qemu-web-desktop: Installation

This documentation details how to install qemu-web-desktop. The Data Analysis Remote Treatment Service (DARTS) is an implementation of `qemu-web-desktop` at Synchrotron SOLEIL.

Table of contents:

1. Installation: The short story
2. Manual installation
3. Issues

Installation: The short story

System	Install	Quick configuration
Debian/Ubuntu/Mint	`sudo -E apt install qemu-web-desktop` (deb)	`sudo -E qwdctl enable slax.iso`
Arch/Endeavour/Manjaro	`yay -S qemu-web-desktop` (AUR)	follow final steps see https://aur.archlinux.org/packages/qemu-web-desktop
Apple/Darwin/MacOS (homebrew)	Install Brew then use `make deps; make install;` (no `sudo`).	`qwdctl enable debian-live-testing-arm64-gnome.iso` (no `sudo`)
RedHat/Fedora :warning:	`sudo dnf install make` then use manual install.	:warning: `qwdctl start VM` works, but the web service is broken
Manual Install	`git clone https://gitlab.com/soleil-data-treatment/soleil-software-pro` `cd qemu-web-desktop; sudo make deps; sudo make install;`	`sudo -E qwdctl enable slax.iso` `qwdctl edit machines`

Then, if you are in a hurry:

1. Enable a virtual machine as listed by `qwdctl status`, for instance `sudo qwdctl enable slax.iso`.
2. Open the URL http://localhost/qemu-web-desktop/, enter a fake user ID (the authentication is inactivated at start), and click on Create. A VM should start.

It is recommended to follow the CONFIGURE.md notes (e.g. `sudo qwdctl edit config`). If you wish to use a GPU, have a look at GPU.md. To manage virtual machines, refer to VIRTUAL_MACHINES.md.

:warning: This service is currently only validated on intel/amd64/`x86_64` and arm64/`aarch64` architectures.

Manual installation

Hardware requirements The service will launch virtual machines for which some memory and CPU (possibly GPU) are allocated. However, the QEMU memory and CPU management allows to redistribute the unused resources, so that in practice the total amount of requested CPU and memory can exceed the physical limitations. This relies on the assumption that not all sessions will use the full requested CPU and memory at the same time.

For most purposes a simple 'gamer' machine should suffice, with typically 4 GB memory per available core, to run most virtual machines. We have tested for instance the service on 64-core computers with 256 GB memory. Last, you may even run QWD from *inside* a running virtual machine, in case you wish to completely virtualize your infrastructure (performance may be lower still).

Software requirements (dependencies): Get the source code (or a release):

- git clone https://gitlab.com/soleil-data-treatment/soleil-software-projects/qemu-web-desktop
- :warning: if you just cloned the repository, make sure you collect the `html/desktop/machines` directory with LFS. Install it with `sudo apt install git-lfs` then, from the repo, `git-lfs install; git lfs pull`.
- you may alternatively get a release at https://gitlab.com/soleil-data-treatment/soleil-software-projects/qemu-web-desktop/-/releases
- `cd qemu-web-desktop`

The following dependencies are installed with:

- `sudo make deps` (no `sudo` under MacOS)
- **Debian-class** systems (Ubuntu/Mint...): `apache2 libapache2-mod-perl2 novnc websockify qemu-kvm bridge-utils qemu iptables dnsmasq libcgi-pm-perl liblist-moreutils-perl libsys-cpu-perl libsys-cpuload-perl libsys-meminfo-perl libnet-dns-perl libproc-background-perl libproc-processtable-perl libemail-valid-perl libnet-smtps-perl libmail-imapclient-perl libnet-ldap-perl libemail-valid-perl libjson-perl libwww-perl libguestfs-tools libapache2-mpm-itk libtext-qrcode-perl libnet-ssh2-perl wget` and on arm64/aarch64 architectures `qemu-efi-aarch64` (EFI boot)

- **Arch-class** systems (Endeavour, Manjaro, Cauchy): `python-numpy python-setuptools apache mod_itk qemu-desktop bridge-utils qemu-img dnsmasq guestfs-tools perl-cgi perl-list-moreutils perl-sys-cpu perl-sys-meminfo perl-net-dns perl-proc-processtable perl-mail-imapclient perl-ldap perl-json perl-libwww mod_perl novnc websockify perl-proc-background perl-email-valid perl-net-smtps perl-text-qrcode perl-net-ssh2 wget`
- **Fedora/RedHat-class** systems: `mod_perl novnc python3-websockify guestfs-tools perl-CGI perl-List-MoreUtils perl-Sys-CPU perl-Sys-MemInfo perl-Net-DNS perl-Proc-Simple perl-Proc-ProcessTable perl-Email-Valid perl-Net-SMTPS perl-Mail-IMAPClient perl-LDAP perl-JSON perl-libwww-perl perl-Text-QRCode perl-Net-SSH2 perl-CPAN wget`
- **Apple/Silicon** systems: `apache2 qemu perl python3 python-setuptools grep wget gnu-sed nano cpanminus` (homebrew)

Get the source code and install Get the source code as above and `cd qemu-web-desktop`.

Then install QWD with:

- `sudo make install` (no `sudo` under MacOS)

In case the `make` command fails, have a look at the Makefile...

:warning: In case you manually install the software, and target directories are customized, make sure to edit the `qwdctl` file to change the filepath to match configuration files (at start).

Test the service It is possible to test that all works with:

```
cd qemu-web-desktop
make test
```

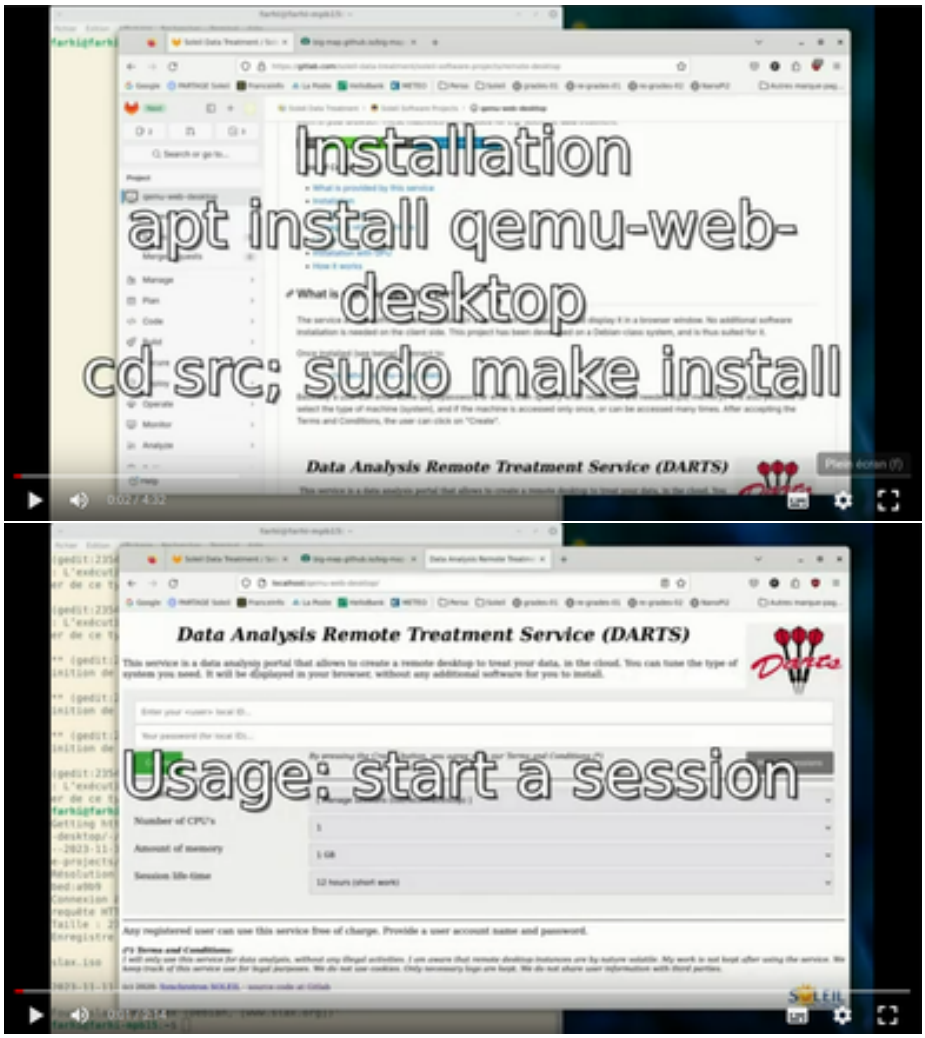
A URL will be displayed, and the session will then be closed after 30 seconds.

Uninstall To un-install, just do a:

- `sudo make uninstall`

or remove the distribution package (Debian, Redhat, Arch).

We have prepared two videos that demonstrate the semi-automatic procedure (Debian).



Building a Debian package

To build a Debian package out of this repository or https://salsa.debian.org/debian/qemu-web-desktop, use:

```
sudo apt install git-buildpackage dh-apache2 dh-sysuser devscripts help2man pandoc
make deb
```

will create a `.deb` package in the directory level above.

Building an Arch package

```
cd packaging/arch
```

```
makepkg -s
# install
pacman -U ./qemu-web-desktop*.zst
```

Issues

Refer to the end of the CONFIGURE.md documentation.