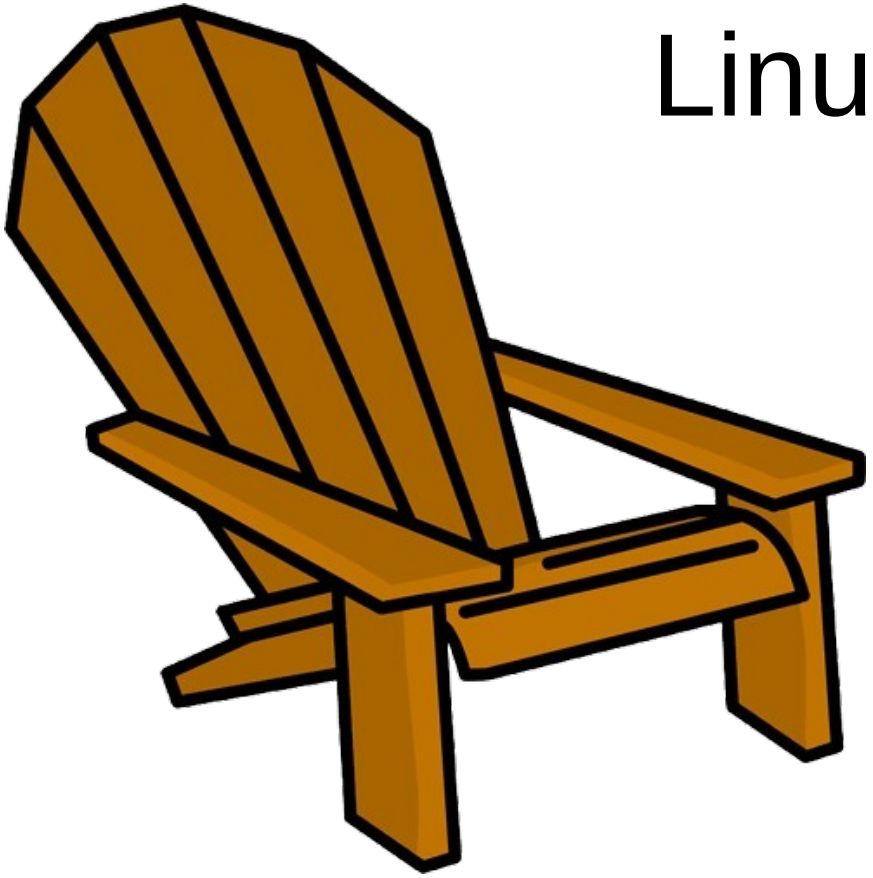
Linux in the Garden



Images removed for privacy reasons

My use case, and goals

- New house, new garden
- Food garden
- Ornamental garden
- Hummingbirds
- Zone 5a
- Deck oven
- Cedar is crap
- Privacy

“Privacy” vs “full-frontal botanical nudity”

Website – GeoOttawa

- https://maps.ottawa.ca/geoottawa/
- Geographic Information System (GIS)
- esri arcGIS (may, or may not be Linux).
- Useful information without payment.
- Lidar and photogrammetry data, for a fee.

GeoOttawa

- Boundary dimensions
- Building locations
- Background aerial views
  - 1928 to present
- 1965 (built 1967)
- 2002 (track old tree roots)

geoottawa

- Dimension measurement
- DIY survey… all fine

ROW land

- Municipal Right Of Way
- Strip of land perpendicular to road
- Homeowner obliged to maintain
- Allowed to plant (restrictions)
- Municipality may
  - dig whenever they need
  - do whatever they want
  - won’t restore when finished
  - charge for extra dig costs

ROW depth from geoottawa

- ROW includes road
- Measure lot to lot, and remove road.
- (18.2 – 8.9)/2 = 4.65m = 15’3” of front garden is municipal ROW
- 96’ lot width at road = 1464 ft²

ROW land

- At 1464 sqft, as much as some new build houses!
- One advantage of ‘bad pie’ and corner lots, if you use it.
- Neighbours seem to think the ROW is about 6’, so I keep quiet!

Hedges *could* be removed as they are on ROW land.

My use case – software goals

- Plan food garden.
- Plan ornamental garden.
- Model and track garden.
  - Learn and improve next year
  - Dynamic db model.

“FOSS projects” meet “Tug’s opinion”

- I’m looking for…
  - data projects, to model data.
  - API projects, to access data.
  - UI projects, to enter and present data.
- If databases resources exist, use them.
- Too many *all-in-one* combo projects
- I will build my own DB
- So there will be no API or UI projects

My database goals

- I like Relational databases.
  - Map to OO model
  - SQL perfectly functional, and portable
  - A db is for storing data, not implementing functionality
- Evolution, not creation.
- Don’t model everything, only if relevant to *my* garden.
- DIY db + using open source database tools.

Open source projects I have rejected

Completely necessary image, generated by AI, for illustrative purposes.

Database – trefle.io

- trefle.io
  - https://github.com/treflehq
  - French origin application
  - "botanical API for plants species, allowing you to build the next gardening apps and farming robots."
  - Trefle is dead
  - Trefle never worked
  - Acknowledges it sources its data from 3rd parties…

Database – Garden_Database

- https://github.com/callahnr/Garden_Database
- Just a database model.
- Another "not quite" idea.
  - Location (field name, row in field, column in field), so only rectangular fields,
  - no dimensions.
- A weekend school project for the Relational Database course, SQL lecture?

Db tool - beekeeper

- Uses nodejs. Electon based.
- requires HUGE build infrastructure - chromium and yarn, GBs of shite... basically a distro worth of source code
- Trying a bin version.
  - Limited, as it is a freemium open source project. Extras missing include
    - backup and restore
    - more than 2 table features
    - import from file
- demo version? I gave up on it.

Projects that had some useful stuff

Completely necessary image, generated by AI, for illustrative purposes.

Application - garden_database

- https://github.com/papakonnekt/garden_database
- Django web application
  - comprehensive horticultural data.
  - includes REST/GraphQL APIs.
  - requires docker.
  - “integrates with AI tools via prompts for efficient data generation.”
- Includes a data model
  - worth looking at – most software reuse is in the mind, not the file.
  - https://github.com/papakonnekt/garden_database/blob/main/docs/DATA_MODEL.md

Application - openjardin

- Openjardin
  - https://sourceforge.net/projects/openjardin/
  - French, and limited documentation.
  - Needed porting from qt5 to qt6.
  - Flaky GUI
  - Aimed at permaculture. Food only. Beds of crops
  - No dimensions, so “topological gardens”
  - Seems to be dead
  - Gantt charts
    - like real life - "done", "not done", "skipped"
    - No garden sharing of tasks in our garden, so not applicable
  - Worth a fork, if that’s what you’re looking for.

Website – powo.science.kew.org

- very wordy, but not the words I want. I need
  - bloom time,
  - hardiness,
  - flower colour, etc
- Does have native / invasive / introduced status
- Lists of sources. Great for academic research.
- Example – bee balm
  - https://powo.science.kew.org/taxon/urn:lsid:ipni.org:names:452034-1

Open source projects I have accepted

Completely necessary image, generated by AI, for illustrative purposes.

Application – PlantNet

- https://github.com/plantnet, stylised as Pl@ntNet
- Useful phone app to identify plants from photos. Often good, sometimes useless.
  - a citizen science project
  - plant identification API
  - research tools
- Europe based, French and English.
- Free for testing and non-commercial use, EUR 1000/year for commercial use.
- My goals are to select and track plants, not to identify them
- Future - everyone is a Victorian landowner, with robot gardeners.
- App based on data from Kew Gardens' POWO, and TDWG's WGSRPD

Website – garden.org

- https://garden.org/plants/
- Gives the data I want.
- not an API, but a website.
  - maybe there’s an accessible API.
  - I'm happy to use this, as is, to research data.
- Needs a free account
- https://garden.org/plants/view/77839/Bee-Balm-Monarda-didyma/

Website – pfaf.org

- https://pfaf.org/user/default.aspx
- Good source of plant research
- Focus on edible, medicinal, temperate perennials.
- Seems constantly in financial stress
- Projects out there to suck the db
- Example
  - https://pfaf.org/user/Plant.aspx?LatinName=Monarda+didyma

DB design tool

- Will build a new database
- Need to
  - Modify tables to add features as required
  - Input data as needed
  - Populate from multiple sources of data.
- Use SQLite, as it is sufficient and simple.
  - If not, migrating to other SQL is easy.

Db tool – sqlitebrowser

- My usual sqlite tool
- Is more than a browser, also allows for database models to be created
- Seems as usable as beekeeper.
- Can export to DDL
  - Easy to port to other SQL environment.
  - Good candidate

Db tool – sqlitestudio

- New to me
- Similar to sqlitebrowser.
- Has views, which sqlitebrowser lacks.
- IT HAS VIEWS!!!
- Ideal for evolving, populating, and viewing the db.
- *Demo for how to evolve, populate and view the db*

Interfacing to sqlite3 db

- Present GUI interfaces?
- Plenty of C++ wrapper libraries
  - I already have my own direct to dev-db/sqlite3
  - libodb-sqlite https://codesynthesis.com/products/odb/
  - sqlite_orm https://github.com/fnc12/sqlite_orm
  - wxsqlite3 http://wxcode.sourceforge.net/components/wxsqlite3
  - vsqlite++ http://evilissimo.fedorapeople.org/releases/vsqlite--/
- Plenty of wrappers for many languages you might prefer

  Guile, Haskell, Ocaml, Perl, Python, Ruby

Happy Tux in garden

Completely necessary image, generated by AI, for illustrative purposes.


"That's all Folks!"