

# Clear.Dental

---



Full re-evaluation of git

# Why reconsider?

# Already moving a lot of the frameworks and foundations



5



6

qmake



cmake



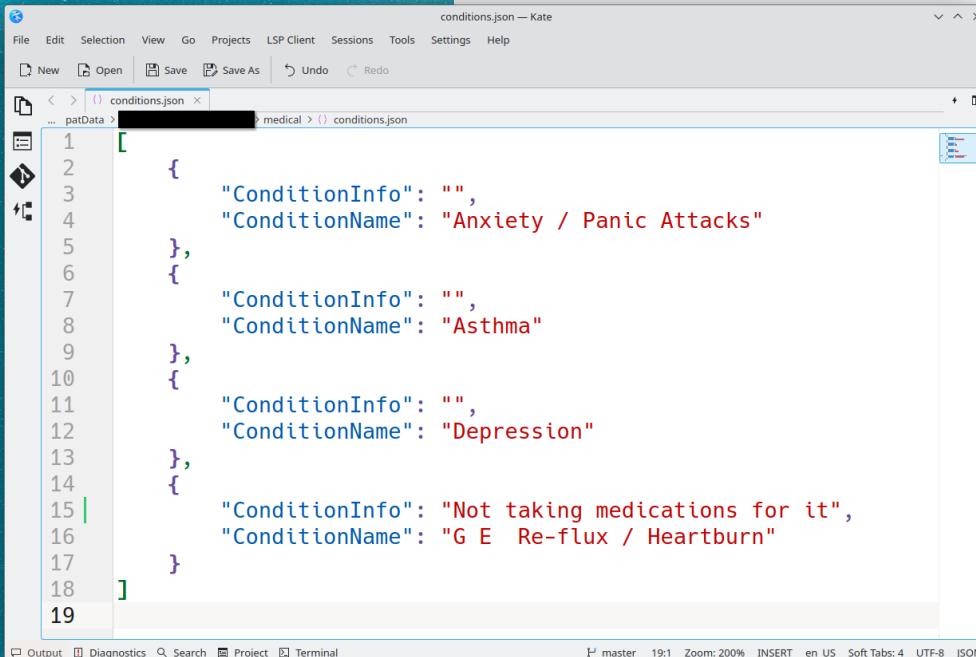
QML Material



What I like  
about using  
the filesystem  
as a database

# Much easier to teach doctors about .ini and .json files vs. SQL

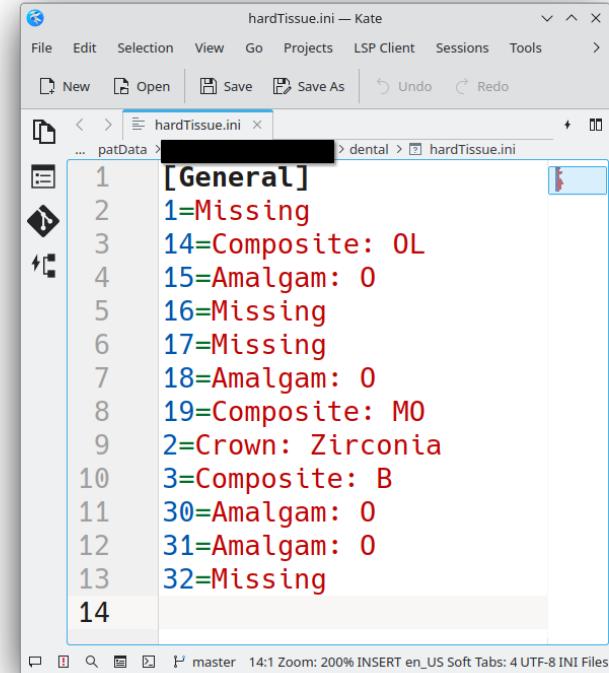
Typical .json file



```
conditions.json — Kate
File Edit Selection View Go Projects LSP Client Sessions Tools Settings Help
New Open Save Save As Undo Redo
conditions.json < ... patData > medical > conditions.json
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
{
  "ConditionInfo": "",
  "ConditionName": "Anxiety / Panic Attacks"
},
{
  "ConditionInfo": "",
  "ConditionName": "Asthma"
},
{
  "ConditionInfo": "",
  "ConditionName": "Depression"
},
{
  "ConditionInfo": "Not taking medications for it",
  "ConditionName": "G E Re-flux / Heartburn"
}
]
```

Output Diagnostics Search Project Terminal master 19:1 Zoom: 200% INSERT en\_US Soft Tabs: 4 UTF-8 JSON

Typical .ini file



```
hardTissue.ini — Kate
File Edit Selection View Go Projects LSP Client Sessions Tools
New Open Save Save As Undo Redo
hardTissue.ini < ... patData > dental > hardTissue.ini
1
2
3
4
5
6
7
8
9
10
11
12
13
14
[General]
1=Missing
14=Composite: OL
15=Amalgam: 0
16=Missing
17=Missing
18=Amalgam: 0
19=Composite: MO
2=Crown: Zirconia
3=Composite: B
30=Amalgam: 0
31=Amalgam: 0
32=Missing
14
```

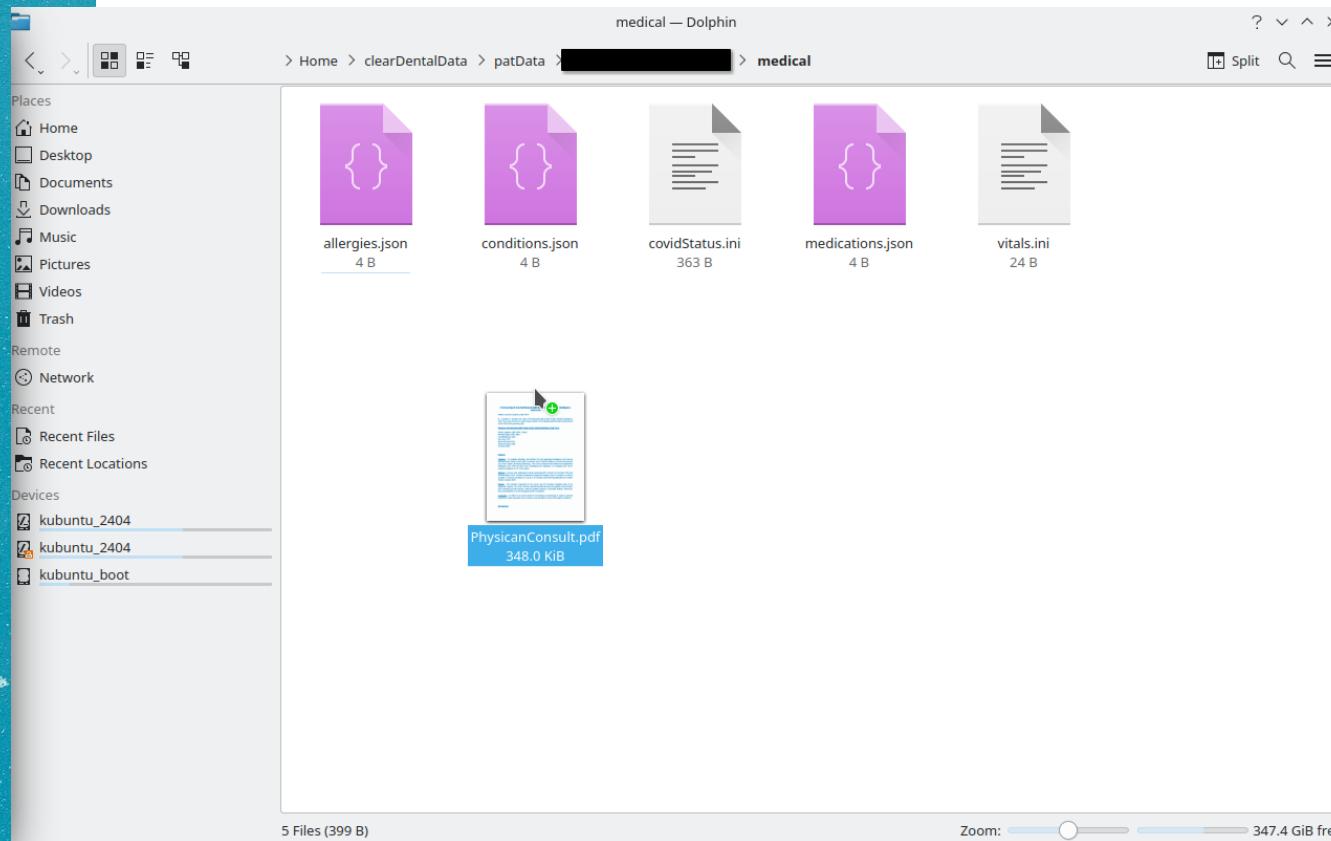
Output Diagnostics Search Project Terminal master 14:1 Zoom: 200% INSERT en\_US Soft Tabs: 4 UTF-8 INI Files

# No need to make a full API, pathnames are the API

```
:
  appointments
    └── Jul-4-2021@10:00 AM.ini -> ../../locInfo/local/appointments/Jul/4/2021/appt1.ini
  caseNotes.json
  dental
    dental
      EoE.ini
      hardTissue.ini
      IoE.ini
      perio.ini
      txPlan.json
    financial
      claims.json
      dentalPlans.ini
      fullBilling.json
      ledger.json
    images
      consents
        Jul
          4
          2021
          └── newPatient.png
      radiographs
        Jul
          4
          2021
          └── BW_Left_Distal.png
          └── BW_Left_Mesial.png
          └── BW_Right_Distal.png
          └── BW_Right_Mesial.png
    medical
      allergies.json
      conditions.json
      covidStatus.ini
      medications.json
      vitals.ini
    messages.json
    newPatInterview.ini
    personal.ini
    reviews.ini
```

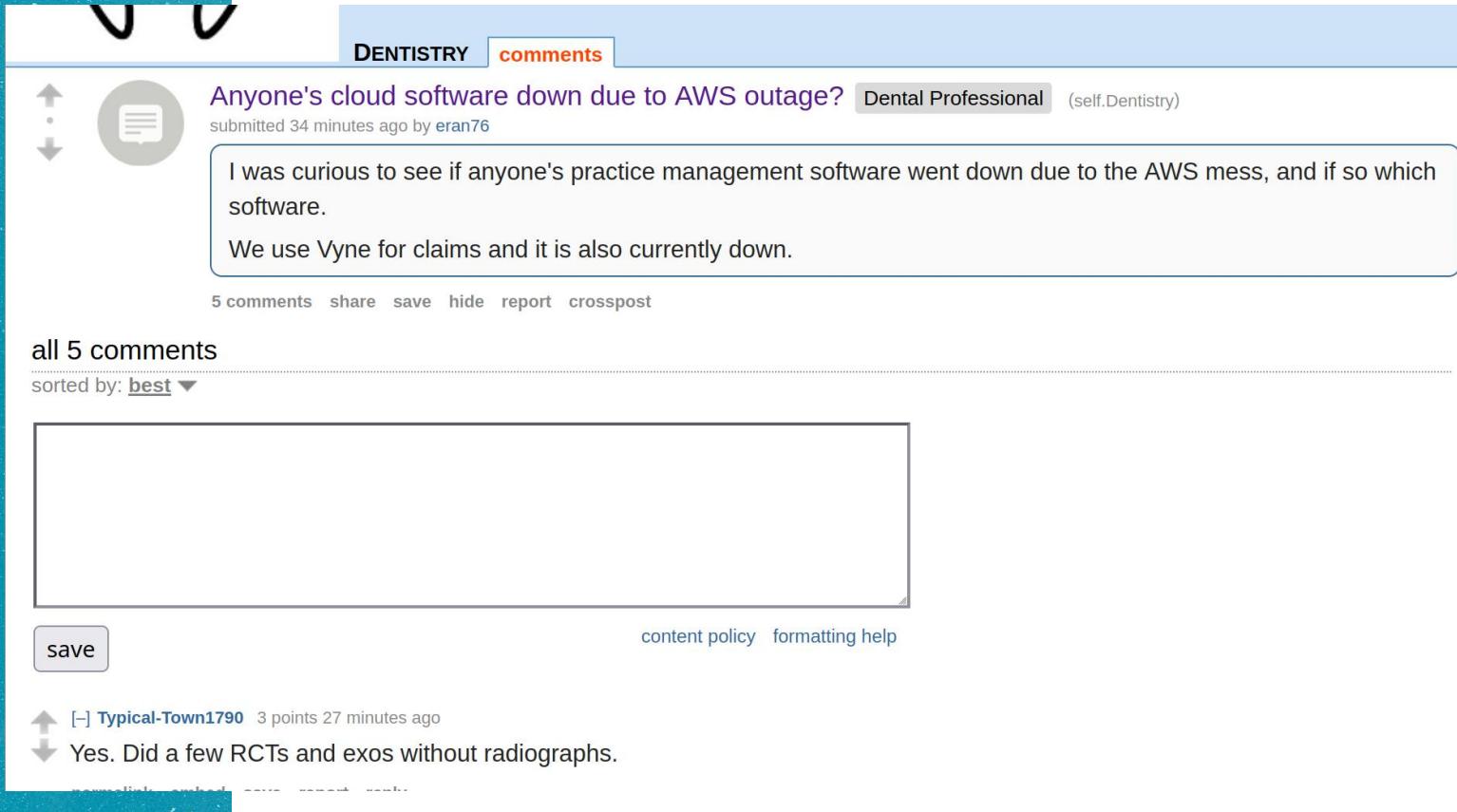
# You can import any kind of file just by dragging and dropping

You can drag/drop .stl files and it would be part of the patient's chart



# You can access data even if the internet / cloud server is down

No more having to worry about AWS outages



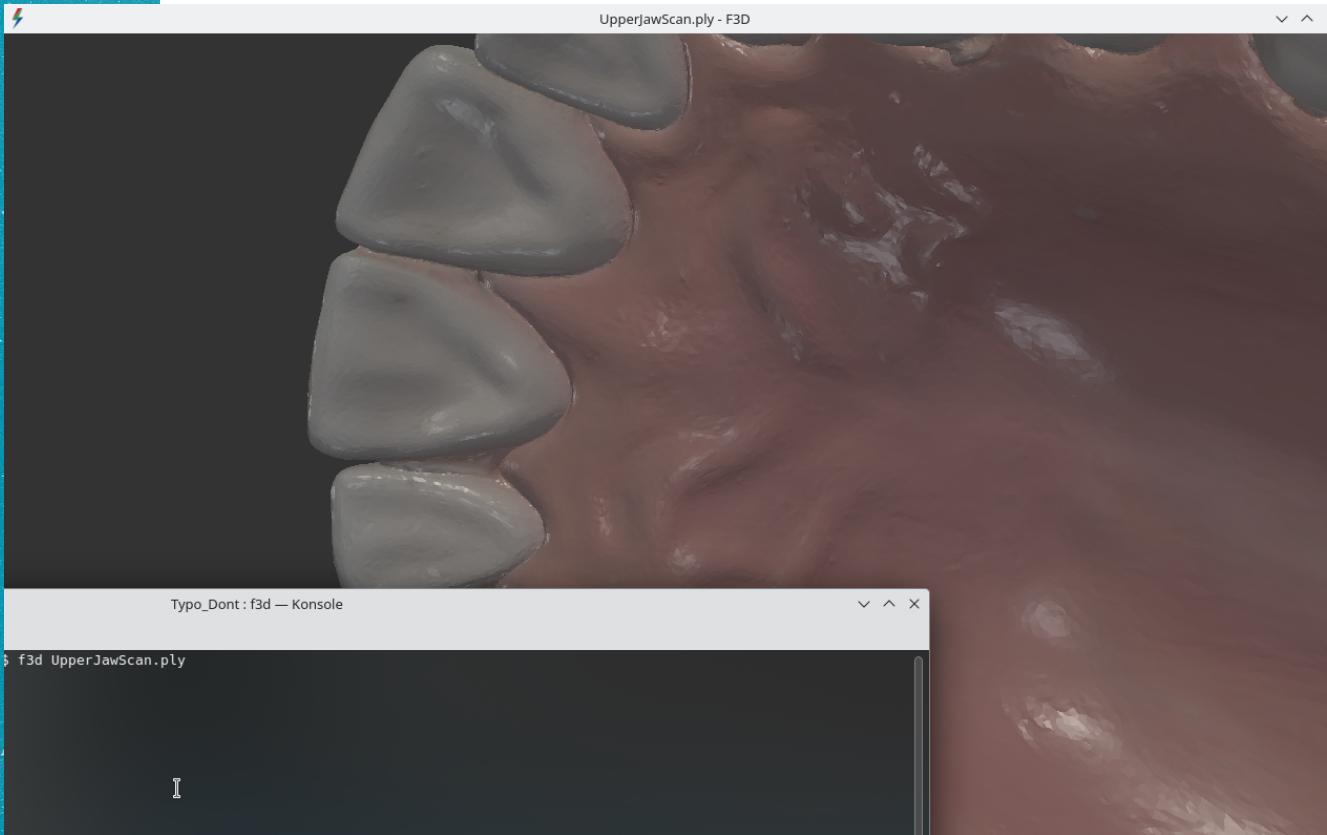
A screenshot of a Reddit post in the **DENTISTRY** subreddit. The post is titled "Anyone's cloud software down due to AWS outage?" and is made by a **Dental Professional** (self.Dentistry). It was submitted 34 minutes ago by **eran76**. The post content is as follows:

I was curious to see if anyone's practice management software went down due to the AWS mess, and if so which software.  
We use Vyne for claims and it is also currently down.

Below the post, there are links for 5 comments, share, save, hide, report, and crosspost. The post is sorted by **best**. A large empty box is present, likely a placeholder for a comment or image. At the bottom, there are buttons for **save**, content policy, and formatting help. A reply from **Typical-Town1790** (3 points 27 minutes ago) states: "Yes. Did a few RCTs and exos without radiographs."

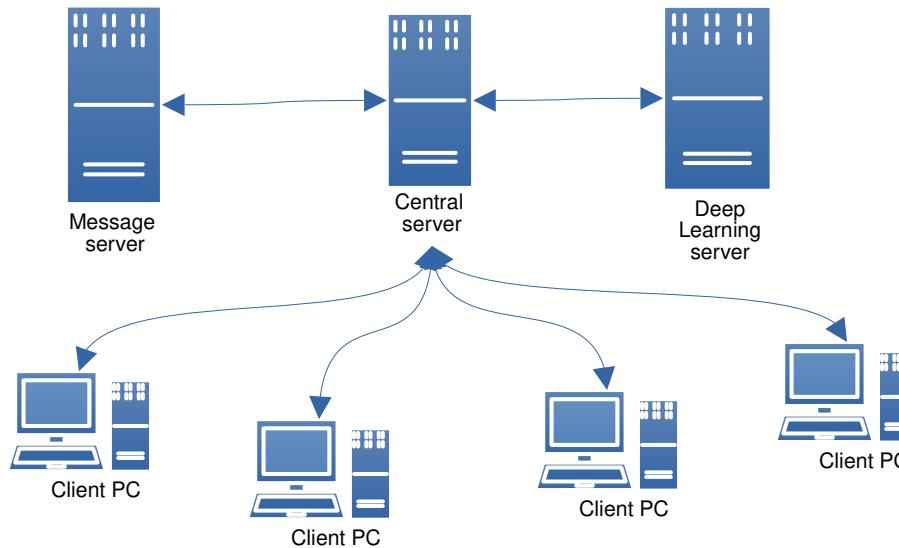
# You can use any built-in app for opening files

Use something like F3D or Blender3D to view the .stl or .ply files



# You can distribute the computational load to other servers

So the other servers don't slow down the clients when reading the data for analysis.

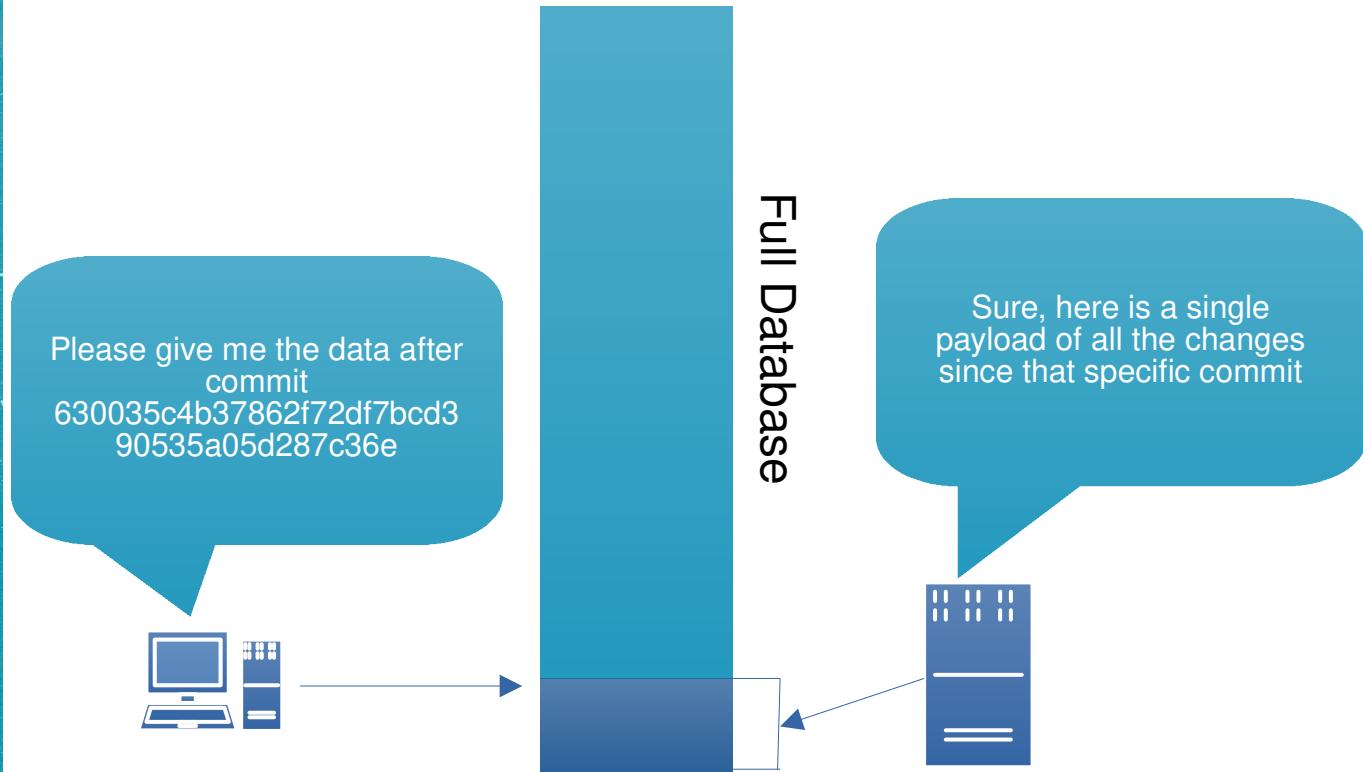


What I like  
about git

# Really easy to setup via command line; easy to start anywhere

```
tshah@durga:~/exampleRepo$ git init -b main
Initialized empty Git repository in /home/tshah/exampleRepo/.git/
tshah@durga:~/exampleRepo$ █
```

# Built-in way to deliver payloads since a specific commit



# You can see who made what changes and when

```
commit 5e673addb30ad035af84930cf9d0cd515d5ca0df (HEAD -> master)
Author: Tej A. Shah, DMD <tshah@zenfamily.dental>
Date:   Tue Nov 11 22:46:32 2025 -0500
```

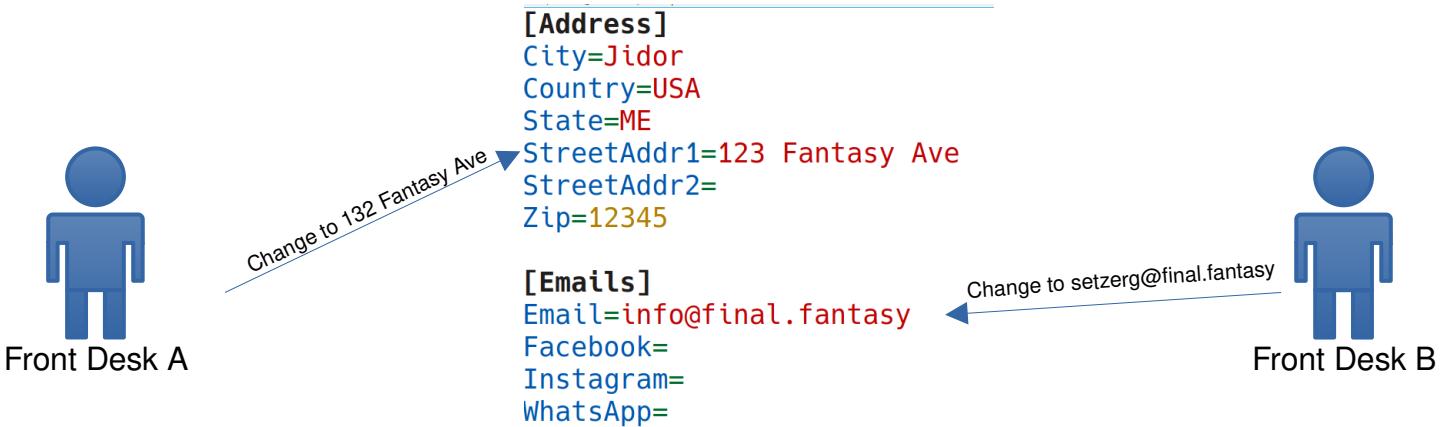
Updated tooth #1 to missing

# Nothing is ever lost; you can find what changes are done years ago

```
        Updated hard tissue Exam for Gabbani, Setzer (Jan-1-1970),  
commit 4af48fb8c17f823454fc8cc95265282081ca5865  
Author: Tej Shah, DMD <tshah@zenfamily.dental>  
Date: Mon Jun 14 18:34:49 2021 -0400  
  
        Updated missing teeth for Gabbani, Setzer (Jan-1-1970)
```

```
zenuser@durga:~/clearDentalData/patData/Gabbani, Setzer (Jan-1-1970)/dental$ git diff 4af48fb8c17f823454fc8cc95265282081ca5865 4af48fb8c17f823454fc8cc95265282081ca5865~  
diff --git a/patData/Gabbani, Setzer (Jan-1-1970)/dental/hardTissue.ini b/patData/Gabbani, Setzer (Jan-1-1970)/dental/hardTissue.ini  
index e1c2eb0f873..dd29182dade 100644  
--- a/patData/Gabbani, Setzer (Jan-1-1970)/dental/hardTissue.ini  
+++ b/patData/Gabbani, Setzer (Jan-1-1970)/dental/hardTissue.ini  
@@ -1,7 +1,4 @@  
[General]  
1=Missing | Decay: 0D  
-16=" | Missing"  
-17=" | Missing"  
2=Composite: 0B  
-32=" | Missing"  
8=
```

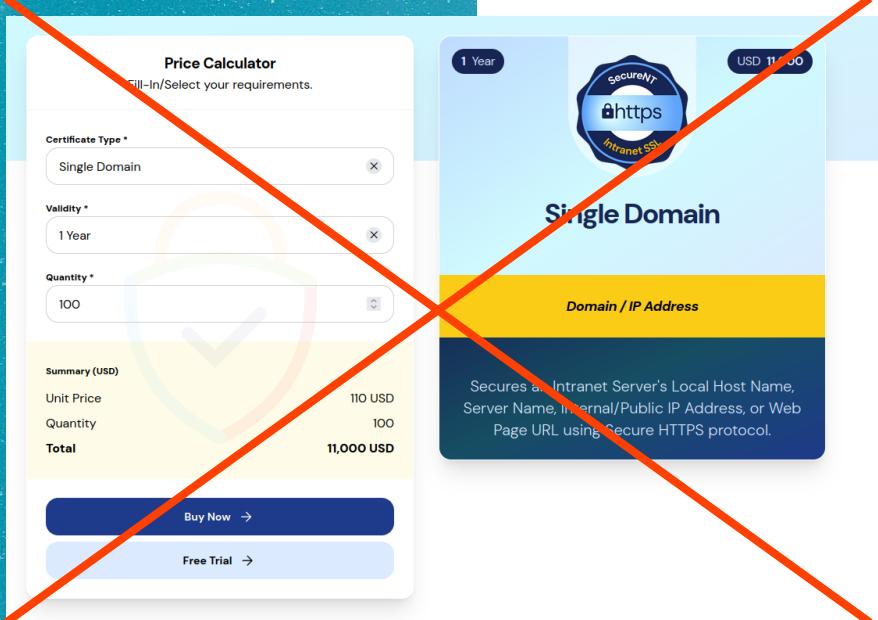
# Not only two staff members can work on the same patient, two staff members can work on the same file



# You can use symbolic links for lookup optimizations (great for the schedule app)

```
enuser@durga:~/clearDentalData/patData/Gabbani, Setzer (Jan-1-1970)/appointments$ ls -lah
total 8.0K
rwxrwxr-x 2 zenuser zenuser 4.0K Nov 11 23:34 .
rwxrwxr-x 7 zenuser zenuser 4.0K Nov 11 23:26 ..
rwxrwxrwx 1 zenuser zenuser 56 Apr 29 2024 'Apr-3-2021@11:00 AM.ini' -> ../../locInfo/local/appointments/Apr/3/2021/appt2.ini
rwxrwxrwx 1 zenuser zenuser 56 Apr 29 2024 'Aug-2-2021@9:00 AM.ini' -> ../../locInfo/local/appointments/Aug/2/2021/appt1.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Aug-23-2023@7:55 AM.ini' -> ../../locInfo/local/appointments/Aug/23/2023/appt1.ini
rwxrwxrwx 1 zenuser zenuser 56 Apr 29 2024 'Dec-9-2021@10:00 AM.ini' -> ../../locInfo/local/appointments/Dec/9/2021/appt2.ini
rwxrwxrwx 1 zenuser zenuser 57 Jul 31 2024 'Jul-11-2024@9:55 AM.ini' -> ../../locInfo/local/appointments/Jul/11/2024/appt4.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Jun-14-2021@8:00 AM.ini' -> ../../locInfo/local/appointments/Jun/14/2021/appt1.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Mar-22-2023@3:00 PM.ini' -> ../../locInfo/local/appointments/Mar/22/2023/appt1.ini
rwxrwxrwx 1 zenuser zenuser 56 May 9 2024 'May-1-2024@4:30 PM.ini' -> ../../locInfo/local/appointments/May/1/2024/appt1.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Oct-12-2023@3:00 PM.ini' -> ../../locInfo/local/appointments/Oct/12/2023/appt2.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Oct-21-2021@9:40 AM.ini' -> ../../locInfo/local/appointments/Oct/21/2021/appt1.ini
rwxrwxrwx 1 zenuser zenuser 56 Apr 29 2024 'Oct-4-2023@8:00 AM.ini' -> ../../locInfo/local/appointments/Oct/4/2023/appt1.ini
rwxrwxrwx 1 zenuser zenuser 57 Apr 29 2024 'Sep-20-2021@10:00 AM.ini' -> ../../locInfo/local/appointments/Sep/20/2021/appt1.ini
enuser@durga:~/clearDentalData/patData/Gabbani, Setzer (Jan-1-1970)/appointments$ █
```

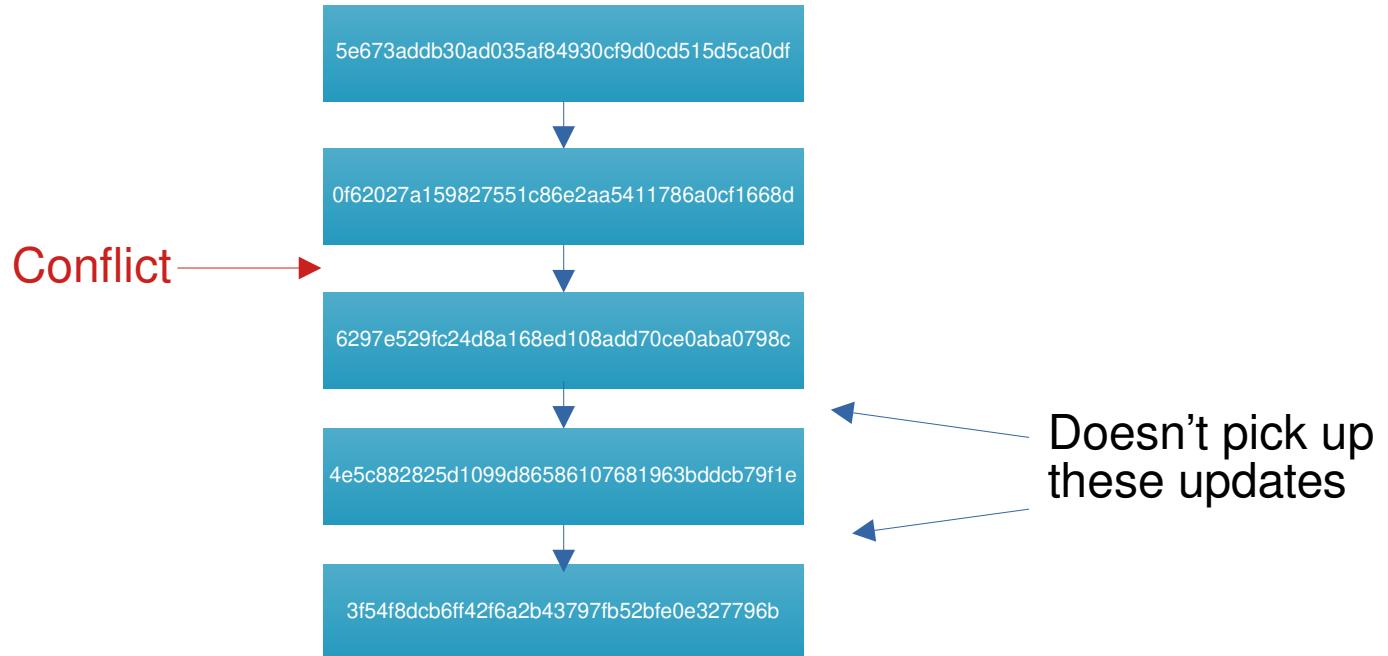
# You can use ssl certs and/or use RSA keys for authentication; great when using encrypted intranet



```
[benuser@durga:~]$ ssh juldhi.local
The authenticity of host 'juldhi.local (192.168.0.130)' can't be established.
ED25519 key fingerprint is SHA256:NY0RGHG5bg9X9GEsuXmYCJwDFCBFSdcUg+XwP3NCjTg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? █
```

What I don't  
like about git

# Git really wants you to manage the conflict, will not accept any more merges until the current merge conflict is resolved





# Git is every easy to use via command line; but not many people in Dental IT know how to use the command line

```
- : bash — Konsole
File Edit View Bookmarks Plugins Settings Help
tshahedurga:~$ cd clearDentalData/
tshahedurga:~/clearDentalData$ git status
On branch master
nothing to commit, working tree clean
tshahedurga:~/clearDentalData$
```

# libgit2 is rather big and much more complicated than using just the git binary



# Android development with git is a mess

libgit2 fails to link (undefined symbol error) if you try to link it to an Android app #7134

[Open](#)



DesiOtaku opened on Sep 19

Using Android NDK 27.2.12479018 and targeting API 35 with arm64\_v8a. I'm using Qt/C++ for the UI but I need to link to libgit2.

I can get libgit2 itself to compile just fine (along with the git2 executable). However, I want to use libgit2 in my Android app.

This line of code in my app's cmake file is fine:

```
add_subdirectory(Libgit2)
```

This, by itself, will compile the libraries and executable but I can't use it in my app.

However, having this near the end of the cmake file for the app:

```
target_link_libraries(appAndroidSimpleTests
    PRIVATE
    Qt6::Quick
    libgit2
)
```

will cause a bunch of `ld.lld: error: undefined symbol errors` like:

```
ld.lld: error: undefined symbol: git_allocator
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_file__clear_rules)
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_file__clear_rules)
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_rule__free)
>>> referenced 1392 more times

ld.lld: error: undefined symbol: git_vector_dispose
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_file__clear_rules)
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_file__clear_rules)
>>> referenced by attr_file.c attr_file.c
>>> libgit2/src/libgit2/CMakeFiles/libgit2.dir/attr_file.c.o:(git_attr_rule__free)
>>> referenced 147 more times
```

with 20+ similar errors. Is there a different way I should be compiling libgit2? When I try to integrate a different library like libssh2, I don't have any issues so I am think maybe there is something special about the libgit2 for Android.

Thanks.

[New issue](#)

## Assignees

No one assigned

## Labels

No labels

## Type

No type

## Projects

No projects

## Milestone

No milestone

## Relationships

None yet

## Development

Code with agent mode

No branches or pull requests

## Participants



# You will always have 2 copies: the .git objects and the working copy

Folder	Size	Contents	Modified
clearDentalData	28.8 GB	77884 items	Today
▶ 50% .git	14.6 GB	145 items	Today
▶ 48% patData	14.0 GB	69913 items	Today
▶ 0% locInfo	166.5 MB	7816 items	Today
▶ 0% docPref	36.9 kB	9 items	2 months

Other  
possible  
alternatives to  
replace git

# Requirements:

- Must be open source (OSI)
- Must handle a decentralized system
- Should work on Linux, Android, iOS
- Should be able to work in an intranet along with an internet node
- Need to be functional parity with git
- Doctors must be able to read / write with minimal training



# Apache Couch DB

## Pros:

- Open Source (Apache License 2.0)
- Uses JSON; which makes it easier to integrate with QML / Javascript
- You get a RESTful API for free
- Has tools for a distributed database
- Allows for local, cloud, and hybrid cloud solutions

## Cons:

- You can't read and edit the database using regular text editor
- You need the "server" to be running to read / write data; if server process crashes, you can't treat patient
- Uses Erlang, which is difficult to port to Android and iOS (for local database)
- A lot of the team members who worked on Couch DB now work at Couchbase which is not Apache Open Source

Accessing data via HTTP <small>[edit]</small>		
Action	Request	Response
Accessing server information	curl http://127.0.0.1:5984/	{ "couchdb": "Welcome", "version": "1.1.0" }
Creating a database named wiki	curl -X PUT http://127.0.0.1:5984/wiki	{"ok": true}



# rsync

## Pros:

- Open Source (GPL v3)
- You still use files as the actual database
- Conflicts are simply overwritten
- Can handle intranet encryption without ssl certificates
- Handles update payloads without having to clone the whole system each time
- Written in C; easy to port

## Cons:

- You have to write your own logging system along with a way to keep the old data (to keep it in same functional parity as git)
- No API or library, you need to use the binary for the functions

A way to mimic the capabilities of Time Machine (macOS)<sup>[27]</sup>

```
$ date $date "+%Y%m%d-%H%M%S" # rsync interprets ":" as separator between host and port (i.e. host:port), so we cannot use BT or SSH here, so we use SSH-SK
$ rsync -p --link-dest=$HOME/Backups/current /path/to/important_files $HOME/Backups/back-$date
$ ln -nfs $HOME/Backups/back-$date $HOME/Backups/current
```

Make a full backup of system root directory<sup>[28]</sup>

```
$ rsync -avAXHS --progress --exclude={"/dev/*","/proc/*","/sys/*","/tmp/*","/run/*","/mnt/*","/media/*","/lost+found"} / /path/to/backup/folder
```



# Syncthing

## Pros:

- Open Source (MPL v2.0)
- You still use files as the actual database
- Automatically / silently handles conflicts
- Can handle intranet encryption without ssl certificates
- Handles update payloads efficiently

## Cons:

- You have to write your own logging system along with a way to keep the old data (to keep it in same functional parity as git)
- No official Android port
- The “API” is just for managing the server, not the data
- When asked in multiple forums, synching users recommended to continue using git



# Jujutsu

## Pros:

- Open Source (Apache v2.0)
- You still use files as the actual database
- Automatically / silently handles conflicts
- Can handle intranet encryption without ssl certificates
- Handles update payloads efficiently
- Keeps track of changes automatically; can “push” them without having to write code to keep track

## Cons:

- It relies on git, and git's own dependencies to work
- No API or library; requires command line
- Clear.Dental is supposed to handle the data distribution transparently; not have the user care too much about the push/pull; therefore jj doesn't add that much



# InterPlanetary File System

## Pros:

- Open Source (MIT v2.0)
- You still use files as the actual database
- Works transparently between intranet and internet servers (works similar to BitTorrent); can work when many of the nodes are down
- Written in Javascript which can integrate with QML directly

## Cons:

- No real security in terms of making sure only valid users can access the data; “pinned” data can be accessed once you know the hash
- Conflicts are not resolved transparently
- More designed to distribute publicly available websites rather than private data

Conclusion:

Going to stick  
with git

# Other things noted:

- Other cloud services don't give doctors any options to manage the conflicts; this is a major issue that the dental community needs to address
- Doing an in-house solution would take far more effort then modifying git

# Thank you

---

