Recent Software Projects & Updates

By Alex Taylor

http://www.altsan.org

https://github.com/altsan

About Me

- Former:
  - OS/2 LAN & workstation support technician
  - Solaris system administrator
  - Lotus Domino administrator
  - English teacher in Japan
- Current:
  - Independent (mostly OS/2) software developer and UI/UX consultant
  - Aspiring font designer
  - All-around gadfly

My Software

Home page: http://www.altsan.org/
Github: https://github.com/altsan/

I tend to write programs that I want/need for myself.

Since I finished my postgraduate studies I have gone back to some long-idle projects/aspirations.

Some areas of interest:

- User interface & user experience design
- Internationalization & text encoding (especially CJK)
- Fonts, typography & printing

Contributed Work

NewView
ConfigApps
RPMs

NewView

- Reorganized NewView source repository: http://trac.netlabs.org/newview
  - Fixed drive letters/paths no longer required to build
  - Rewrote build instructions
  - Provided prebuilt CompLib.DLL
- Updated & released NewView 2.19.5, 2.19.6:
  - Copying multiline text hopefully no longer adds garbage to the end.
  - Fixes for DBCS systems (Japanese, Korean, Chinese) – text wrap, cursor navigation, highlighting now working.

ConfigApps

- Aaron Lawrence's original tool for configuring default Internet applications.
- Added sources to NewView repository.
- Added multilingual support (external message files) using the same method as NewView & AE.
- Minor cosmetic fixes.

New RPM Packages

- Repackaged some existing software as RPM (*rsync*, *termcap*, *TeTeX*).
- Ported FreeWnn to OS/2.
  - FreeWnn is an open source input (IME) system for CJK (Chinese/Japanese/Korean) text.
  - Core and development RPMs are available for: Japanese (*FreeWnn*), Simplified & Traditional Chinese (*FreeWnn-Chinese*) and Korean (*FreeWnn-Korean*).
  - This is only an IME engine/back-end. It does not include a client (i.e. GUI support), so by itself it is not very useful… …yet. ☺

IMERJ 2.0

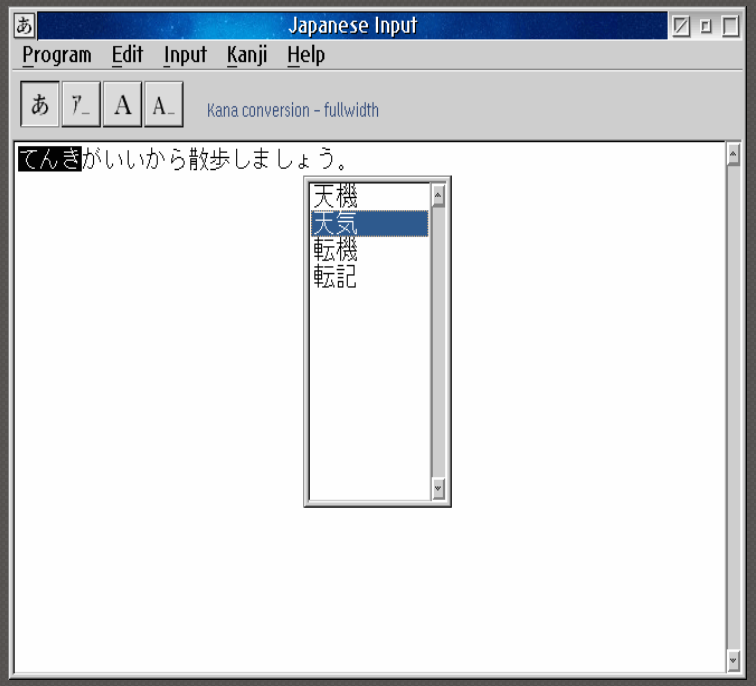
Simple Japanese Input Method Editor

IMERJ: Background

- Need to type in Japanese on English OS/2 system.
  - The OS/2 Japanese IME only runs on Japanese OS/2. ☹
  - None of the open source IME engines had OS/2 ports either.
- So, I decided to write something crude but functional…
- → IMERJ 0.*x* (c.2007–2010)
  - Type Japanese text in a window, then copy/paste it to where it's needed.
  - Unicode clipboard support.
  - Single line of text only.
  - Kana (phonetic characters) only, no ideographic support.

IMERJ 2.0 (2018)

- Major new features:
  - Multi-line text now supported (i.e. full text-editor mode).
  - Revamped user interface.
  - Online help.
  - Undo (last action only).
  - Kanji conversion (sort of)!
  - Uses open source EDICT & KANJIDIC dictionaries.



WnnIM for OS/2

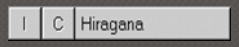
A true Input Method Editor based on FreeWnn

WnnIM/2: Motivation

- Using a program like IMERJ is a poor alternative to a proper IME.
  - Can't type directly into arbitrary applications.
  - Ideographic conversion is highly limited (dictionary words or single characters only) and the interface is awkward.
- The actual OS/2 IME has its own limitations.
  - Will not run on SBCS OS/2.
  - Limited to the system (NLV) language only.
  - Old and not very sophisticated.
  - Some applications don't support it very well.
- On the other hand, there are several mature open source IME engines (FreeWnn, Canna, Anthy)…

WnnIM/2: Development

- I chose FreeWnn as the IME backend because:
  - It is sophisticated and highly mature.
  - It has fairly extensive documentation (albeit in Japanese).
  - It supports Japanese, Korean, and both forms of Chinese.
  - I was able to build it and get it working on OS/2. :p
- Full RPMs (core, development, source) built in 2018.
- Wrote some simple test programs to learn the API.
- Began writing a PM integrated front-end in March.
- First alpha release supports Japanese phonetic input only (no ideographic conversion).
- Second release will support clause and phrase based ideographic (kanji) conversion.

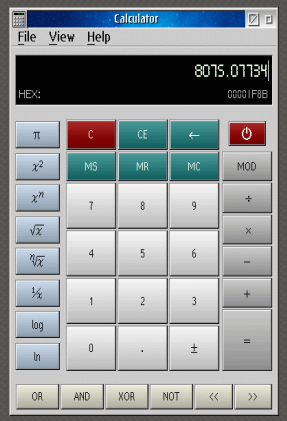


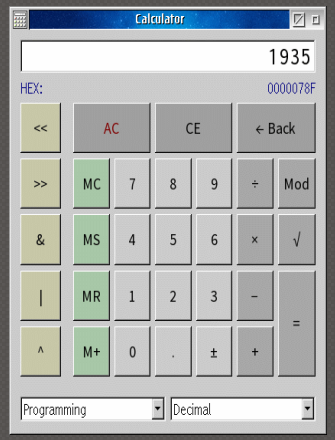
Useful Calculator
AT Calculator

Simple desktop calculators

Calculators

- *Useful Calculator* is a simple desktop calculator (native PM app).
  - No external dependencies (just standard OS/2).
  - Runs on Warp 3 (but Warp 4+ is recommended).
- *AT Calculator* is a (slightly) more sophisticated calculator written in Qt4.
  - Decimal or hexadecimal input modes.
  - Dynamically resizable with fully customizable fonts.
- Common Features:
  - Arithmetic, algebraic and bitwise functions.
  - Mathematical order of operations.





LVMPM

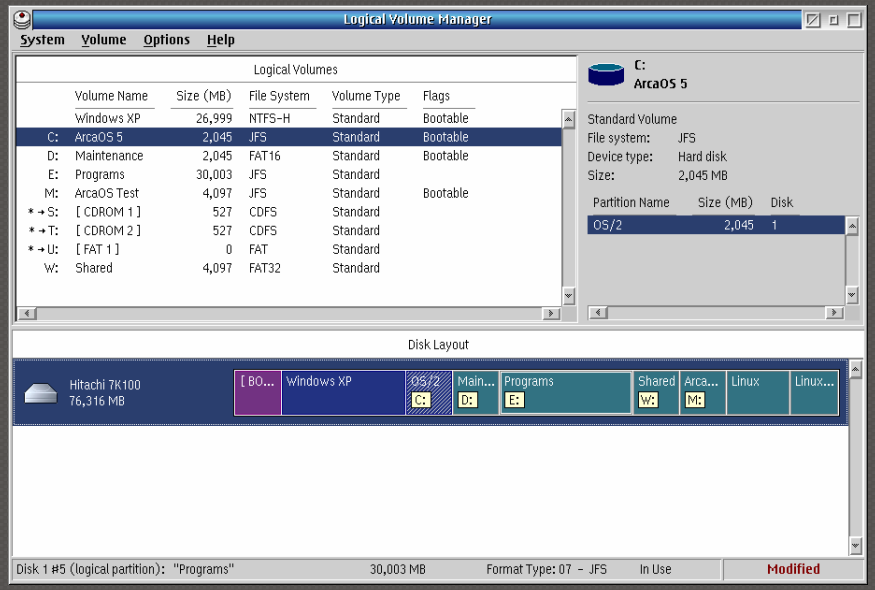
Graphical Logical Volume Manager
(replacement for LVMGUI)

LVMPM: Background

- *OS/2 Warp Server for e-business* introduced the Logical Volume Manager.
- IBM provided two user interfaces to LVM:
  - LVM.EXE (text mode/VIO program)
  - LVMGUI (graphical Java application)
- Both had serious (but very different) shortcomings, and OS/2 users found them confusing and difficult to use.
- The 'LVM Redesign Project' was born:
  1. LVM developer's toolkit & documentation (c. 2002)
  2. Installation Volume Manager (c. 2003 – present)
  3. LVMPM: https://github.com/altsan/os2-lvmpm/releases

LVMPM: Concept

- Replacement for LVMGUI
  - Native PM application.
  - Shows logical and physical information on one screen.
  - Allows access to advanced LVM features.



QE

A ‘simple’ cross platform text editor

QE: Motivation

- Sometimes we just want a simple text editor that:
  - Starts up quickly
  - Is simple & uncluttered
  - Can search/replace, toggle word wrap, load large files…
  - Doesn't waste resources on unnecessary features
- There are several options (E, AE, EE, …) but they inherit the weaknesses of the standard MLE.
- Even sophisticated OS/2 editors are almost all trapped in the 'codepage ghetto'.
- Anti-aliased fonts would be nice, especially on modern high-resolution displays…

QE: Concept

- We live in a post-codepage world, but many of OS/2's basic tools don't reflect this.
  - I have to deal with text files in English, French, German, Russian, Japanese, Korean, Chinese and others, using encodings like Windows-1252, ISO-2022, EUC, Shift-JIS, KOI8-R, UTF-8, UTF-16...
- Qt4 provides full support for all of these. However, most existing Qt4 editors are specialized or gimmicky.
- I wrote QE to fit my needs. It looks and works much like (A)E, but supports Unicode and multiple text encodings.
- It also lets me use an anti-aliased editor font.

QE: Other Features

- Written in Qt4, but with some OS/2 specific enhancements:
  - 'Native' file dialog
  - OS/2 native help
  - Additional OS/2 text encodings
  - EA aware
- Printing via CUPS (not native printers… yet).
- Optimized for working with text – not code, not formatted documents.
- Simple, simple, simple.

Note: *QE* does not officially stand for anything (although I am considering 'quintessential editor').

Conclusions

Conclusions

- Maybe now we have a few better tools for dealing with different languages and text formats.
- Simple system-type utilities can be just as valuable as fancy applications.
- Qt makes it incredibly easy to write very powerful applications!
  - → Please consider donating to the Qt5 project!