

For example, there are various encodings (methods to determine the most efficient way to transfer these rectangles). The VNC protocol allows the client and server to negotiate which encoding will be used.

The simplest encoding, which is supported by all clients and servers, is the raw encoding where pixel data is sent in left-to-right scanline order, and after the original full screen has been transmitted, only transfers rectangles that change.

This encoding works very well if only a small portion of the screen changes from one frame to the next (like a mouse pointer moving across a desktop, or text being written at the cursor), but bandwidth demands get very high if a lot of pixels change at the same time, such as when scrolling a window or viewing full-screen video.



VNC: client/server examples

- RealVNC: <http://www.realvnc.com/>
- TightVNC: <http://tightvnc.com/>
- UltraVNC: <http://www.uvnc.com/>
- TeamViewer: <http://www.teamviewer.com>



Remote visualization facilities at CINECA



Remote Connection Manager (RCM)

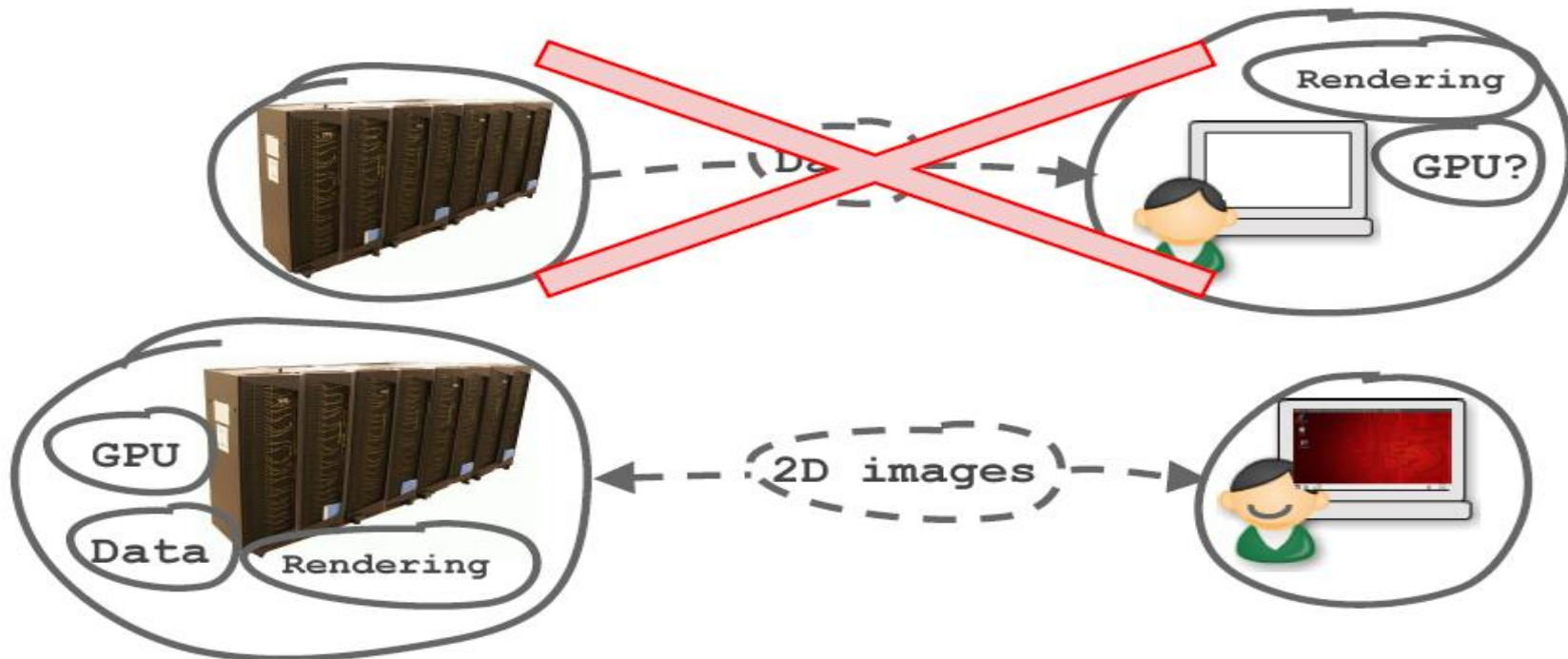
The Remote Connection Manager (RCM) is an application that allows HPC-users to perform remote visualization on PLX cluster.

The graphical interface of RCM allows to the HPC-users to create remote displays and manage them (connect, kill, refresh).



Objective

- Perform scientific visualization on large amounts of data produced on CINECA HPC systems
 - without moving data
 - using high performance machine





Infrastructure

- **NODES:**
 - **2 PLX compute nodes** (96, 97) (no inbound connection):
 - Processors: 2 six-cores Intel(R) Xeon(R) CPU E5645 2.40 GHz per node
 - GPU: 2 NVIDIA Tesla M2070Q per node
 - RAM: 48 GB/node
 - **2 RVN** (01,05) (inbound connection and login allowed):
 - Processors: 2 Quad-core Intel(R) Xeon(R) CPU X5570 2.93GHz per node
 - GPU: 2 NVIDIA QuadroPlex 2200 S4
 - RAM: 128 GB/node
 - The infrastructure is **scalable**: more nodes can be added
 - **Common home** and **scratch** filesystem mounted by login node, compute nodes and RVN nodes



Infrastructure (PBS)

- **PBS SCHEDULER:** VNC remote display are created through batch jobs
- **Queues:**
 - **visual** (*shared* resources on nodes 96/97):
 - default queue for standard users (nodes with 48 GB each)
 - **rvn_visual** (*shared* resources on nodes RVN05/01):
 - queue for specific users who need more RAM (node with 128 GB each)
 - **reserved** queue (*dedicated* resources on a node)
 - industrial users can request to reserve a node not to share resources with other users.
- Remote Display **WallTime** limit set to 6 hours.
- Number of concurrent displays a user can create is limited to **2**.



Technologies

- **TurboVNC**: free remote control software package that support VirtualGL. TurboVNC performs very well on high-latency, low-bandwidth networks.
- **VirtualGL**: open source package that gives any Unix or Linux remote display software the ability to run OpenGL applications with full 3D hardware acceleration. It optimizes user experience of remote 3D applications by rendering on remote GPU while streaming only the 2D result images.
- **Remote Connection Manager (RCM)**: python cross platform application developed by Cineca that simplifies and automates the steps needed for setting up a VNC connection to the visualization nodes (job submission for VNC server start, ssh tunneling, vnc client connection) and managing it (reconnection, list, close). It makes automatic all the step needed to **create, connect** and **kill** remote displays



Requirements

- To use the remote visualization service you need:
 - valid PLX user
 - username authorized to use the service (superc@cineca.it)
 - Internet connection
 - Pc with linux (most of the popular distributions), windows or mac on which install the RCM



RCM: login

USER

SYSTEM



RCM Login:

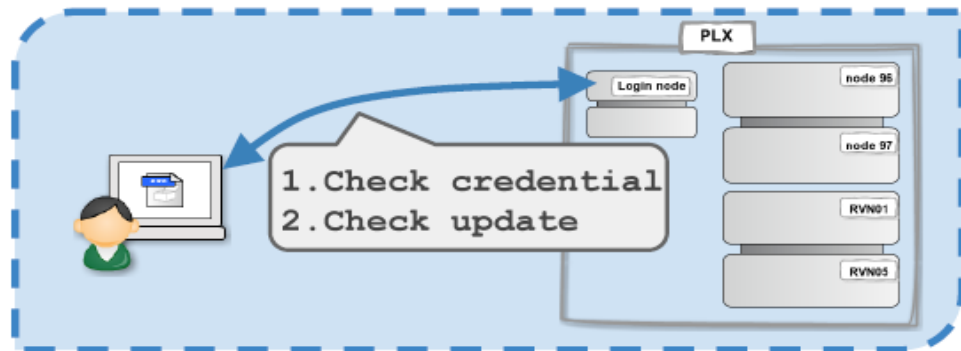
CINECA

REMOTE CONNECTION MANAGER
version: 1.0.268

User name:

Password:

LOGIN





RCM: display info

Remote Connection Manager 1.0.268 - CINECA

CREATED	DISPLAY	NODE	STATE	TIMELEFT	USERNAME	WALLTIME
20121105-09:03:50	2	node096ib0	valid	05:59:59	cin0588a	06:00:00

CONNECT KILL NEW DISPLAY REFRESH

Idle

Kill the remote display (kill the job)

Connect to the remote display

Create a new remote display

Refresh list of available displays

Information about created displays



RCM: new display

USER

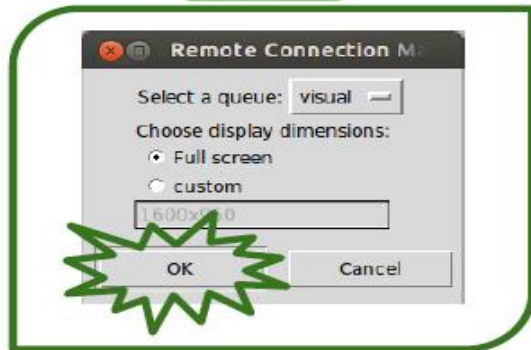
SYSTEM



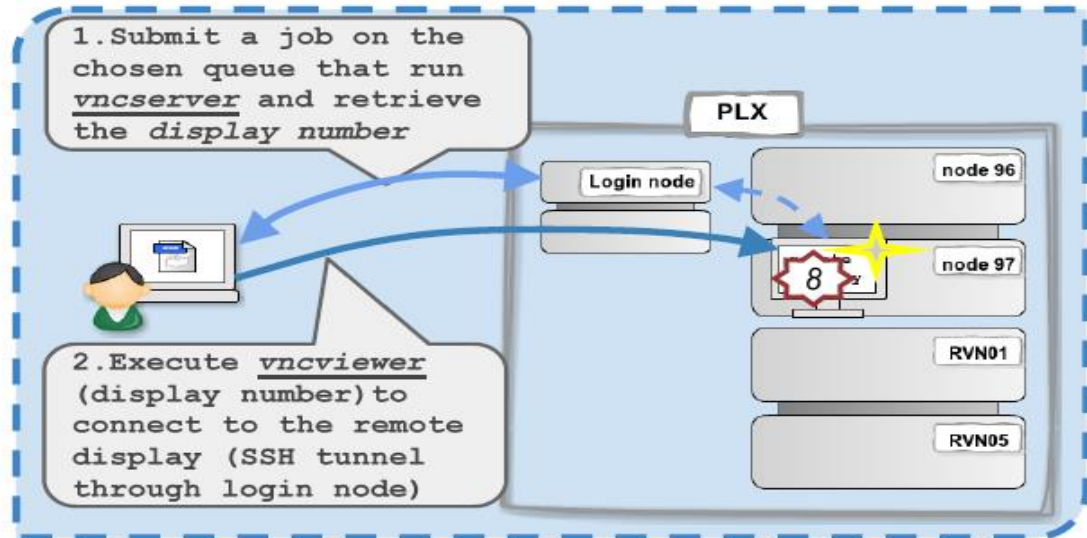


RCM: new display

USER



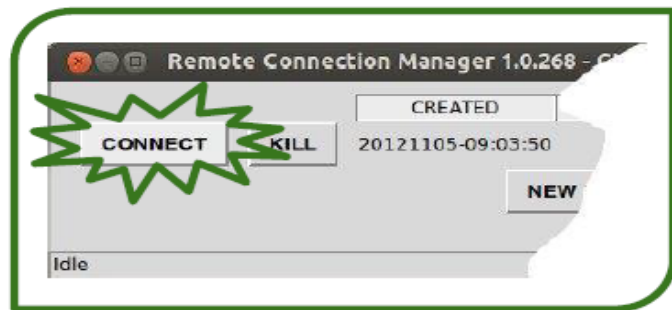
SYSTEM



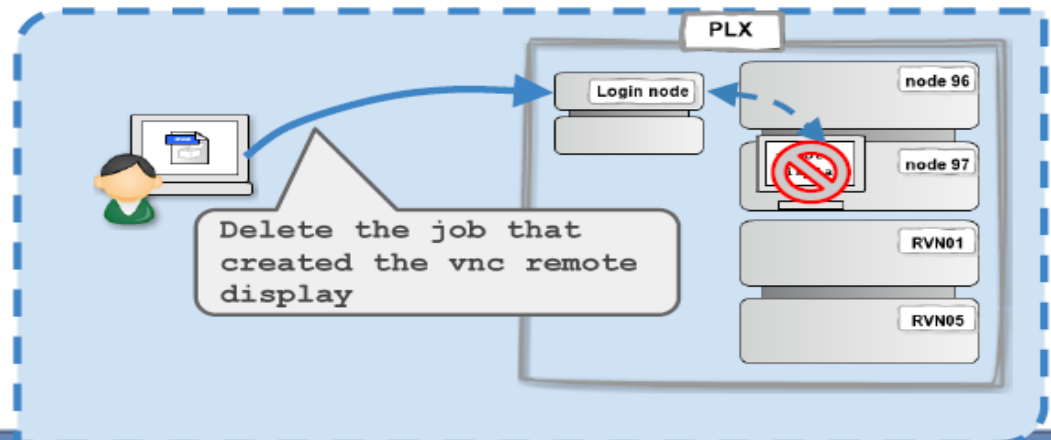
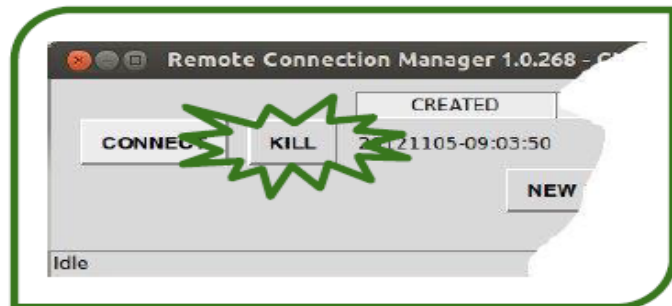
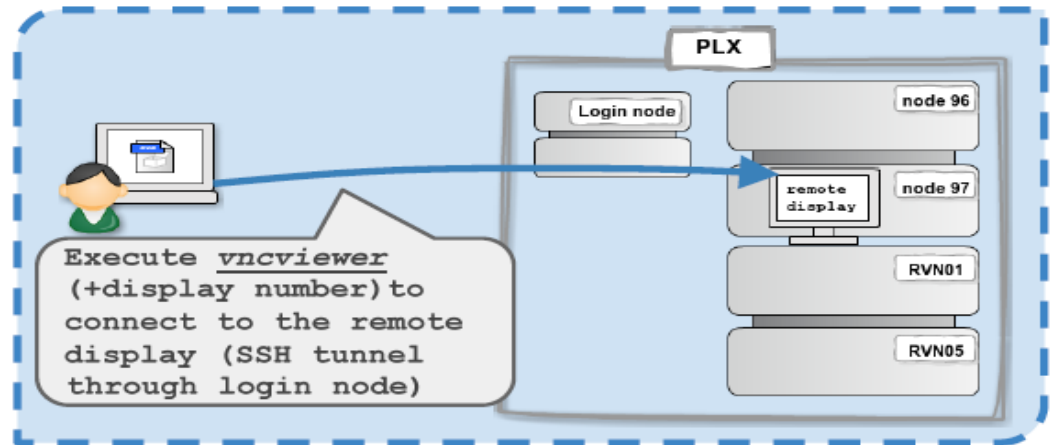


RCM: connect & kill

USER



SYSTEM





RCM: references

- RCM user documentation and download page: <http://www.hpc.cineca.it/content/remote-visualization>
- Service enabling request and problem report: superc@cineca.it
- IBM PLX user guide: <http://www.hpc.cineca.it/content/ibm-plx-gpu-user-guide>
- VirtualGL: <http://www.virtualgl.org/>
- TurboVNC: <http://virtualgl.svn.sourceforge.net/viewvc/virtualgl/vnc/trunk/doc/index.html>



CINECA's Computing Services Portal



Welcome, pramieri

HPC Services

Home

My Sessions

All Sessions

My Data

My Jobs

All Jobs

Cluster Load

File Manager

Help Desk

RADIOSS

Local User

ANSYS CFX

CFX 14

ANSYS WorkBench

Tools

XTerm

Pointwise

Pointwise 16

ParaView

Welcome to CINECA's Computing Services Portal

Pick an application from the menu on the left.

For visualization services please download and install VNC-DCV client:

- Windows users: [VNC-DCV Client for Windows](#)
- Linux users: [VNC-DCV Client for LINUX](#) (Untar the package and use the script "dcvviewer.sh" to open .vnc files)
- Mac OS X users: [VNC-DCV Client for Mac OS X](#)

<https://webcompute.cineca.it/engineframe>



EnginFrame Views: Key features

User friendly Web based access

Flexible service offering for end users

- Full 2D / 3D desktop

- Individual application publishing

Multiple remote display protocols support

- VNC

- NICE Desktop Cloud Visualization (DCV)

- HP Remote Graphics Server

Comprehensive load balancing policies

Built-in collaboration capabilities

Accounting and monitoring of resource usage

Comprehensive authentication options



NICE Desktop Cloud Visualization (DCV)

NICE Desktop Cloud Visualization (DCV) is an advanced technology that enables Technical Computing users to remote access 2D/3D interactive applications over a standard network.

In a typical visualization scenario, a software application sends a stream of graphics commands to a graphics adapter through an input/output (I/O) interface. The graphics adapter renders the data into pixels and outputs them to the local display as a video signal.

When using NICE DCV, the scene geometry and graphics state are rendered on a central server, and pixels are sent to one or more remote displays.

This approach requires the server to be equipped with one or more GPUs, which are used for the OpenGL rendering, while the client software can run on "thin" devices.

Web-based session management

The screenshot displays the MCAE Portal web interface, which is powered by engframe. The main navigation bar includes links for HOME, MY SESSIONS, MY DATA, MY JOBS, ALL JOBS, CLUSTER LOAD, FILE MANAGER, LICENSES, SETTINGS, and LOGOUT. The user is logged in as 'efadmin'.

The interface is divided into several sections:

- MCAE Services:** A tree view on the left lists various services including CFD Services (Fluent, CFX), FEA services (ANSYS), Crash services (LS-Dyna, PamCrash, Radloss), Desktop Sessions (Linux Session (2D/VNC), Windows Session (2D/VNC), Linux Session 3D, Windows Session (3D/DCV)), and Interactive Applications (ANSYS Workbench, ANSYS CFD Viewer, ANSYS FLUENT, 3D Navigator, CEI EnLiten, Avizo, ParaView, XTerm, Edit a File, Create Location).
- Session List:** A table with columns for Name, Status, Sharing, and Started on. It shows a session named 'ANSYS' with a status of 'Running' and 'Not shared', started on 'Today 11:42:quad'. There is also a 'Linux Desktop (2D/V)' session listed.
- Filters:** A section on the right allows filtering sessions, with 'Submitted Today' selected.
- 3D Model Viewer:** A window titled 'A: Geometry - DesignModeler' shows a 3D model of a mechanical part. The viewer includes a toolbar with various tools like Select, Undo, Redo, and a menu for 'Thin/Surface', 'Blend', 'Chamfer', 'Point', 'Generate', 'Share Topology', and 'Parameters'. The ANSYS 14.0 logo is visible in the top right of the viewer.
- NICE DCV Console - e4:** A console window in the bottom right shows the quality and bandwidth usage settings. The quality slider is set to 80. The console also displays bandwidth usage statistics: Current 698.5 KiB, Average 663.7 KiB, and Peak 1.4 MiB. A graph below these statistics shows bandwidth usage over a 50-second period.



Built-in collaboration

The screenshot shows the MCAE Portal interface. At the top, there's a navigation bar with 'MCAE Portal' and a search icon. Below that, the 'NICE' logo is on the left, and 'POWERED BY enginframe' is on the right. A user profile 'dcvuser' is visible in the top right corner. The main navigation menu includes 'MY SESSIONS', 'MY DATA', 'MY JOBS', 'ALL JOBS', 'CLUSTER LOAD', 'FILE MANAGER', 'LICENSES', 'SETTINGS', and 'LOGOUT'. The left sidebar shows a tree view of 'MCAE Services' with categories like 'CFD Services', 'Crash services', 'FEA services', 'Desktop Sessions', and 'Interactive Applications'. The main content area displays a table of running sessions. A context menu is open over the 'Avizo' session, showing options: 'View Details', 'Connect', 'Share Settings', and 'Close'. The 'Share Settings' option is highlighted in blue.

Name	Status	Sharing	Started on	Owner	Host
<input type="checkbox"/> XTerm	running		Mon 19:07:23	dcvuser	quad
<input type="checkbox"/> Edit /home/dcvuser/test.txt	running		Mon 19:10:24	dcvuser	quad
<input type="checkbox"/> Linux Desktop (2D/VNC)	running		Mon 19:14:23	dcvuser	quad
<input type="checkbox"/> ParaView	running		Mon 19:20:44	dcvuser	quad
<input type="checkbox"/> CEI EnLiten - EF 3D Views	running		Mon 19:24:37	dcvuser	quad
<input type="checkbox"/> GID	running		Yesterday 11:47:22	dcvuser	quad
<input type="checkbox"/> Avizo	running		Yesterday 17:36:44	dcvuser	quad

Copyright © 1998 - 2010 NICE s.r.l.
All trademarks and logos on this page are owned by NICE s.r.l. or by their respective owners.

Batch job / workflow submission



The screenshot shows the MCAE Portal web interface. At the top, there's a navigation bar with tabs: MY SESSIONS, MY DATA, MY JOBS, ALL JOBS, CLUSTER LOAD, FILE MANAGER, LICENSES, SETTINGS, and LOGOUT. The user is logged in as 'dcvuser'. The main content area is titled 'LS-Dyna' and contains a form for job submission. The form includes fields for Job Name, Project (set to 'Test'), CPU time (30 min), and Scratch size (1 GB). There's a 'Memory options' section with a dropdown for 'Computing queue' set to 'normal (Active, 0 jobs, 0 running)'. Below that are three 'Input file' fields with 'Select...' buttons, a 'Restart' dropdown set to 'No', and a 'Mail address' field. At the bottom, there's a 'Load profile' dropdown set to '- Select one -', a 'Save profile' button, and a text input field containing 'Dyna-profile'. A large 'Submit job' button is at the very bottom. The left sidebar shows a tree view of 'MCAE Services' including CFD Services, Crash services, FEA services, Desktop Sessions, and Interactive Applications.

**User friendly,
Application-oriented
Job submission**

**Hide complexity of
Underlying scheduler**

**Flexible and efficient
Input file management**

Monitoring

Jobs, Hosts, Queues, Licenses, ...

Host Name	Status	Jobs	CPU	Memory	Swap
apras02	Ok	0	0%	7.57 GB / 7.80 GB	2 GB / 2 GB
apras03	Ok	0	0%	7.57 GB / 7.80 GB	2 GB / 2 GB
apras04	Ok	0	0%	7.57 GB / 7.80 GB	2 GB / 2 GB
stria03	Ok	0	0%	7.10 GB / 7.80 GB	2 GB / 2 GB
stria04	Ok	0	0%	7.09 GB / 7.80 GB	2 GB / 2 GB
stria02	Ok	0	0%	7.07 GB / 7.80 GB	2 GB / 2 GB
bruja43	Ok	0	0%	6.85 GB / 7.80 GB	2 GB / 2 GB
stria11	Ok	0	0%	6.85 GB / 7.80 GB	2 GB / 2 GB
bruja26	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja38	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja27	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja04	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja25	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja06	Ok	0	0%	6.84 GB / 7.80 GB	2 GB / 2 GB
bruja09	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
bruja34	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
bruja35	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
bruja08	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
bruja36	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
stria08	Ok	0	0%	6.83 GB / 7.80 GB	2 GB / 2 GB
bruja31	Ok	0	0%	6.82 GB / 7.80 GB	2 GB / 2 GB
bruja03	Ok	0	0%	6.82 GB / 7.80 GB	2 GB / 2 GB

Queue Name	Status	Jobs	CPU	Memory
quad	Ok	0	13%	3.89GB/5.84GB
vizin01	Closed_Adm	0	---	314MB/498MB
vizin02	Unavail	0	---	---
vizwin01	Ok	0/1	8%	109MB/511MB
vizwin02	Unavail	0/1	---	---

FEATURE	IN-USE/TOTAL	IN-USE	TOTAL
eclipse	104/142	104	142
e300	25/25	25	25
compositional	39/100	39	100
rescoupling	9/120	9	120
parallel	3/12	3	12
cbm_template	0/100	0	100
cmt_template	0/17	0	17
coalbed	0/100	0	100
datacheck	1/100	1	100
eclipse4d	0/100	0	100
flux	1/124	1	124
foam	0/100	0	100
friction	1/141	1	141
frontsim	0/1	0	1
gasfield	0/110	0	110
gaslift	0/142	0	142
gasquality	0/100	0	100
geomchanics	0/100	0	100
gf_flogrid	0/4	0	4
gf_floviz	14/32	14	32



Application Data Management

Application data can be organized into projects

Application data can be marked as starred

Metadata can be associated to application data

The screenshot shows the 'enginframe' web interface for MDA Services. The main content area displays a table of application data with columns for Name, Project, Created on, and Delete on. The 'Project' column is highlighted with a red box. The 'Name' column also has a red box around the star icon. A 'Filters' sidebar on the right shows 'Projects' with a list of project names and a pie chart. The pie chart shows the distribution of projects: Car1 (1), Car2 (2), Car3 (1), and Default (6).

Name	Project	Created on	Delete on
Abaqus	Car1	Today 12:16:47	Jul 24 12:16:47
Nastran	Default	Today 12:16:55	Jul 24 12:16:55
Fluent simulation	Default	Today 12:17:02	Jul 24 12:17:02
PowerFlow	Default	Today 12:17:12	Jul 24 12:17:12
Radioss	Default	Today 12:17:20	Jul 24 12:17:20
PamCrash	Default	Today 12:17:25	Jul 24 12:17:25
Dyna	Default	Today 12:17:30	Jul 24 12:17:30
Nastran Test 2	Car2	Today 12:17:48	Jul 24 12:17:48
Fluent Simulation 2	Car2	Today 12:18:17	Jul 24 12:18:17
Test 3	Car 3	Today 12:20:38	Jul 24 12:20:38

Projects:

- Car1 (1)
- Car2 (2)
- Car 3 (1)
- Default (6)

Pie Chart Legend:

- Car1
- Car2
- Car 3
- Default



Data transfers & file management

The file manager component allows to seamlessly navigate and access server-side files from the web browser

File Edit View History Bookmarks Tools Help

nice-software.com https://quad.nice-software.com/ef2010demo/demo/eda/demo.ed

Google

engineframe Welcome, efaadmin | Logout

Home My Data My Jobs All Jobs Cluster Load File Manager

Upload Download Create Folder Delete Compress Refresh

Location: opt > nice > ef2010rc1 > engineframe > WEBAPP > icons

Places	Name	Size	Date Modified
efadmin's Home	16x16	--	Jul 02 09:06
File System	20x20	--	Jul 02 09:06
	22x22	--	Jul 02 09:06
	32x32	--	Jul 02 09:06
	dtree	--	Jul 02 09:06
	error	--	Jul 02 09:06
	mime-types	--	Jul 02 09:06
	rfb	--	Jul 02 09:06
	AC.gif	964 bytes	Jul 01 20:24
	COMPAQ.gif	2 KB	Jul 01 20:24
	CRAY.gif	1 KB	Jul 01 20:24
	HP.gif	2 KB	Jul 01 20:24
	IBM.gif	1 KB	Jul 01 20:24
	LINUX.gif	4 KB	Jul 01 20:24
	MAC.gif	6 KB	Jul 01 20:24
	MACOSX.gif	6 KB	Jul 01 20:24
	NEC.gif	2 KB	Jul 01 20:24

NICE Copyright © 1998 - 2010 NICE s.r.l. All trademarks and logos on this page are owned by NICE s.r.l. or by their respective owners.

Done

Architecture

