I could introduce this project in a million different ways, and they’d all be eleven pages long. Allow me to attempt to sum it up in a few small paragraphs.

What is UML?

In my first conception of this, I coined the term “Universal Machine Language” (UML) referring to my idea of a programming language based solely in an extensible, device and information classifying database. That is, for every unique function a device can have, a word would be created to represent that feature. For example, the word “print” for a printer can represent it’s “printing” function, and computers would simply interpret the word and use the printer to print a photo that was associated with that word. No convoluted syntax, just well organized logic maps with natural language labels that computers can be taught to understand. Engineers can design their devices to interpret and understand the UML programming language, or even add to the UML programming language by implementing new contexts and words to interpret, making their inventions programmable with computers.

As soon as I came up with this concept, I did a search for the acronym “UML” to see if it had already been used. Sure enough, there was a search result. Turns out, UML actually stands for “Universal Modeling Language”, and the article described UML as mapping out logic maps that defined machine functionality. The article continued to talk about the implementation of UML, via XML “Extensible Markup Language” which featured a set of rules for formatting information into data structures using <tags>, and it was purely focused on data. It was noted by the tech community that XML was more of a principal than a technology, and hardly useful without special implementation. Microsoft Spreadsheet uses XML format to store data tables, and HTML5 (web technology) works XML into their system somehow too; and in fact, HTML is essentially the first realization of the XML principal.

What baffled me was how much my thoughts completely matched what society had thought of and tried out. I had to conclude that UML was a truly omnipresent principle where man is bound to unite under, even without special advocation. In implementing UML, we as a society are discovering the eternal truth that will unify all our machines, and ourselves.

The problem:

Modern computer technology is very powerful and effective, but between the depths of computer hardware, the overly technical nature of programming language syntax, and the redundant and cluttered nature of various gui’s, computer technology is a mess. What do I mean exactly?

In particular, consider your windows desktop, a file browser, a web browser, and a websight. What do they all have in common? Utility access (program icons or apps), file access, folder organization, and file search. These four features are essentially duplicated between the four software contexts. The chart below compares these four features between contexts and provides a short description or example of how the feature is redundantly executed.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Utility Access | File Storage & Access | Folder Organization | File Search |
| Desktop | clicking program icons on the screen | saving files to the desktop | Right click on the desktop and select “new folder” | Open Start menu and type in the search bar that appears |
| File Browser | If programs are organized into folders, file explorer can be used to open programs; or file explorer can access the “desktop” tab to open programs from there. | main feature | Main feature | Main feature |
| Web Browser | Websights are practically the same thing as “programs and files”. The main difference is that web pages are much like news papers in layout. | Browsers can “bookmark” (save) pages to the browser menu | Pages (files) saved in bookmarks can often be organized into folders within the browser menu | Main feature |
| Websight | Often have a home page with quick access links to other pages, which are much like files and programs; including video playback, online utilities, calculators, measurement conversions, and resources for research. | “File access” goes along with the links on a webpage that “access” other pages with information or utilities. | Most websights allow you to sign up for an account, after which you can save several web pages or documents from the websight into your profile (folders). Video sights, for example, allow you to create “playlists” (folders). | Most websights have a search bar embedded in them to search their content with. |

This is just one quirk of information technology.

Another issue is how many different people create so many of the same web utilities, such as grocery or recipe apps, calendar or alarm apps and video streaming services that can’t access each other’s databases. For example, I have a google calendar, and a work schedule calendar. I like the alarm feature on my google-powered phone calendar, but the only way to utilize it is to manually copy the schedule from my work calendar into it. I can’t just sync my work calendar with my phone so I can have my alarm set automatically. This is just one example of potential utility from a unified world data base.

With modern phones and tv’s slowly evolving into computers, and everything becoming more integrated with “the cloud”, more and more there is a need for a technologically unifying standard for information devices.

My ultimate goal

* To unify all of the world's information technology into a fine-tuned, fully interconnected world communication technology standard
* Converge desktops, file browsers, web browsers and websights into a central information & utility mapping gui
* Converge email providers, file storage, web accounts and web providers into a central databasing framework (securely of course)
* BONUS: Subsitute utf8 encoding with a comprehensive **written language analysis encoding system** (wlaes). This will render computers completely independent of a central character encoding agency and make computers a truly open ended communication machine.

My background

I’m purely self-taught, though I have had one programming class and a one year internship. I’ve never made a living out of it, but I’ve worked in C, C++, C# and HTML. I’ve build my most recent computer from parts, spending about a thousand dollars on it. I’ve gained a strong engineer/programmer + consumer sense. The issue of technological unity comes up many times, and many failed attempts ensue that lead society to doubt the ones who actually succeed at it. I don’t consider myself anything special (if I can help it), but simply knowing what you want out of technology as a consumer, then having the technical knowledge to implement a prototype that meets your own personal needs effectively is all one really needs to create something marvelous. But if there’s anyone doing exactly what I’m doing, quite like I’m doing it, let me know.