Issues (problems) faced during my project

- unable to download the tools because of low connection quality (time consuming) and also when the power goes off it starts from zero

- lack of mentor and well organized tutorial with clear language on platforms like youtube(makes hard to figure out alot of things especially while installing and updating opensuse , things that are too professional showed up and i was unable to fix it quickly .)

-Screen resolution was limited

-Network not workinging VM

Solutions

1. Use a download manager like wget -c or aria2c that supports resume capability.

2. Schedule downloads during off-peak hours for better speed.

3. Use VirtualBox snapshots to save progress frequently.

4. Refer to official openSUSE documentation and forums for clear guidance.

5. Join online communities (Reddit, Stack Overflow, openSUSE Forums) to get help.

6. Break down complex tasks into smaller steps and document them.

7. Install VirtualBox Guest Additions to fix screen resolution issues.

8. Enable “Auto-resize Guest Display” in VirtualBox setting

9.Switch VirtualBox network settings to Bridged Adapter or NAT

10. Restart network services using:

11.sudo systemctl restart network

12. Bring up the network interface with:

```
sudo ifconfig eth0 up
sudo dhclient eth0
```

Install VirtualBox Extension Pack compatible with your version.

File system supports

openSUSE supports a wide range of file systems, both for general-purpose use and specific scenarios. Here's a list of file systems commonly supported by openSUSE:

1. Btrfs (default for root in openSUSE)

Features: Snapshots (with Snapper), subvolumes, checksums, compression, copy-on-write.

Best for: Root file system, systems where rollback and versioning are important.

Note: openSUSE's YaST and Snapper work tightly with Btrfs.

2. XFS

Features: High-performance journaling, scalable, great for large files and parallel I/O.

Best for: /home or data partitions, servers.

3. ext4

Features: Journaling, widely supported, stable.

Best for: Compatibility, simplicity, and general use.

4. ReiserFS (legacy)

Status: Still available, but largely deprecated in favor of ext4 or Btrfs.

Best for: Not recommended for new installations.

5. FAT32, exFAT

Use case: External drives, USB sticks, compatibility with Windows/macOS.

Note: exFAT support is included by default in recent openSUSE versions.

6. NTFS

Use case: Read/write access to Windows partitions.

Supported by: ntfs-3g driver.

7. ZFS

Status: Not officially supported due to licensing, but can be installed via third-party repositories.

Best for: Advanced setups needing features like deduplication, snapshots, etc.

8. ISO 9660 / UDF

Use case: Optical media (CD/DVD).

Advantages and disadvantages of openSUSE

Using openSUSE as an operating system has several important benefits, especially for developers, system administrators, and Linux enthusiasts. Here's why openSUSE is valuable:

1. Stability and Reliability

openSUSE, especially the Leap version, is known for its rock-solid stability as it shares the same core as SUSE Linux Enterprise (SLE).

It's ideal for production environments where uptime and reliability are critical.

2. Cutting-Edge Option with Tumbleweed

The Tumbleweed version provides a rolling release model, meaning you get the latest software updates and Linux features as soon as they're stable.

Perfect for developers who want the newest tools and technologies.

3. YaST – Powerful System Administration Tool

YaST (Yet another Setup Tool) makes system configuration easy. It’s an all-in-one control panel for managing software, hardware, network, and more.

4. Strong Security Features

openSUSE includes AppArmor, robust user permission settings, and timely security updates.

It supports encryption, firewalls, and secure boot options.

5. Developer and Server Friendly

Comes with powerful developer tools and is optimized for software development and server deployments.

Supports Docker, Kubernetes, and virtualization tools like KVM and Xen.

6. Active Community and Documentation

Strong community support through forums, IRC, and mailing lists.

Detailed documentation and tutorials available for both beginners and advanced users.

7. Customizability

openSUSE gives you full control to customize your environment — from the desktop environment (KDE, GNOME, etc.) to system behavior.

8. Free and Open Source

100% free to use with no licensing costs.

Transparent development process, allowing users to contribute or modify the OS.

Below are some disadvantages of openSUSE

1. Not Beginner-Friendly

openSUSE can feel a bit complicated if you’re new to Linux.

Tools like YaST are powerful, but they might be overwhelming at first.

2. Less Software in Default Repos

Compared to Ubuntu, openSUSE might not have as many apps available in its default software center.

You might need to add extra repositories or use the command line more often.

3. Community Over Corporate Support

Whiles it has great community help, it doesn't have official support like some other enterprise Linux systems (unless you use SUSE Linux Enterprise, which costs money).

4. Slow Boot Time (in some cases)

Some users report that openSUSE, especially Leap, can boot slower than other Linux distributions.

5. Tumbleweed Risks (if you use it)

if you use the Tumbleweed version, which updates constantly, you might run into bugs or broken packages if you’re not careful.

About virtualizatio in Modern Operating Systems

Virtualization is the process of creating virtualinstancesbof computing resources. It allows running multiple operating systems simultaneously on a single physical machine using software like VirtualBox,VMware,orKVM.

Why we use virtualization?

-Safe environment for testing

-Easy snapshot and rollback

-Reduces hardware costs

Howdoesitwork?

Virtual Box creates avirtual environment that emulates hardware.The guestOS(openSUSE) interacts with this virtual hard ware as if it were real ,venabling isolated testing and development.

1. System Update & Development Tools

```
sudo zypper refresh && sudo zypper update
```

This refreshes the software list and updates your system to the latest versions.

```
sudo zypper install -t pattern devel_basis
```

Installs a group of development tools (like compilers, libraries, etc.) needed for building software.

```
sudo zypper install gcc make kernel-devel kernel-default-devel
```

Installs specific tools:

gcc: A compiler for C/C++ programs.

make: Helps build and compile programs.

kernel-devel & kernel-default-devel: Needed to compile kernel modules (important for VirtualBox Guest Additions).

Hello, fdatasync!y the Guest Additions ISO in VirtualBox).

```
sudo mount /dev/cdrom /mnt/cdrom
```

"Mounts" the VirtualBox Guest Additions CD so your system can access it.

```
sudo sh /mnt/cdrom/VBoxLinuxAdditions.run
```

Runs the installer script for VirtualBox Guest Additions, which helps with:

Better screen resolution

Shared folders

Mouse integration

```
sudo reboot
```

Restarts your system so everything takes effect.

Alternative Way to Access Guest Additions (if CD doesn't auto-mount)

```
sudo mount /dev/sr0 /mnt
```

Another way to mount the CD manually.ls /run/media/elshaday(user)

Lists the contents of the media folder to see where the Guest Additions is mounted. ("elshaday" is likely your username.)

```
cd /run/media/$USER/VBox_GAs_*
```

Enters the directory containing the Guest Additions installer (the * means "whatever comes next"

to match the version).

```
sudo sh VBoxLinuxAdditions.run
```

Runs the installer script again.

ImplementingSystemCalls:`fdatasync()`

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <unistd.h>
#include <string.h>

int main() {


int fd = open("example.txt", O_WRONLY | O_CREAT | O_TRUNC, 0644);
if (fd == -1) {


perror("open");


return EXIT_FAILURE;
}


const char *data = "Hello from openSUSE with fdatasync()!\n";


if (write(fd, data, strlen(data)) == -1) {


perror("write");


close(fd);


return EXIT_FAILURE;
}

if (fdatasync(fd) == -1) {

perror("fdatasync");

close(fd);

return EXIT_FAILURE;
}

printf("Data successfully written and flushed to disk.\n");

close(fd);

return EXIT_SUCCESS;
}
```

Conclusion

Working on this project with openSUSE through Oracle VirtualBox has been both a challenging and rewarding experience. It pushed me to step out of my comfort zone and explore a Linux distribution I wasn’t very familiar with before. Setting up openSUSE in a virtual environment helped me understand not just the technical steps involved, but also the logic behind how operating systems work and interact with virtual hardware.

There were moments of frustration—especially when configurations didn’t go as planned—but those struggles made the learning process more meaningful. I’ve gained hands-on experience that I know will be useful in my future career in tech. This project reminded me of the value of patience, curiosity, and continuous learning. I'm walking away from this experience more confident in using virtual machines and more open to diving deeper into Linux system .

References

Linux syscall table (x86_64)

https://filippo.io/linux-syscall-table/

Kernel source code for fdatasync

https://elixir.bootlin.com/linux/latest/source/fs/sync.c#L478

GNU C Library (glibc) implementation

https://sourceware.org/git/?p=glibc.git;a=blob;f=sysdeps/unix/sysv/linux/fdatasync.c