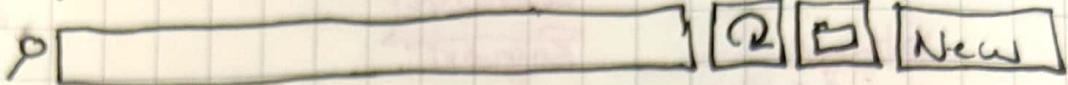
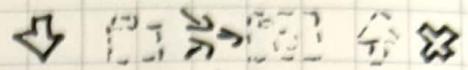
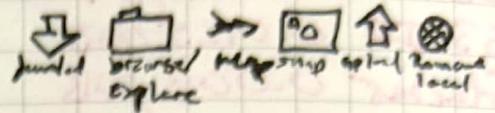
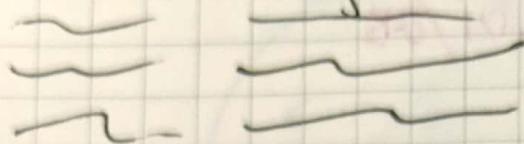


Project List

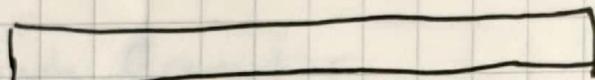


2018-01-01 Sunday Service

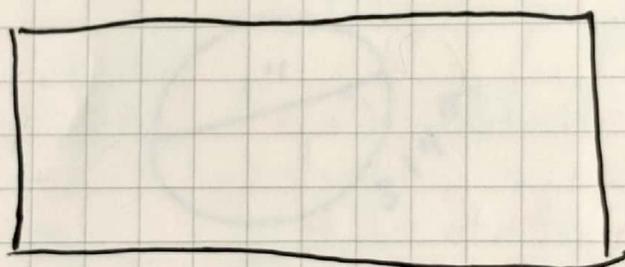


If there are
errors

Snapshot
Summary



Detail

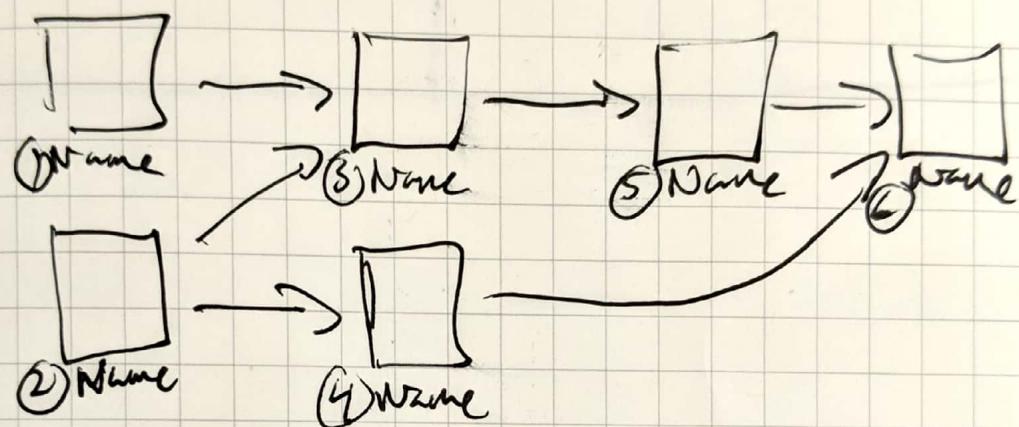
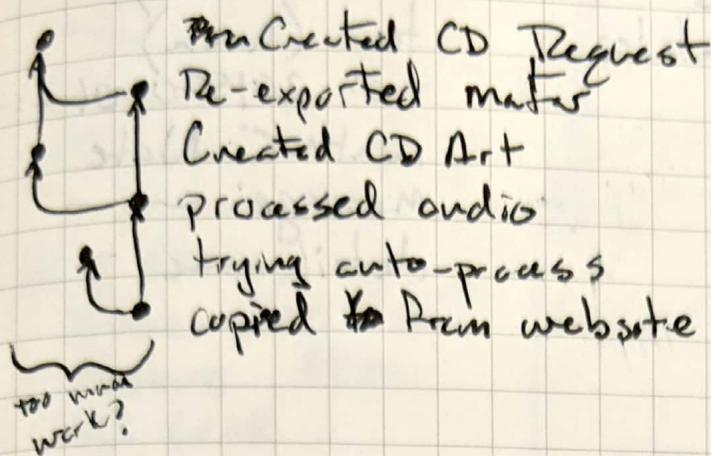


Merge

2018-01-01 Sunday Service /

2018-01-01 Sunday Service ($\frac{1}{2}$) / $\frac{1}{2}$ trans

2018-01-01 Sunday Service ($\frac{1}{2}$) / $\frac{1}{2}$ move



b2 state/snapshots

use sqlite3?

De-Normalized:

host: trogdar

hid: abc123

states:

- from: 000000
- to: 123abc

files:

- name: 2018-01-04 Sunday Service - rec.flac

size: 30,123,456

sha1: 123cba

message: ~

b2name: abc123.1.flac

detail:

b2name: (sha1)(size).file

ts:

b2 tag: 'v1.0.0'

- name: 2018-01-04 Sunday Service - source.flac

size: 30,456,789

sha1: abc123

b2name: abc123.30456789.flac

- from: 123cbc

message: ~

to: ...

detail: ~

authentication?

Normalized:

host: trogdar

hid: abc123

paths:

1: '2018-01-04 Sunday Service - rec.flac'

2: '2018-01-04 Sunday Service - source.flac'

3: 'CDs/180104 CD Request.zip'

versions:

- path: 1

size: 30,123,456

sha1: 123cba

snapshots:

from: 000

'000000'~

- path: 2

size: 30456789

sha1: abc123

snapshots:

'123abc':

base: '000000' (multiple)

versions: 1, 2

'...':

base: '123abc'

versions: ...

snapshots:

[b2 bucket]

Project Folder /

db /

(tmpdir)

abc123.snapshots
(PAW-PC)

efg456.snapshots

file data /

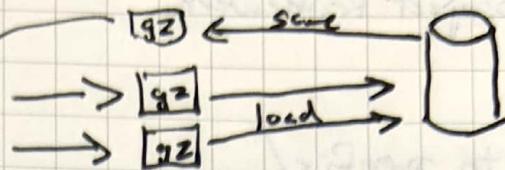
(sha1).(size).file

(sha1).(size).file

project.info

~~Project info~~
title: ~~~~~

[local]



Working Folder /

:snaptree /

:st /

project.info remote.info

hashcache.db

snapshots.db < load

snapshots /

b2 copy it → abc123.snapshots
hash off efg456.snapshots

snap.hash

lockfile

tokens.db

stage /

(sha1).(size).file

2018-01-04 Sunday Service-rec.flac

2018-01-04 Sunday Service-sec..

.stignore

remote.info

account-id: xyz

account-type: b2

prefix:

bucket:

host info

h-id: xyz

b2 account ID:

None

stores: none

st init (name) b2://bucket-name/prefix

st checkent b2://bucket-name/prefix

st update - Get new files
Get db updates

st

st commit - pm "message"

st status --nosync

st sync - Get db updates

st merge (commit)

~~st~~

st tag 'v1.0.0' (--delete)
(can only delete from same host?)

st log

st switch (commit)

st revert (Pole)

~~st~~ log

\$ st init b2://bucket/prefix (--nocheckout) (-i ID) (~~token~~)
Using default account id 123456
auth key: _____ (store in Keyring)
creating project framework
~
~

checkout to prefix/

\$ st checkout b2://bucket/prefix

auth key: ~

checkout to prefix/

2018-01-04 Sunday Service.flac

2018-01-04 Sunday Service - rec.flac

\$ st status

~~Getting updates...~~

New Files:

Changed Files:

\$ st sync

Getting Updates

Goals

- 1) Free up local space by placing files on remote storage
- 2) Collaborate on projects
 - Pull in updates
 - Interface / tool to merge conflicts
- 3) Light code
- 4) Transfer in the background / concurrently
 - (don't get blocked by long running transfers)