What's New in EF Portal 2024.0

Table of Contents

Support for new Operating Systems

NI SP EF Portal 2024.0 adds support for the following operating systems

- Redhat 8 and 9, CentOS 8 and 9, Rocky 8 and 9, AlmaLinux 8 and 9
- Ubuntu 22 and 24
- Amazon Linux 2023

File Manager Operations

Support for file, ssh and S3 remote spoolers

EF Portal now supports in addition to file spoolers, ssh and AWS S3 block storage remote spoolers
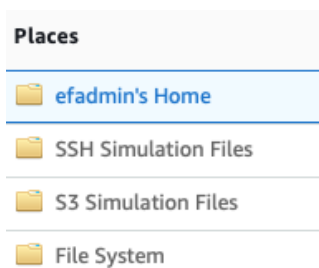
Please see also in the EF Portal Admin Guide "File and Remote Spooler Places in User File View"

Drag and Drop of Files into File Manager

With the drag and drop support files and directories can be dragged from the local machine into the EF files space including support for file, ssh and s3 remote spoolers so the user can e.g. drag files into a S3 bucket.
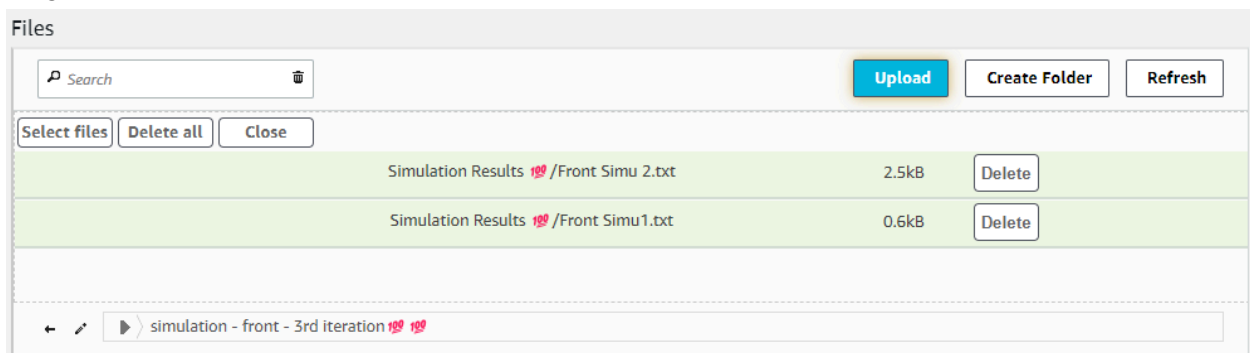
Support for Copy/Paste/Move operations on files and directories between spoolers, remote spoolers and places

When selecting files or directories the user is offered to copy or paste/move the files or directories. The user can move to a different spooler, remote spooler or place and copy or paste/move the files or directories into the present directory.

To enable this functionality the json parser **jq** should be installed. If it is not installed the user will also be informed by a message that jq should be installed.

Preview of data in spoolers supported for images, text and pdf as overlay in the file manager, draggable and resizable

EF Portal offers a convenient preview for text, PDF and image files. The preview appears when moving the mouse over the respective file.

Right click keeps the overlay in position. This preview can then be moved and resized as required. You can click on the overlay to close it.

Preview of image files

Hovering over the file name (e.g. the jpg file name below) shows the preview of the file.
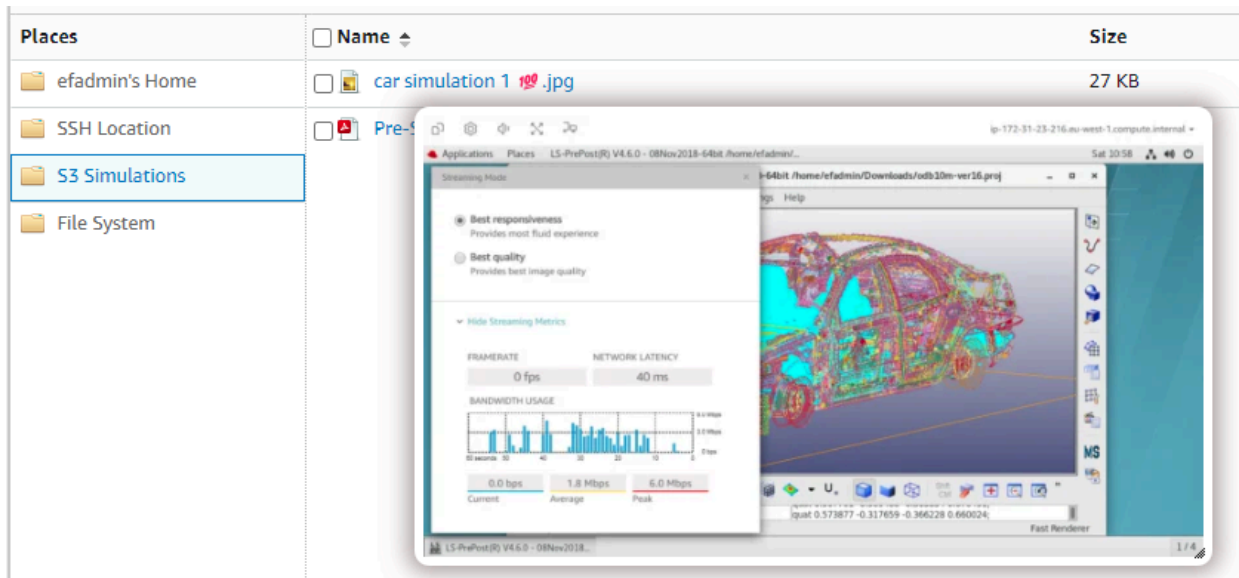
Preview of *.pdf files

Places	Name ▲	Size	Date Modified
efadmin's Home	Sandbox	--	Feb 13 13:27
SSH Simulation Files	NI SP EF Portal.pdf	69 KB	Today 15:50
S3 Simulation Files	Nice		
File System	Reco		
	mitsi		

Right click to keep the overlay; Click on the overlay to close it

Getting started

Topics
- • About NI SP EF Portal
- • Obtaining NI SP EF Portal
- • Planning a NI SP EF Portal deployment
- • Installing NI SP EF Portal
- • Running NI SP EF Portal

About NI SP EF Portal

EF Portal is a grid-enabled application portal for user-friendly HPC job submission, control, and monitoring. It includes sophisticated data management for all stages of job lifetime. It's integrated with most important job schedulers and middleware tools to submit, monitor, and manage jobs.

EF Portal provides a modular system where you can easily add new functionality, such as application integrations, authentication sources, and license monitoring. It also features a sophisticated web services interface that you can use to enhance existing applications and develop custom solutions for your own environment.

Preview of *.txt files

Places	Name ▲	Size
efadmin's Home	Sandbox	--
SSH Simulation Files	NI SP EF Portal.pdf	69 KB
S3 Simulation Files	Nice DCV.png	532 KB
File System	Record 12.txt	4 KB
	mitsimu3.inp	565 bytes

Right click to keep the overlay; Click on the overlay to close it

```
SOL 101
CEND
TITLE = NASTRAN(tm) Input File
ECHO = NONE
BEGIN BULK
GRID,1,,0.,0.,0.
GRID,2,,1.,0.,0.
GRID,3,,1.,1.,0.
GRID,4,,0.,1.,0.
CTRIA3,1,1,1,2,3
CTRIA3,2,1,1,3,4
CTRIA3,3,1,4,2,1
CTRIA3,4,1,3,2,4
PSHELL,1,1,0.1
```

File Editor for Text Files

When previewing a text file there is the option to start a File Editor to edit a file with different highlighting modes and themes.
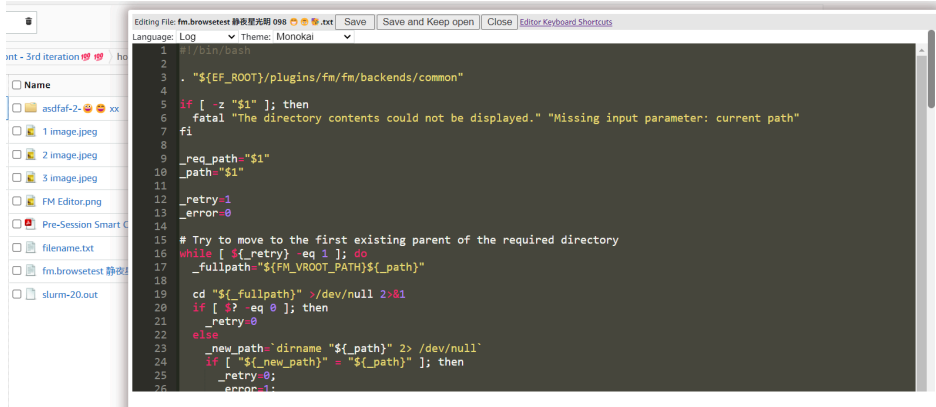
The file can then saved into the respective file location like standard file, ssh or S3 file locations.
.

Remember last location in a file manager and spooler and return to that path when viewing the "Place"

EF Portal always remembers the last directory you have opened. The following example shows the directory "Places/SSH Simulation Files/BaseSystem"
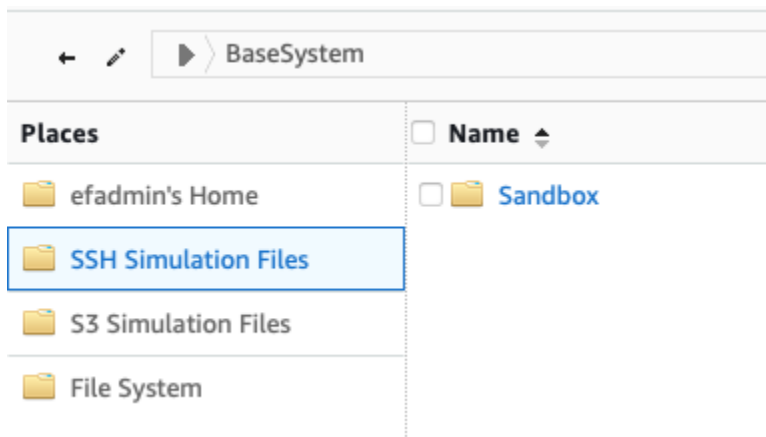
If you open a new folder structure under Places (example below: "Places/File System/bin") and want to return to the previous directory, just click on the desired root directory under Places and your previous directory will be entered again.
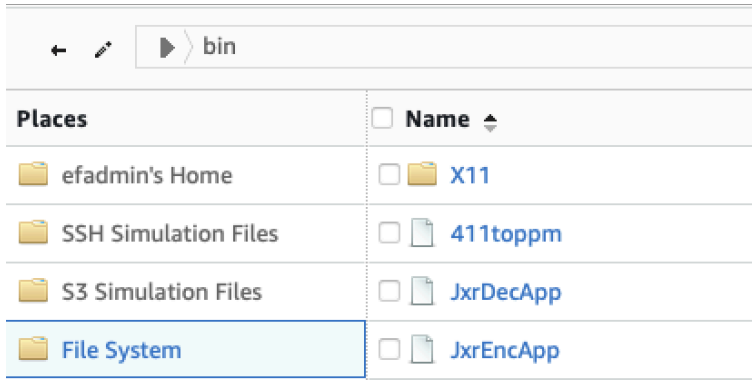
Show history of paths in places (spooler, filemanager) and allow user to jump to a selected previous locations

If you move the mouse pointer over "Places", EF Portal displays a history of up to the last 20 directories viewed.
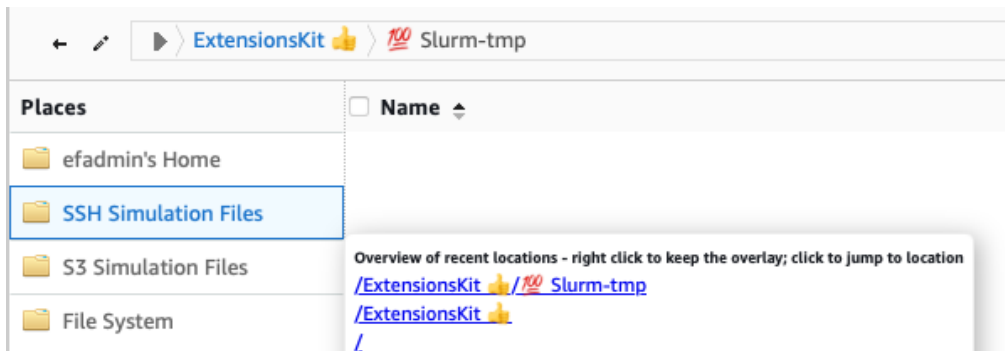
A right click keeps the overlay in position. Click on the directory to jump directly to the desired directory. This functionality can be toggled by hitting "t" after clicking on one of the places.

Support for UTF-8 Characters in Filenames

Filenames can include characters from extended character sets and emojis

Favorite Places: Additional File and Remote Spooler Places in User File View

NI SP EF Portal introduces easy creation of additional Favorite Places to allow for quick access to local or remote file locations frequently visited. In the example, below, the file locations Simu File Location, SSH Simulation Files and S3 Simulation Files are created under Places by putting the following configuration or code into ~/.ef/ef.places (permission 600):

The are two options to configure EF Places:

1) Configuration: A list of locations with URI and description separated by **~** (URIs should not include blanks):

```
file:///home/simu/simu_3_💯/simu_results~Simu File Location
ssh://simu@simu.ni-sp.com:/home/simu~SSH Simulations
s3://@simu-bucket/~S3 Simulations
```

2) A Bash script similar to the example below triggered by the SOURCE_ME string creating the places via EF syntax. You can just simply adapt the vroots array below or modify the bash script itself:

```bash
#!/bin/bash
# SOURCE_ME
# Create additional Places for quick access

# Indexed array, each element is "URI"~"Label"
vroots=(
    "file:///home/simu/simu_3_💯/simu_results"~"Simu File Location"
    "ssh://simu@simu.ni-sp.com:/home/simu"~"SSH Simulations"
    "s3://@simu-bucket/"~"S3 Simulations"
)

# Typically no need to change code below
for vroot_entry in "${vroots[@]}"; do
    # Split the entry into URI and Label using "~" as the delimiter
    IFS="~" read -r uri label <<< "$vroot_entry"

    # Create the virtual root using the URI and capture its ID
    _file_vroot=$("${EF_ROOT}/plugins/fm/bin/fm.vroot.create" "${FM_BROWSE_SPOOLER}" "fm" "$uri")

    if [ $? -ne 0 ]; then
        echo "Failed to create vroot for $uri"
        exit 1
    fi
    # Use the captured ID and Label to create the XML entry
    cat << EOF
  <hy:add-vroot id="${_file_vroot}">
```

```
        <hy:label>${label}</hy:label>
    </hy:add-vroot>
EOF
done
```

Resulting in the following additional Places:

Job History

A new Job History service was introduced to display job history data in table view and graphics which can be enabled in the Applications Admin Portal:

Daily:

Timeframe	Nr. of jobs	CPU time (hours; minutes; seconds)	Memory usage	Swap usage
29 December 2023	4	220:01:41.00	16.27 MB	22.97 MB
28 December 2023	5	212:35:13.00	35.2 MB	21.09 MB
27 December 2023	2	101:06:50.00	9.98 MB	16.37 MB
26 December 2023	3	123:02:40.00	11.94 MB	17.13 MB
25 December 2023	1	48:28:39.00	9.98 MB	6.91 MB
24 December 2023	2	139:35:59.00	14.02 MB	11.4 MB

Monthly aggregation:

Table | Graph

Show 10 entries

Search:

Timeframe	Nr. of jobs	CPU time (hours; minutes; seconds)	Memory usage	Swap usage
December 2023	94	4333:59:07.00	473.2 MB	505.78 MB
November 2023	94	4690:36:26.00	445.89 MB	410.36 MB
October 2023	77	3918:23:46.00	403.85 MB	369.44 MB
September 2023	91	4945:28:01.00	401.72 MB	456.4 MB
August 2023	74	3747:28:23.00	381.5 MB	353.16 MB
July 2023	98	4627:30:05.00	478.37 MB	483.06 MB
June 2023	65	3740:33:22.00	343.61 MB	282.94 MB

JOB HISTORY

Owner:	Queues:	Accounts:	Execution hosts:	Job Managers:	Status:
All	All	All	All	All	All

Aggregate report view:
Monthly

Date range

Table | **Graph**
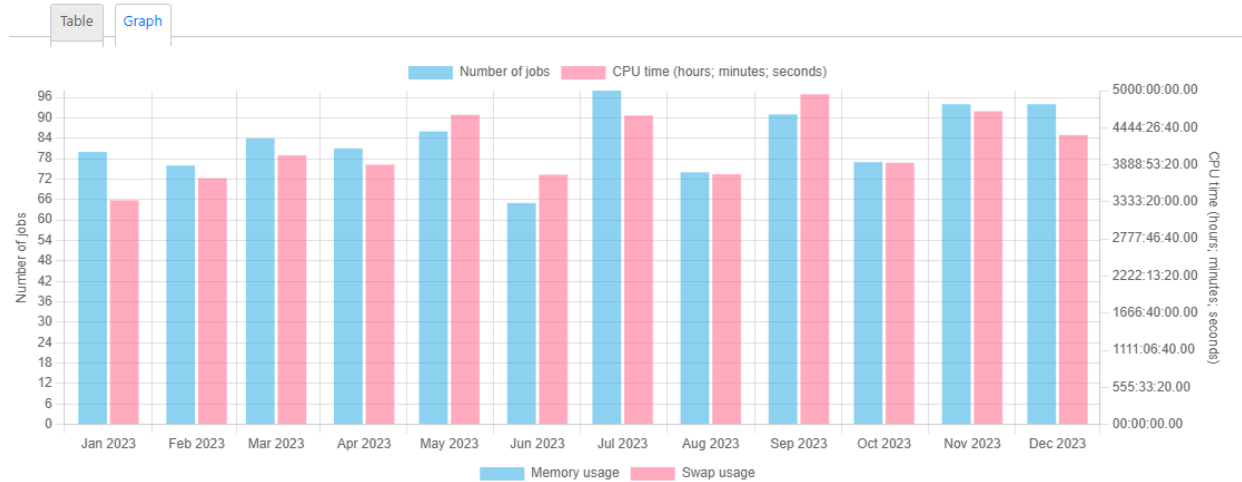
GPU Monitoring

EF Portal supports monitoring GPU compute load and memory load. Average of the GPU compute load and memory load over all GPUs is provided and visualized as well as the compute load and memory load of the individual GPUs.

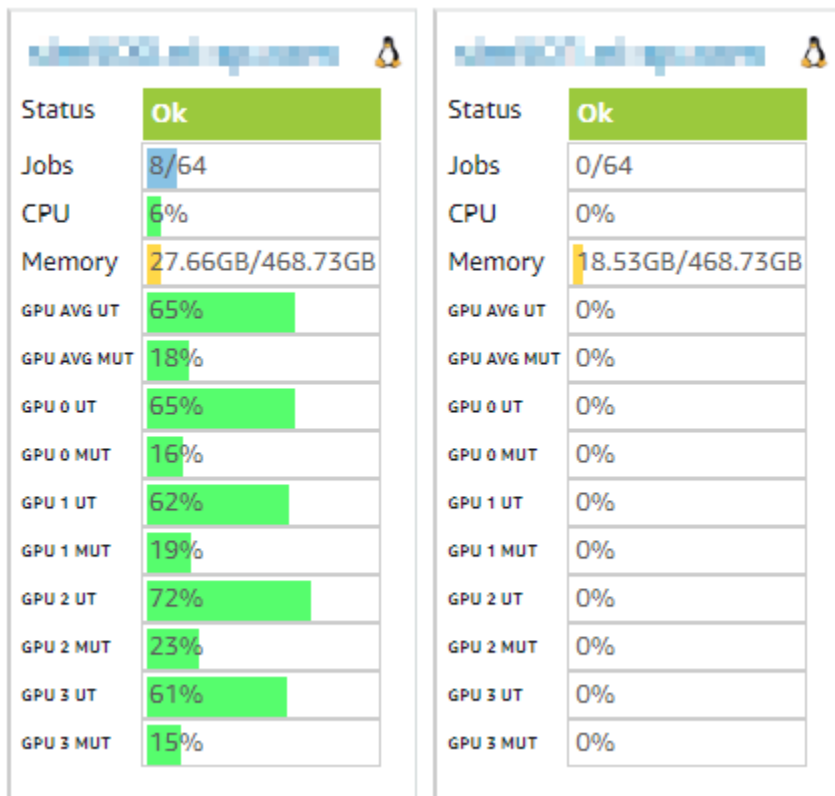GPU monitoring is supported for LSF with native GPU integration and custom GPU monitoring collection and display for all other schedulers like SLURM, DCV Session Manager, …

Here is a sample screenshot:

Status	Ok		Status	Ok
Jobs	8/64		Jobs	0/64
CPU	6%		CPU	0%
Memory	27.66GB/468.73GB		Memory	18.53GB/468.73GB
GPU AVG UT	65%		GPU AVG UT	0%
GPU AVG MUT	18%		GPU AVG MUT	0%
GPU 0 UT	65%		GPU 0 UT	0%
GPU 0 MUT	16%		GPU 0 MUT	0%
GPU 1 UT	62%		GPU 1 UT	0%
GPU 1 MUT	19%		GPU 1 MUT	0%
GPU 2 UT	72%		GPU 2 UT	0%
GPU 2 MUT	23%		GPU 2 MUT	0%
GPU 3 UT	61%		GPU 3 UT	0%
GPU 3 MUT	15%		GPU 3 MUT	0%

For other schedulers than LSF without GPU load integration the GPU load can be collected and stored for visualization in the EF Portal.

EF Portal will automatically display GPU load data stored in the directory $EF_TEMP_ROOT/gpumon (e.g. /opt/nisp/efportal/tmp/gpumon) with files gpu-log.$HOSTNAME. More information can be found in the EF Portal Admin Guide.

EF Views Session Management

DCV SM Integration

EF Portal adds following DCV Session Manager delegate options implementing all available DCV SM API session creation parameters:

- Added option to specify the **DCV storage root** to enable file transfer via DCV: –delegate-storageroot "path_to_directory", e.g. "%home%/Desktop"
- Added **Enqueue API support** with default to enqueue DCV SM sessions so that DCV SM sessions are not automatically terminated when resource request is not fulfilled and enqueue commandline option in interactive submit to control enqueue behavior. –delegate-enqueue (default is true)
- Added option to control **disable automatic retry of the request** if session creation fails: –delegate-disableretryonfailure (default is false)
- Added option for **command line parameters for the autorun delegate script** executed during initialization of the session; new parameter: –delegate-autorun-arguments

DCV options support for Linux scheduler VDI submit

EF Portal adds DCV option support for following Linux scheduler VDI submit options in the Linux Desktop action-script:
- –dcv2-permissions-file
- --dcv2-client-eviction-policy
- --dcv2-disable-login-monitor

Already supported options:
- –dcv2-gl
- --dcv2-storage-root   (e.g. --dcv2-storage-root %home%/Desktop)
- --dcv2-init
- --dcv2-max-concurrent-clients
- –dcv2-pid-check-ttl (controls the behavior of the check for the Xdcv process)

Larger Preview and Quick Access to DCV session
Added quick access to different connection options to NICE DCV sessions (DCV Connection File, DCV URI, DCV Web Client) on top of screenshot
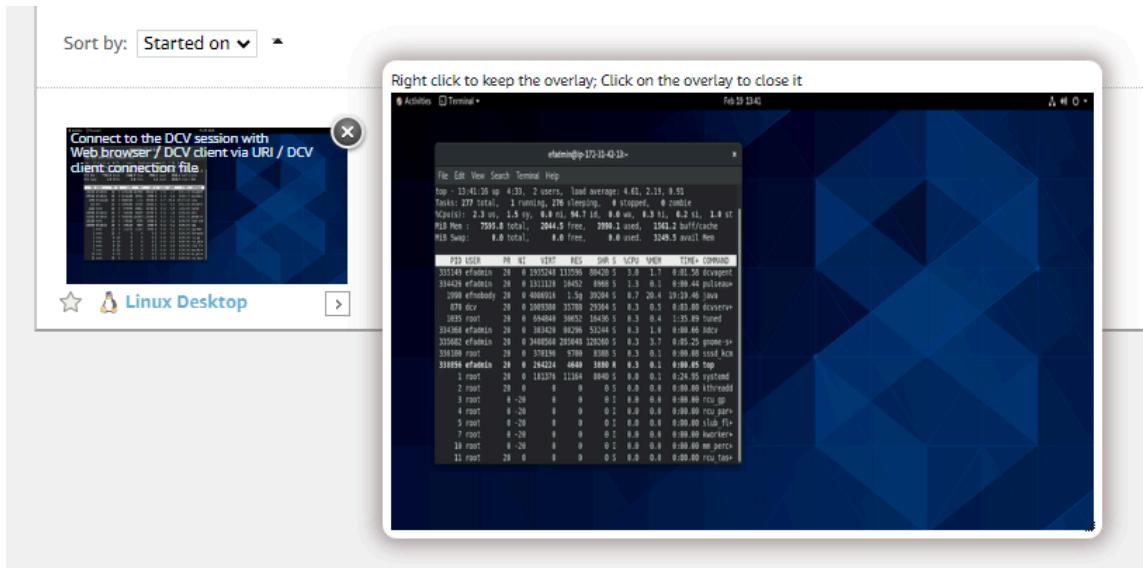
In addition a larger preview of the screenshot is shown when hovering over. The default screenshot size is set to 640x480.

Other Updates

SLURM Integration

- Fixed issue in display of tmp disk
- Fixes for job list
- Added support for SLURM 22.x and 23.x

LSF Integration

- Enhanced support for LSF 10.1.x
- Support for LSF MultiCluster for HPC jobs and VDI Sessions

Other changes

- Support search in multiple columns in Job View, File Manager, Spooler, …
- Show full hostname in title of hostview when hovering over hostname
- Updated the EF Portal installer to accept DCV SM as only scheduler/session manager

Security / Upgrades

- Updated Apache Tomcat to latest version 9.0.86 from Feb 14 2024; more here:

https://tomcat.apache.org/tomcat-9.0-doc/changelog.html
- Updated Apache Commons Text from 1.9 to latest version 1.11
- Upgrade to jQuery 3.7.1
- Upgrade to jQuery UI 1.13.2
- Upgrade to jqGrid 4.7