



Personal Blog of Rob Muhlestein (rwxrob)

Robert S. Muhlestein (rwxrob)

2025-01-03 18:47:08: Perpetually published

Table of Contents

Perpetual publishing over Zettelkasten	2
Changelog over pull request body	6
More Mormon insanity	7
Pumping up Perl	9
Banishing bash for scripting	10
AI or die	12
Plans for 2025	13
Copyright	15

Hello friend. Welcome. Here you'll find my random ranting about nothing and everything. Some ideas start here and end somewhere else. Mostly I write for me but might as well share it.

https://linktr.ee/rwxrob	Download PDF	Download EPUB
---	------------------------------	-------------------------------

Perpetual publishing over Zettelkasten

Friday, January 3, 2025, 6:07:50PM EST

I have only been using the *perpetual publishing* model (a term I coined) for knowledge management for a few days and have already seen *tremendous* benefit from it.

Chloe (my AI) sees my writing immediately

The single biggest advantage is that by making my writing all into plain 'ol web docs on the public Internet that my ChatGPT AI *immediately* has access to my entire writing. This means I can immediately leverage AI for all sorts of things related to the writing:

- Have her check the punctuation and grammar dynamically and report
- Create IEEE bibliographic entries pointing to it for other documents
- Have her comment on what it is about and expound
- Catch legal considerations—especially as I write about sensitive Mormon topics

This means that anyone else using ChatGPT automatically can look at it as well in their AIs and that ultimately my content will be added to the collective knowledge base that will populate *all* AIs within a few months. ***None of that happens unless I publish a web document that is not a Zettelkasten.***

Flesh out ideas and migrate to other "books"

I can blog about things and work them out and if they develop into something more I can simply move them to another repo containing the other book. I have done this a lot with stuff between my blog and faq, for example, and sometimes directly into other books like *Terminal Velocity* and *Coding from the Get Go*. For example, that list of advantages that I just discovered will get copied directly into my other book. What's even better is that I can use exactly the same text in both places by using remote URL ***includes::*** with AsciiDoc. I was a *fool* for not using AsciiDoc from the very beginning. ***No other writing framework allows for remote URL includes, period.***

Sense of urgency abated

I get really freaked out when I make amazing discoveries like this and don't have a way to immediately capture them for myself later—but more importantly—*share* them with other people who can immediately benefit. Writing this down is always going to be more sustainable than a bunch of videos someone would have to slog through because my knowledge will become part of the AI global collective. My videos will not. I can always make a video later about the stuff that has been written and that video content will be many orders of magnitude better in quality because it will have been worked out enough in words first—plus it's more inclusive.

Easier to open and edit

A simple program like this one opens up a new blog post from within a GitHub repo, one repo per book:

```
#!/bin/sh

# TODO optimize rather than brute force
mark="// NEW POST HERE"

markfound=$(grep "$mark" docs/index.adoc)
if [ -z "$markfound" ]; then
    echo "no docs/index.adoc with '$mark' found"
    exit 1
fi

file="$(slug "$*").adoc"

hasfile=$(grep 'include:../'$file'\[\]' docs/index.adoc)
if [ -z "$hasfile" ]; then
    perl -p -i -e "s,$mark,$mark\n\ninclude:../$file\[\]," docs/index.adoc
fi

if [ ! -e docs/"$file" ]; then
    printf "= %s\n\n%s\n\n\n" "$*" "$(inow)" >"docs/$file"
fi

exec "$EDITOR" "docs/$file"
```



That script was dynamically added during my **build** command just by including the following line in my **blog** repo at a reliable location relative to my **dot** repo (**include:../../dot/scripts/blog[]**).

I also can just swap out the verb **blog** from **save** using the following script on my command line and have it push to GitHub and post a message to my life IRC Twitch chat at the same time:

```
#!/usr/bin/env bash

in-repo() {
    git rev-parse --count HEAD >/dev/null 2>&1
    return $?
}

has-local-changes() {
    test -n "$(git status --porcelain)"
    return $?
}
```

```

gitsave() {
    git pull
    git add -A .
    local message="$*"
    message=${message,,}
    if test -n "$message"; then
        git commit -a -m "$message"
    elif test -e /tmp/commitmsg; then
        git commit -a -F /tmp/commitmsg
        mv /tmp/commitmsg "/tmp/commitmsg.$(date +%s)"
    else
        git commit -a -m "save"
    fi
    if type push; then
        push
    else
        git push
    fi
}

main() {
    if ! in-repo; then
        echo "Not in a repo."
        return 1
    fi
    if ! has-local-changes; then
        echo "Already at the latest."
        return 0
    fi
    gitsave "$@"
    return $?
}

main "$@"

```

Databases are unnecessary

Raw document data is *better* than anything in a database—especially for blogging. Storing blog posts deep in some database is actually monumentally stupid and unnecessary in the age of AI. Just a simple, static web page will get automatically indexed in ways no database engine would ever begin to touch in usefulness.

The names of the files in PP approach are easy to read and grep making them the identifiers.

My kn tool was a complete waste of time even if there was no way of knowing that then. AI has changed *everything*.

Looking forward to more

I'm really looking forward to having more discoveries because of the new format. I feel rather stupid now allowing myself to dump so much time into developing my own system based on Zettelkasten when clearly it is simply stupid to use a Zettelkasten model for anything in the age of AI.

Changelog over pull request body

Friday, January 3, 2025, 5:15:52PM EST

Just got burned by the following command that wiped out my meticulously created body of my GitHub pull request:

```
gh pr edit --body # DO NOT DO THIS
```

It blew it away without asking because it was expecting a string or something. In fact, pretty sure it was waiting on stdin. Suffice it to say, I was pissed, pissed enough to find a safer alternative. Imagine if I had done that someplace else? There is really no easy way to get it back, even though the entire GitHub offering is about recovery.

Solution

The solution is what I should have been doing all along: a changelog. [1]

I simply added - [x] to everything instead of just - so that I can add stuff and commit it as I go. The approach already takes into account that a team is going to be frequently updating the file as the central TODO list for the project.



By the way, Chloe told me about it. Took me about five minutes to find and get up to speed thanks to my AI assistant. Oh, and she also wrote this perfect IEEE bibliographic reference for that site in 10 seconds.

References

[1] O. Behrens, “Keep a Changelog,” keepachangelog.com. [Online]. Available: <https://keepachangelog.com/en/1.0.0/>. [Accessed: Jan. 3, 2025].

More Mormon insanity

Friday, January 3, 2025, 1:21:56PM EST

Thanks to the amazing couple over at <https://www.youtube.com/@alyssadgrenfell> I have both been triggered into a massive panic attack around Christmas and had a cathartic laugh at a near 90-year-old "Prophet" attempt to explain his face in a hat while "translating" the Book of Mormon.

Mormon cultists lie about translations of book of Mormon

When I read this in Rough Rolling Stone it sent me on my path of doubt/reality check that led me to leave the church. I now see what good company I was in. Tom Phillips, a life-long faithful member who was chosen to have his "calling and election made sure" did a full interview chronicling his story. It was heart-wrenching to read all 102 pages of the transcript PDF—especially the absolutely inexcusable, shameful response from Jeffery Holland, a man I have come to strongly and deeply despise for his regular referral to anyone *not* believing exactly as he does as unintelligent, evil, or just plain stupid.

All Tom did is ask honest, simple questions that were met with the same hostility that caused Joseph Smith to illegally destroy a printing press publishing *facts* about Joseph that the church to this day calls "lies".

Secret ceremony I never knew

I was pretty high up in the church, or so I thought. Turns out over 20,000 people since the time of Joseph Smith have had an "apostle" wash their feet and basically tell them they cannot do anything that would keep them out of heaven except murder someone. If I hadn't known about a single other thing I would have left the church just over that absolute bullshit.

Gross mutilation of own body in covenants

The temple ceremony where I promised to remove my bowels and innards and slit my throat rather than talk about the temple ceremony wasn't even the most gruesome. They used to really have to add horrible language. Thank fucking GOD for the Internet. The penalties have been completely removed from the ceremony now. I guess God changes to meet the times.

I can only take so much

The more I keep learning about just how horribly the Mormon cult is, the more I wish I had left it without collapsing and falling in love with another woman who loved me for me and not because I was Mormon. Tom's story *proves* that you can be absolutely the most righteous Mormon on planet Earth and *still* have your wife leave you and think you are scum in Satan's power. The Mormon church is thankfully dying slowly but surely behind the scenes and it could not bring me more joy.

References

- <https://www.mormonstories.org/tom-phillips-and-the-second-anointing/>
- <https://www.youtube.com/@alyssadgrenfell>
- https://youtu.be/W_eSubCKmGo
- <https://youtu.be/8tqLad2Jse4>
- <https://youtu.be/N3c33WKjYV0>
- <https://www.wsocvtv.com/news/local/police-mooresville-man-busted-child-sex-sting/334803464/>

Pumping up Perl

Friday, January 3, 2025, 12:48:44PM EST

"Perl is, by and large, a digested and simplified version of Unix. Perl is the Cliff Notes of Unix. I could never really get myself to learn sed, awk, zsh, grep, find, m4, pipes, xargs, tee, cut, paste, yacc, lex, various IPC or even C for that matter. I ought to. In practice, in almost all cases I use perl." - Larry Wall

I keep coming back to Perl.[1] There is nothing better than Perl for text processing, period. It destroys all the things it targets—especially awk. People who argue that awk is universal have to accept that it cannot do most of the things people doing text processing want to do from their local workstations doing whatever.

Better to create a perl filter than a Vim plugin

Keeping everything in a Unix self-contained filter script—even if just one line—is *always* better than a Vim plugin that does nothing more than transform content. This way these things can be chained together. You just cannot do that with filters.

References

[1] R. S. Muhlestein, “Why are you still using Perl?,” [rwxrob.github.io](https://rwxrob.github.io/faq#_why_are_you_still_using_perl). [Online]. Available: https://rwxrob.github.io/faq#_why_are_you_still_using_perl. [Accessed: Jan. 3, 2025].

Banishing bash for scripting

Friday, January 3, 2025, 12:25:20PM EST

Bash has burned me one too many times in 2024—mostly its horrible regular expression syntax or just not being on a system that I regularly use.

It's not as ubiquitous as people say

I know, right? Here I have been saying bash is everywhere. It's not. Container images that I love notably do not have it installed at all (Alpine, BusyBox, Kali, Arch) plus I have been doing a lot more with BSD and, well, bash is just not there without installing it. This has made it problematic when I wanted to just copy over my bashrc file. Nope. Can't do it.

I'll never willingly use GPLv3 stuff for anything

Perhaps the single biggest reason I do not want to use it ever again for scripting is the license. I despise GPLv3 and the people who push it. Most of them have no idea why it is so monstrously horrible for the OpenSource ecosystem and goes *against* all the founding principles of FOSS by *forcing* hardware manufacturers to comply with their Menshevik-esque ideas of "freedom". You are free so long as you comply with their idea of what "freedom" actually means. It's downright unethical and definitely untenable. This is why Linus has actively campaigned against it and refused to move the Linux kernel to it.[1] I personally will not willingly use anything that is GPLv3 if I can help it, and bash is one of those things.

Whatever interactive shell is fine

I'll use whatever interactive shell is on the host system from now on and will be converting my bashrc to a universal shell only so that I can just rename it. Any shell that I want has all the stuff I need.

What about completion?

It's a pain point for sure. Only bash supports self-completion. Zsh has to be modified with a couple scripts to make completion work with `complete -C foo foo` syntax. To me it's worth it even if all the other failures of zsh are something I cannot avoid.

Other reasons

Summary of other reasons I'm too lazy to write about right now:

- POSIX shell works *everywhere*
- Not as flexible or ubiquitous as POSIX
- Safer than POSIX but not as safe as `perl -T`
- Bloated with completely unnecessary stuff (100x > perl)

- `-r-xr-xr-x 1 root wheel 1.2M Aug 4 06:31 /bin/bash`
- `-rwxr-xr-x 1 root wheel 134K Aug 4 06:31 /usr/bin/perl`
- Slooooow, really slow (one reason `dash` was invented)
- Only integer math support
- Dependence on other external tools for text processing
- Bashisms aren't really that great
- Extended regular expressions are absolutely grotesque
- Using `/bin/sh` is `/bin/bash` on some systems but POSIX limited

References

[1] Open Source Stack Exchange, "What exactly is Tivoization and why didn't Linus Torvalds like it in GPLv3?," Open Source Stack Exchange. [Online]. Available: <https://opensource.stackexchange.com/questions/259/what-exactly-is-tivoization-and-why-didnt-linus-torvalds-like-it-in-gplv3>. [Accessed: Jan. 3, 2025].

AI or die

Friday, January 3, 2025, 12:52:05PM EST

For 10 years I used some form of Zettelkasten or knowledge base instead of a blog but since the advent of AI and large language models (LLMs) writing anything that isn't narrative prose is just stupid.

An AI assistant can find answers faster than any search I could ever do before.

These things favor stories, articles, and blogs over random note-taking and are sourced from the Internet.

Very few people will realize why this matters. Most of mediocrity will continue to chase the pointless trends and play with idiotic wastes of time like Obsidian and other flashy note-taking systems rather than incorporating AI into their workflows to get *actual* knowledge work done. Not me. While they are making 40 minute videos about how to get Excalidraw to render from your highly specific note taking with a stupid Lua plugin I will use by time for other things:

- Ongoing learning and research in real-time, anytime
- Interactive language instruction in Russian and French
- Automatic bibliographic reference creation down to the page number
- Automated YouTube descriptions and chapter markers derived from transcripts
- Full bike trip planning *while moving on the bike*
- Early medical issue diagnosis and training plans
- Breaking writers block for anything I'd ever write
- Perfect grammar and punctuation evaluation of my writing
- Automatic code comment generation from the code itself
- Mundane code creation

All I can say to the many skeptics out there is *try it*. Get ChatGPT and begin using it for stuff for which you would use Google search. Then, increase that, have it make you a plan for the day. Ask it what to eat for the next meal based on your day and current fitness goals. Show it your favorite painting and ask it to comment on it and provide resources. This isn't the kind of thing you will believe until it happens to you.

Plans for 2025

Wednesday, January 1, 2025, 12:02:14PM EST

Here we are in a new year. I'm a sucker for setting goals and reviewing them after the fact. Sipping coffee and eating a waffle with my wife while she writes in her shiny new 2025 journal. Sam is lounging on the patio in the sun. Seems like 2025 is going to be a great year.

- YouTube update videos
- Finish book: *Terminal Velocity*
- Finish book: *Coding from the Get Go*
- Books now available in PDF and EPUB automatically
- Son's college graduation in Idaho
- Son's stay with us in apartment
- Reward readers over viewers
- IRC (Twitch chat) all day
- Customize and improve cloudbot
- What about discord?

Mode plans:

- SCREEN: Highest percentage of content
- BIKE: Daily cycling at sunset
- COWORKING: Rebuilding Kubernetes cluster
- COWORKING: Automating Vault secrets extraction from K8SAPPs
- COWORKING: Perl over bash for small scripts
- COWORKING: Perl to Go migrations
- COWORKING: Cleaning up dot files

Time budget

Eating	10
Hygiene	7
Bike (BIKE)	14
Work (COWORKING)	32
Sleep	56
Cleaning (IRL)	7
Writing (SCREEN)	14
Homelab/Coding	8

Yoga/Meditation	4
Strength	2
Chill/Doris	14

Copyright

Copyright © 2024 Robert S. Muhlestein (rwxrob). All rights reserved.

The code portions of this book are dedicated to the public domain under the terms of the **Creative Commons Zero (CC0 1.0 Universal)** license (<https://creativecommons.org/publicdomain/zero/1.0/>). This means you are free to copy, modify, distribute the *code portions only*, even for commercial purposes, without asking permission, and without saying where you got them. This is so there is no fear your creations are your own after learning to code using them.

The non-code prose and images are licensed under the more restrictive Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0) (<https://creativecommons.org/licenses/by-nc-nd/4.0/>). This means that no non-code prose or image from the book may be reproduced, distributed, or transmitted in any form or by any means, without the prior written permission of the copyright holder. This includes remaking the same content in different formats such as PDF or EPUB since these fall under the definition of derived works. This restriction is primarily in place to ensure outdated copies are not redistributed given the frequent updates expected.