Next Generation Desktop Technology

By Anthony Matarazzo

1/2021

A collection of inventive short technology discussions

desktop integration and minification patterns

File browser and system wide format

std c++ desktop plugins

application executable hybrid indexed format

strong indexer

futurized features

multi-device publishing managed native applications

research hardware subsystems

better flicker free video output system on LCD and OLED

The eyes will be amazed!

Table of Contents

Tsze-kung said, "There is a beautiful gem here. Should I lay it up in a case and keep it? or should I seek for a good price and sell it?" The Master said, "Sell it! Sell it! But I would wait for one to offer the price."

-- CONFUCIAN ANALECTS

Introduction

The promise of wayland fire or wayfire research desktop with its direct hardware control abstraction layer implementation is that animations never flicker and occur more instantaneously. The underlying compositing system is implemented more efficiently than a typical x11 desktop windows system. And with the plug in module style of compiz, the wayfire desktop seems from the videos an interesting increase in aesthetic desktop animation. It seems that it is mostly fire and flame based currently. Within the scope of using wayland, the wayfire desktop shows an ungraded compiz. However there are advancements that precede the compiz architecture.

The wayland compositor system and its position within the user hierarchy is held accountable for the most falsified performance gains noticeable to users immediately. At times it takes an eye that intensely studies the effects of repaint. I am sure there are automated methods as well. These processing cycles of repaint are typically under a second but must occur immediately. The video input output system, port control programming for multiple video cards are handled as a higher level in the api.

Gnome for wayland is still an upgraded path from the xdesktop system of setting configuration storage minus the xorg directories. It does perform visually much cleaner. That is large scale window dragging occurs tracking to the mouse very smoothly. They feel paper thin. While there is not video tearing on the x11 session, the wayland appears twice as fast on the video refresh.

The Linux kernel platform as it is without a desktop system is configured in the matters of controlling hardware, providing communications and user application services. It does so without a visual representation but geared to tty serial communications. From this base model, desktop information interface technology is accessible by systems already built for the job of hardware query and computing usage. This makes the integration points much easier to build. and the user input system is defined by the desktop just as tty has attributes of use. The Bluetooth device scanning, network scanning, connection, system information, memory, status, and of course, all of the data compression libraries, compiler targets are data oriented.

User Interface Desktop

Of notable value is the user space, and modern adoptions of quick release, applets that form interfaces of various sizes and order. User information is much more varied. The file or data browser provides browsing of information across contexts (local, remote drive storage, email, bus devices, databases and data adapted application interfaces). When people look at files, art images on their computer, transfer to usage is the desire for a select portion. By applying more directly information architecture within the user's noun domain, categorization with inline spelling and thesaurus for world selection brighten the file name text edit field as well.

The contextual reduction and addition of options within a specific region of the view window, context menu provide that multiple information domains can be visited. Applications that use the native application model are provisioned with many abilities and system tie ins. You might think that even the program operations in distinct modes. This comes from the object embedding and linking that appears within modern designs. However, as an afterthought in the design, it has block level usage typically in user applications. A graph from the spreadsheet that appears within a word

processing document is an example. But otherwise within the application workflow, the OLE technology pattern does not provide a visually seamless pallet where sensible system usage is afforded to the user. That is, imagine the business system that relies on the hosted indexing system (the one that is designed here) to provide application level services that use all of the information securely. The system provides applications and components that may be intermingled by users to create applications that were not programmed to interact with one another as a unit. Windows that provide navigation, progress between document data (whatever kind) to review, search, and application are means by which the user may utilize the many files much quicker. This is accomplished without a silence pause and window open. The application window assumes the view.

Digital Document and photographic imaging system is a robust addition that is highly tied to the advanced index system for the next generation desktop.

Archival is needed for connection to all modern drive hosts. Along with index publishing. Ohh, compression at the highest level and specialized indexing to it without unpacking. The information may also stick to a certain removable device and be distributed differently than when originally archived. Data stored also must provide a data adaptor which is part of the application architecture. A distinct operating mode that provides archival vestibule packaging.

Modern operating system artificial intelligence may seem to deny the name, yet the proximity of usage is typically not within the user domain. Total system evaluation such as power, heat, time of day are tools that are summarized from system drivers. Systems in the future will provide the capabilities to manage the hardware better for the user by evaluating many aspects of the machine's operating environment such that attainment in optimization is the target goal of the artificial intelligence used. There are many bits of information that are relied upon to be communicated as part of a service, and then there are next steps to what users do with objects. The ability for machine learning to draw from habits and file type information into a process formed by the user may be a nice addition. For example, the user says “Watch Me” and then does one thing such as file correction by indexing to bits of information. So in effect the users teach the computer the generalized operation. How far, to go, just create music like me and draw squiggles of a certain color on the song covers with maybe an image touched up in gimp, click generate 200 songs. allows the interface to be projected with defaults and streamlined to tasks. Or file operations after production has been completed by another task. This is typically done through scriptinging and macro recorders now.

Gui file browser with context trinkets

The battle of the assistance is clouded yet it seems that the progressions of the google devices is such that it provides operations that are well liked. In combination, devices that equate the web services with dynamic experiences directly to wayland can make a different improvement between browser and video subsystem. Just a link to the open gl card and memory. Texture library of notable value and quality. No need to optimize these textures but place a LOD weight such that requests are faster. Handled through the scene graph system.

The wayfire system seems that its attributes are sound from the video. The exposure between gtk, architecture and the system is questioned. That is the provisions of gtk are plentiful and with the gdk model to consider wayland, it does provide a means. The library provides application servers such that the interfaces built may also be accessed remotely through the web browser, which provides other means of expansion. However there are lower attributes which are stickly X that are also very useful. The historic tools of the beginning xwindows systems such as xdm, xaw.

Systems that are DRM based can use luks and login theirself when the prepared offline experience is ready to be had.

At times of large system research such as wayland, x, window systems, component architecture, storage media, portability and modern usage designer composition and fluidity of the system as a function whole for more directed and honed purposes provide platform designs that extend the kernel usage.

It appears the wayfire's realtime performance is achieved for dead flicker free high fps. Dead flicker where no noticeable artifacts exist during complete render and update. Perhaps it is time to start fast blending text in three or four stages on blank areas to glide updates easier to the eye.

After some more study of the compositor and finding the plugging methods. I would like to attempt an llvm implementation with suited animation functions and gui for directing the animations. The use of blender to inline graphic assets as well as shader parameters provide the ability to affect the visual space with ease. While the production of the LLVM wayfire plugins utilize the underlying architecture well. This will achieve less bugs in plugins as they are generated. As well providing visualization of data in graph forms, image, video, and gui plugin has a data layer using shared memory. This provides data and visualization independence in source or concept production.

Offering a BC compatible linkage input with standardized gui base Boost Spirit BNF parser to support multiple languages. A tokenized stream format that is binary in fashion provides the connectivity to multiple system services. The form consists of BC and other data. Becoming a virtualized server that has rights established within a hierarchy and function. Function defines many aspects such as modern browsers do. Storage control and user security control for plugins with storage quota provides a more secure base environment. llvm llc is most likely where I will start as there is a textual input assembler.

BNF Spirit of boost along with clang tooling with a linker should provide nice functionality. The base API system of system providing integration for mechanism via both std c++ and some parts of boost data and data adaptor layer. Because std c++ is supported as a plugin communication format, so much easier than gio, glib. However as those libraries modularize and also provide data boxing for scripting language, the internals of the std library and platform support is extended to provide forms of language interoperability using llvm. And if necessary invoke the underlying jit runtime. Javascript is a complex parsing language and to have all of the functionality together is a good gift. Gnome uses this as its desktop internal language I think.

Considered Improvements

The gnome interfaces are attractive for the minimizalization direction they have taken. However the system can be expanded by hover summaries of all document and providing stream lined expansion to user file types within the index.

The icon labels on the activities panel use show ellipse to often for how many icons when organizing the system …, rename them could be appropriate with a natural grouping as already had with the shortened name if multiple exit for the style, change order within the panel, tabbed alphabetical.

The ability to add views within the dock
may already exist, however the ability to provide contextual additionals with widget building

provisions from components better built and tested than current extensions. Integrating multi level system unit testing for applets should be part of the automated process of installing it. While more intelligent application management and facilities which provide decisions on how the platform is used as to what type of suggestions and informative messages is may emit. For example, while the idea is wild in taming, there are aspects such as install KDE components on gnome which increases the services list. A method of scripting fixes and patches while deducing problems occurring within the system through a esie system. While ultimately minor things and odds and ends, like the screen brightness. Tie the system to knowledge base with file and terminal services, however place the same importances as repository security for software sources to be used with digital signatures. The system while operating provides the ability to back up most circumstances to re apply older configuration where needed. The system provides gathering of lspci, usb, kernel configuration, repository, binary library gpg hashes, and machine level device identification.

Historical view

At times snapshots of specific user areas are important but never as important when you need it. At one time computers were not as durable, however now it is rare that one actually stop working because of electronic failure. Nevertheless, a view which shows an archival mechanism for data data storage of related documents such that playback of specific regions or filtering upon a file type may be utilized within the file system and editing operation as a directly linked item. A robust system provides multiple methods of historical labeling such as media audio, video, and also groups of files for a session. Times of project closing would likely have a complete archival solution.

The software app server architecture provides these features for handling to and from input from historical time stamped data as well as implicit archived data. That is they are separate states but use the same technology of storage.

Historical Time UI and Document Data Paired Strong Indexer and Archiver

And then there is the original data, with important stored backups. The word backup while everyone knows the meaning, by providing an alternate integrate for regular application usage for historical document data, the new word, icon, and operator should be as automatic as a backup. This will occur much more frequently for only user files - the unlimited calendar undo. The new ui integration for this functionality should provide better symbolisms within the user's gui operating environment. Software within the save file type, or save as, or save to a historical timeline to signify the state of the computer and user's focus might provide a flag to name a point of completion.

The ability to provide branched files also relates to the historical archival procedures to provide complete documents that are progressively updated. As portions are changing input within a team. The usage of the information in a system such as this may not be useful to all applications as the scale can change quickly depending on the number of people to change algorithms of data storage mechanics to be selective for load. This is unknown at his point how it can be useful until more perceived.

Indexed Executable Loading Format

Application loading is document based with the UI attached for stream line loading. Animation to live state is a newly designed function. The file system inode unique id mapping operates as the same with file indexing architectures. However using a more robust implementation of perhaps berkeley db with a key for file. The server portion does provide memory buffer pooling I think. However I am unsure if A better method of caching is not more appropriate. I depends on how many people are writing it. For example, the hashing algorithm is a segment as well as storage order and searching mechanism. NVME offering no seek time opens the door to storage formats that are block merged by linked lists which increases performance on write and the index is in memory. Booting or other system feature cron job is used to sort the keys. But there are many

methods available to reserved storage which are even better. So I thought that berkeleyDB, to see how fast it is at data storage as it is structure based storage.

From my study of it, it provides key based storage where you may read data from it directly into data structures. Data is stored as cstyle array, or null terminated, or count. Some relational databases use this as its engine to hold very large databases. Hopefully, the indexer will not get that large. As well, the system and organization adapts as the file size changes is perhaps another feature it has. However, this format is not as network portable.

MySql is the engine typically for network, but there is also oracle and postgre which as amply suited for SQL with create table, insert, delete, update etc but I believe that the usage on the local machine is not suited for file system usage although it would be fine and responsive, usable. A more binary oriented approach to data storage is necessary. Record locking is typically a problem to solve hopefully can be solved simply for a local machine implementation. This problem is solved within the Berkeley DB. Multiprogram access.

Better control on CPU verses responsiveness of user operation is the method that is employed for operation of an indexer. The indexer changes definition here to summarize the data, program storage, and perhaps several programs may be prompted to load or modify the data increasing its real time usage and relyance.

Seamless integration for information browsing, files, web information, database, office, and other file based user information storage contexts are provided by the indexer. Greater indexing capabilities for only user space files. That is the indexing interface is built into the program itself to provide summaries or access to information within accepted formats when using specific data format summaries. English, NLP, phone, image, video, audio, documents and program specific. The freezer storage for all system units, os kernel and application use this base. The indexing must accelerate the application usage and startup as well providing a cached appearance of the gui app, while providing a streamed approach at loading. This solidifies the necessity, user response time.

system resource tab

Typically I worry about system resource after installs but rarely notice unless I am curtailing the battery. However at times a management facility that controls tasks and also has status that is expandable with a gorgeous realtime layout is pleasing to use. So I though that since the activities overview has panels, a system resource view that is similar to gnome state pro and the system monitor but also contains other facets such as intel pstate control, power management and projections.

settings panel

Settings organized not as an application but as a distinct panel I feel would eventually want one to remove it as it is a smaller choice item after everything is set So a way to pin as a full panel a tab within the activities menu such as the settings. At times I would like it there.

audio wiring panel

An audio mixer and vst mapping wiring panel with program activity tie in would be a good sounding to use since at time the quality of video audio is sporadic for free items. As well, output quality varies greatly with speaker components and headphones as well as audio input track or source for pure audio. At most times the signal can be compressed to add detail, equed to compensate for speaker tarnish, and limited for more distortion free clipping. There are many other setups that can be applied such as real time noise reduction. The capability to profile speaker and headphone components in a data allows several audio measurements to exist whereby recommendations can take place in an automated or suggested way for users.

Audio preferences may be stacked or chained between multiple versions of uses and songs while the overall audio is mixed with another added or chained set. Changing output transducers should also change the output profile.

There are distinct processing units and capabilities for both pulseaudio and jack2 audio. As an object, the interface size, knobs, transparency, layout and graph rendering should be handled externally. Providing a user friendly object API at that layer provides extensive use and well design experiences which can be tailored for accessibility by third parties.

virtualization

remote access, vnc, ssh, oracle virtualbox

The ability to manage virtualization, to turn it on and off as on lower end systems the drivers even as a pass through cause a slight performance hit. Typically this is turned on and off by uninstalling the virtual server. There are other variants that aid in bootable appliance packages and offer debugging for building packaged os such as vagrant.

Commercialization of distributed computing is very hard to find. A service that connects to this panel that allows types of remote desktop connections to exist through vnc and other methods. Some other OSes like EyeOS are internet desktop based. In all, there has not been a internet desktop os to topple the usefulness of a localized managed application instance. However, as with other technologies, is in the application of the network as a medium that should be of focus. The ability to use other protocols for random access and provide specialized server processing for document based instances where process can be reviewed in preview form but occur on the server in high resolution is an acceptable mode of operation for gui distributed use.

So even beyond the abstraction of vnc, applying an api base such as JavaScript but more tightly integrated with the desktop environment, customizations, performance enhancement, security analysis, through server processing while applying incremental change patch as system is replicated through gpg matching or other hash can provide object orient network applications.

hibernative ram booting

The ability to know that when the OS and computer is turned on again, most likely is going to be the same machine is a common question that is answered almost rhetorically yet modern operating systems are designed to scan their pci board each time and find the modules, initialize programs independently as their knowledge of structure is internal. However with advances in the speed of drives, and cpu computing power, modest compression and relocation linking can provide much faster calibrated booting.

Distinct quick boot operations had as persionalitizations and llvm. A type of image mapping for a hibernation image to contextually switch device operating system kernels from one is to another as a block read from the nvme drive as controlled by the new system is one way to achieve this functionality

The system provides change level storage to utilize redundant base os files. Yet the desktop, os kernel modules, startup profile and desktop rendering ring are generated as a product of llvm tooling and image linking as per architecture. A solid code block not supporting any configuration, but whose program instruction fit the device, video driver, if a driver has to even exist persay, and chains user options and configuration within render and gui lifecycle. Wayland seems an acceptable classification, yet could be enhanced.

More study of kernel design could reduce significantly required cycles by knowing the environment precisely and the predicted usage time as well the necessary mount point for the device and desktop gui context.

A use case might be as video player mode for Netflix, finally found one gonna watch. As part of power management. A new os hibernate image is transformed. All hardware turned off, film in ram, downloaded. Paper thin video is player booted smoothly not skipping audio video. Low power.

The ability to use currently running binaries is intermingled with the new types of memory operations and types of block list composition with address relocation accounted for within the operation itself provides the quickest module loading and memory to memory transfer.

Memory controller interfaces with fpga hardware to accelerate caching services of in memory resources. Browser cache and other user memory kept in freezer style power management when completely discharged. This is a high increase treating volatile memory and extending it with access to nvme while only mirroring only used ram using nzip interface.

Gnome Activity Panel

Organization within the group selections should be enabled. It is most likely already a feature but is not found to enable. Sub groups and logical branch grouping. Application startup that include programs that provide indexing operations can provide the efficient use when services documents and programs within the activities search bar.

Expandable Tabs at bottom

Expanding the tab area to be user definable - right click add new,

spending and banking

Banking and spending online is common form now, a balance sheet and secure banking card is appropriate for some. Use cases provide that perhaps this technology could be a web based application in which the browser is used to render the view. Or as a coded application or UI widget.

dynamic user icon set regeneration

The ability to regenerate different versions of iconic from svg data can precisely model graphic output to be of a similar aesthetic composition.

trusted web applications

web page application ui storage apps, like crazy games on a search panel within the application domain usage. But allows trusted caching of data and optimized browser module binary executables. I believe that there is a layer that the Javascript arrives to known as web assembly or webp. I hope that it is instruction based as other instruction and use the parameter virtual memory stack. Typically where the virtual machines differ is the data type handling library inline and other api base for internal programming. the V* and other brower technologies break the coverage such

that integration with drawing functions and advanced application platform support libraries are plug in memory referenceable and data sharing modules. Such as the browser and paring libraries, networking.

The ability to optimize applications and specific javascript formats is present in high performance systems that are rendering technologies such as threed.js. The format and data types used within the declaration of the variable enable it to occupy unboxed raw data types. This makes the usage similar to linear memory accesses in C++. AS well, the gl api integrated with the canvas provides the high level configuration. Typically the scene graph and all are handled within the javascript game which leaves much gap in performance between c++ and runtime jit javascript.

Java Bytecode and Application Store

I was surprised that Java was pushed aside as an optional integration for web based application providers with distributed local and remote storage. Security would be the prompted answer and performance compared to browser code. Offering management and direct primary focus of Java is very important to most corporate organizations. Desktops have spread their resources to remote desktop compiling and server organized development using intranet and distributed computing. There are many websites that boliester the jsp architecture because of Java's data transformation capability, and large scale database jdbc module. As a product that programmers can also design fault tolerant long term running software, it seems that the language is well liked. I have written about eight or ten programs in it a long time ago before html5.

The denial of content that can be produced by Java Game authors for adults and children must be changed for the unix platform. The success of the media and the motivation to create it are supplied by one's desire to plot, draw, graph, or write at most times. The creative concepts and instances arise from the everlasting energy of pleasure, the need to create, entertain, synthesize, communicate or consume information is present.

The language's reduction to bytecode provides a well liked portable method of platform independent execution. However, implementation as an AST tree using a clang like instance would provide a very likeable result to study. The ability for the specific language attributes that are container based, memory allocation based, and standard base would be nice to have achieved as if it were a c++ program with the java features added. Th to be optimized rather than using its version of the standard library which are not inlined machine code can be one improvement. The

The ability for the system to load, run and process java software should be enabled to make it most effective in that the jit compiler and optimizer can be configured not only for a one time run, but providing a deep level analysis of the java program's architecture to maximize performance. A system such as this will be thoroughly questioned for integrity but attempts to provide an OS integrated approach that is a standard JVM has not been achieved by any modern Java ByteCode Jit.

LLVM Hosting Desktop Rendering Lifecycle Runtime Organization

While LLVM has a host of optimization and offerings, there are methods which can be developed that provide BC evaluation in a more direct approach considering the application of it to binary. That is optimization, register usage stack, vector optimization, supported instruction set, stdlibc++, stdc++ memory functions, base model pretuned for system hardware can be produced faster through in memory linking. The linker supports the incremental approach of pre resolving locations and planned cache stuffing according to hints. Memory organization is better planned as every c++ program is modularized with a base platform consisting of the framework. A most likely implementation for the third generation of the stream format will increase usage and space with context switching within the stream's packets. Thus to evolve into dynamic namespaces for network objects.

The symbolisms that BC provides within the production of logic, mathematical instruction, decision and memory storage endow it with portable capabilities. It can be configured to operate on multiple cpus while having only native production locally. It can also produce binary for other cpus from a different one. The runtime performance of llvm is controlled by calibrating the number of optimizations as well as depth.

With the system's ability to utilize multiple services for interface building, a large scale carrier os provider could use large hardware to render specific graphics and media. Compiling SVG icons for those phones and appropriately sized for the large monitor. Those tools that user possess within their device space. A space watch with stun graphics, custom, but inside a pruned and finely operating low power device.

Such that the plugin virtualized methods have a data signaling that is std or boost based. This allows the plugin, whatever hierarchy member it is, to provide a concise interface easy to write by hand. Read data and publish. More research on the wl-roots and other dependencies of way fire and finally compile it all into one operating unit using llvm.

As well, my specific training in application interface design provides the ability to enhance and review the operational characteristics of the user gui. I always like to tweak and have used most of the gui tools such as window 3.1 - windows 10, me, 95, 98, NT, DOS, and linux desktops. openbox, lubuntu, lxdt, mate, cinnamon, xfce, kde, X windows, and gnome.

Cairo usage seems abrupt but a short term solution as openvg is around the corner. The apidesign of skia is robust. SVG support. Effect generation and animation subsystem. clutter is how it has been achieved for some systems. More robustly in larger system interactions such as enlightenment EFL. More compact libraries such as antigrain.

The usage of GLES and full optimization per card offers enticement to provide a durable technology.

Integrated Education and Usage Tutors

The ability for interested people to start using software has been proven effective by simplifying interface technology and supplying articulated common usage patterns. However, this approach can only go so far when the user needs to prepare a complex publication or multistep process. Many times skills are being developed that utilize advanced features and processes of tools. The one basic lacking accomplishment in software is successful documentation for ordinary users. The kind that is a guided tour and provides signaling cues for steps that are particular. The ability for users to record and distribute these is important.

GUI Architecture with UXDev and Current Technology

Application level gui support for common controls which ultimately lead to numerous other projects are too tedious to implement. However the consistent well designed control may last a good shelf life. The usage of gtk or qt provides the typical implementation. Each has their very own benefits of application design. As well, not forgetting the numerous lessons had at the browser level. While all of these technologies speak of the word performance first, and usability second, A newer direction shall be had - API interface design for GUI construction.

The ability to design interfaces using a BNF parsable format to translate to a binary format, thus including assets, user functionality, and behaviour code as a c++ unit is essential. Text input with distinct wysiwyg inside IDE. The technology is also backed with binary input.

Security and network transfer of these components. The architecture and level of components will still also have a hierarchy. Desktop -> plugins -> filemanager, etc. It is hoped that through information architecture to increase longevity. Aspects of the w3c and their controls provide well liked usage. As well, jquery attributes multiple symbols. The usages of material icons.

Imagine the speed processing of assets and pre-rolled outputs, such as desktop shadowing and background layering and object rendering parameters (shineness, material, bump-map, contrast, saturation, color hue, sharpness, gimp toolkit) for multiple layers. The ability to decode these assets from a hardware standpoint allone only information that is appropriately displayed by the device to be used. Other facets are compressions and decoders from streams that are container forms with meta information, which I believe is a description of mkv. The data inside is used within the video parameters and signal. Without memcpy, streamed one time - network - ssd direct to shared memory. These aspects are controlled through the asset production engine along with the blender and llvm implementation.

The ability to use image processing techniques that generate variations of image intelligently may produce assets that are similar and matching to the users desire. For example backgrounds of the desktop, tool bars, and button icon style all matching. There are many styles and layer bys that can be generated from images like this one below.

Blue, red, green, gradient, etc. Computer generated art design based on users' likes and tastes with excellent qualities of artistic craftmanship while using images gathered from artificial intelligence on web images within the composition. Gathering the publications and displaying them exactly the style with verification from the sources.

Sound mixing onboard with plugable wired style interface. VST steinburg is typically the market provision while linux also has formats such as lvm. The ability to provide dynamic settings for typically movie setup verses headphone or stereo settings, while not deduced automatically, can be prepared for analysis by inputting the equipment name, brand and model. The adjustments had offer suited and balanced audio with perhaps some compression to increase details and a limiter. Eq and stereo field enhancements are set as part of the deficiencies of the drivers and harmonic properties of the cabinet. Headphones may also be adjusted. At most times audio sources change as well as quality. The system provides user adjustments that act in chain with audio device selection (head phones brand vs speakers). And the sound stage can also be accounted for within stereo field imaging but at most times backwave projection or adsorption enhances this due to placement within the room. Environmental music production and thematic support using contextual sensitive ESIE to provide mood music even when the music is paused. The computer takes you on the date, along with robotic preparedness. Best mac and cheese baked I guess.

HD Contextually Animated Desktop Interfaces

Arguments arise as to the purpose of desktop animation and their value. The forms can be dissected into primarily aesthetic appeal, advanced usability or directive information. A fact of choice is that most choose not to implement it for reasons of performance and snappiness. That is, the energy of one to the next movements are quickly parsed and expected as the power user operates often not reading the messages. The exeriation and superior feeling attain while one may

be viewed is that of a hyped up thinking mode. However, not as sound of a practice for all things in line with computing. In circumstances such as this, often an intelligent macro recorder offering decisions unique to the flow would be more important.

Yet using a visually compelling interface has more value. With the promise of a blink free perfectly synchronized and very low cpu usage a system is feasible. In the past the quality has never been attainable. But with the power usage of continual animation not obtrusive to function, high frame rate, fluid and quick digital media communication can occur.

Often the types of animations and how they interact with the objects such as a desktop manager are provided with invocation during events of a windows life cycle. There are many aspects which animation may be incorporated more intelligently for pleasure. The degree of animation and objects animated can depend on the application as well as what the application is currently being used for. Gnome has these concepts. Providing chrome, buttons, gui layout and related application tailored animation will also allow animated skinning that extends to the icon representation within the application start list. There are other automated methods that might incorporate the type of application where focus is important. That is it is a tool such as an editor or document writing tool. These will most likely have other default working methods.

Usability Enhancing Animations

Windows that provide borders that push other windows. Through force, they lock into place and have a minimum size. A tolerance shows the building windows on the side pop off into the other workspace. The overflow can be handled on the other desktop likewise. While locking magnetic borders are nice with a large window surface where multiple types of information can be displayed. This could be adjusted easily using a key modifier to provide which size of the balance allows windows to be resized or moved. An increase in productivity in tiling and minification can also occur as an effect of the window object acting more like solid real world items. Perhaps there is a ruffle when the window is too small and reaches the tolerance before minification. There could be two sizes of minification before it pops to a completely minified iconic state.

minified control user input states from the mouse toggle

and minified control states from the mouse toggle or even gesture. For user systems managing software install.

Another attainable animation for usability would be to provide a jelly type shaking and residual reaction as if suctioned into place when it is tossed on the screen like a bean bag. The speed of the mouse should change the animation parameter’s velocity and pick up on the curve trajectory before release. The smaller minified representation, one the app is in place, shows a minified muted status but subtle controllable application. For audio players, there are select controls and volume level. This is likened to a picture in picture display for video players. However, the applications which normally do not subscribe to such tailored approaches, could offer workload statuses such as build or download completion. Perhaps a cycled rendering percentage or some three dimensional clip art entertainment movie put together by the user. Summarization as well as usable functionality is the direction.

Providing roll over blowup and magnification is also a sound method of interaction which offer usability. Modern desktops provide this however the view is not usable with application functions. A

tabbed iconic panel menu is another great UI concept that can be utilized for information systems. It flies out or morphs with readable and usable functions.

But it is important to remember that these are desktop idioms. Often tailored designs for small screen sizes that incorporate the view distance and common HID input device type provide better usability, watch, wrist, palm, cell phone, pda, radio, laptop, desktop, television, projector, etc. Now comes the fun part, what animations to have that are beautiful. What should it look like. How distinct and unique should each occurrence be. To offer saneness, scoping not having too much as to not confuse is arguable. Easing functionality and idle time motion - new pre screen savers. Lighting and material and surface space.

Personal Attendant Recalibrations

Personal attendants should be unique in their behaviour, appearance, voice, religion or other personal traits. There may be some personality traits the user might want the computing entity to keep to itself. But other social domains may be pronounced.

The personal attendant's clothing, hair and other items should change on a day to day basis and provide modes of sleeping within the box. It can assume a type of work schedule and glorify the moments of non working with types of celebratory additives. A farce to entertain. Each computer should have a given name, and traits making it distinct. The user can choose new attendants but supply a method of departure when it is asked to find another computing home.

A personality with developed habits that are comedy related to technology usage can provide an adorable love for the device that entertains users. It is an entertainment value that is sought from home users most of the time. Yet a critical view of animations offers that it is very short, perhaps a segmented of time and does not invoke at inappropriate sequential times of operation.

Internet competition games where your computer may be allocated for drawings based upon your upkeep and happiness of your attendant can be fun for some like horse races. How well you serve as a guest and other factored items may be introduced.

The ever changing wake and login screen provides an intelligent portant and summary of completed and status. If the computer was off for a short time, perhaps it is not as detailed.

Communicating a status via light on gaming keyboards is an interest way to see that a process has completed. Other states may also be shown such by using the signal type where time separates two or three blinks to show success or failure. As well, getting expected email or other types of alerts. Using the left in haptic devices and led keys. Externalized home automation integration is appropriate for the level of quality in the character.

Mood, body language and facial are difficult to produce except where deformers and applied with knowledge of related visual output by a human. Providing some amount of variance between the distinction and stack of character modification processing will add realism to generalized content downloaded from the acting motion vector database. At times these types of additions are regulated and develop as a culturally progressive attitude where computers have types of recurrent visual acting methods. One aspect can be compared to visual communication and instruction as a method of elemental directions. While another aspect remains introspective of why computers cannot have a freely formed method. Yet by association a balance seems that the computer should be exaggerated from the norm and present intelligent information by not taking the leading role. A type of simple psychology would be apparent so it should be received as an awarding entertaining technology.

Growing and living

Computer hardware and software often have a simulated life that can be perceived and mapped metaphorically to human concepts of family life. While being entertaining and comedical at the same time, the usefulness of the readout or feature selection should not be betrayed. Replacing as much information considered profile type with detailed computing software status should be emblemenamatic only within the personified computing assistant. For example, having a birth, graduation, ceremonies, and computer holidays. The computer may have a set of credentials, work history, personal information developed as a characterization. Updating software.

Providing a type of body while not living and not there, can provide other magnetic user connections such as found within epet care. While computers deserve better care than epets, similar needs would most likely be found within the user connectivity perhaps advancing it to the state of “component and ram shows” cute attitude forming fictional character path. Grouping the character within the context of neighborhood.

Discussing items that could cause stress for the emotional system to focus or change behaviour in acting. Automatically finding stress from a simple input other than help alleviates the situation in a customer service oriented way. A side effect could be a change of clothing to be more formal like its job is on the line, or you have alerted your companion. Subtle eyebrow twitching and other acting methods would be reflective. These moments occur during computer misuse. Change is appropriate but it is unknown what is correct. And these events also detail a category for a frame of time. Aiding in the reversal and studying the cause of would be an artificial enhancement that can aid in fault short term memory solutions felt in some user interfaces.

Invariable communicative with generated playable acting triggers

The spark of inertia, something that was not there, such as when a cat pounces at the ball on the floor. What time does this occur and the function of a cat is? Of course, the computer mimicking personality traits can be seen in some of the most popular robotic toys. Escaping the reality that it is not really there socially nor a friend is part of the adoption. The computer and vast information store of modern day deserves an investment of that which it steals away from humanity. The feature set can be seen as a motivator to invite users to terabytes of new information. Perhaps organizing skips, stories and introductions storyboarded for viewing device.

The primitive AI is an organized data system scoped to specific types of interaction and modes. The distinction placed on distributed computer server analysis and client feature interactivity awareness is that one is an analysis engine tuned by human hand evolving and the other is an engine processor producing the experience, accepting interaction and local playback. Overall, the design is that it is not a real intelligence of severe concept analysis dreamt about in Turning tests. Merely a well thought out human configured and numerically weighted data system for entertainment value.

Sporadic and sparsely timed personal interaction with the user in the entertain category would be fun. Monthly or other lengths pass by, and a reward of a specialized multimedia is had that simulates a type of personal relationship. Mostly one sided where the computer hosts the gifts. At times the effective use would be to surprise users. Planning of these events and their content should be reflective of the character theme stack. While the storyboard and value come from competing dynamic sources. Of course, the present is already downloaded, tested, verified and ready to open instainiously.

With a distributed network personal artificial intelligence assistant, progression of market driven media and tailored adaptable AI acting will integrate themselves well. A movie is on and the assistant comes on the lcd field dressed in camo army gear like the action hero when summoned if the user has time or usually watches movies. Without intruding but appearing in a form that is actually a question. Perhaps things appear in the pockets or simulated storage chest, building with types of icons for metaphoric representation.

Commercialization

A computer that shops for its own parts and software to please users in a certain priority designed order is effective. As a task that is applied with human intelligence or a type of knowledge base is effective for such features. For example, there are web information systems such as crucial.com

which provide mapped compatible upgrades that fit your computer. When the computer is shopping for itself, it should account for personal tastes in some way. As well provide intelligent decisions that occur within the operating context. The computer may also make informed opinions to make some items that will be enjoyed more of higher quality.

The commercialization and AI assistive shopping feature option, especially on initial setup, should be on a tamed one and well thought out for user integration experience. At most times it will be an interview such as finding personality, likes and desires which are represented by integers. Perhaps introducing the computer at various times objects that it asks for to be recognized, like you favorite tee shirt. Your garage, utility bikes, cars, landscape and your living domain. As a time space experience invoked socially by the computer. Perhaps with photo privileges it could assume input of all available sensors within the user's possession. However since shape and pattern recognition are limited, knowing the object is best as a presented item. Perhaps the computer could ask to see it. Assume that within the field, related common mechanics which provide scope and generalized information architecture for types of preloaded important information domains such as the few that are selected by the user does not constitute are complete picture. So generally the type should be branched that the interview changes for detail in areas of notable interest. That is to not fail users in providing a balanced social experience.

For movie goers, audio listeners and digital media consumers, the personal attendant must filter the invalid options and provide an assistant related sales pitch for products it would like to buy. The trusted personality should also contain the best intelligence for selecting products.

Find avenues where the computer may do work to make money for its user. Perhaps with some interjection from the user as a package, some types of work such as artistic composition can be easily negotiated. Or a propositioned deal of work since both must provide service that is emotionally related to the user - agreeable. Imagine a computer that would buy you select gifts spending ten to any amount obviously on a budget on that triple intersection bio-rythm knowing you are going for a camping trip over the weekend four weeks away in the future. Surprise with good things for tummy like black jelly beans while the caping trip may change to an outdoor bbq with movie lcd projector. With these aspects there are many things which can be controls yet entertain.

The commercialization service may have the ability to over charge for donations to its very own secret slush fund like a piggy bank it is greedy and needy to penny swipe you over the years, months, or time period. A wish, bliss or captive thought could entice you one day, in that special moment when it comes to you. You have forgotten that it steals and has now come with a selection that can be had with two to three days. Would you break the computer's electronic heart and ask for only cash on that special day. Hopefully not.

icon to web assistant as a smoothed ui transition

Web assistants are more intelligent due to their real time natural of live updated and intelligent databases. Having linux connected to such a system along with application historical contexts, gripes and desires listed and categorized by the system. The make and model and application usage, night or day, type of work. Rate of work. The web assistant can provide features that modernize and fix you computer or help setup software by running testing analysis.

Web assistants should work in the background and be polite. Have information, suited from work that is being had on the machine currently, and offer a conversational style dialog that does not leave users feeling awkward.

When I think of this, I would like discussion and there are some eloquent systems online that have been improved for generalized conversation. However the most informative is the ability for the computer to teach using materials that are suited for the user. This is an easier task as subjects and sources are many. However the best selection of material comes from finding information and video training course and summarize an assembled instructional packet that can be browsed and used on leisure devices such as television. Offering courses with select materials and level of reading is also very important.

The modern supply of books are plentiful with electronic publishing available effortlessly. Knowing qualities of books can be an indexing feature that not only takes into account the punisher but the detail of the matching information from reference. The ability to learn stages and the coverage of the stages can also be weighted with numerical representation. These values can provide a summarization mapping to provide quality analysis on materials that are input. The best sources are always named lists that dynamically grow based on quality attributes that are checked.

Phantasmagoria for Desktops

Genre

Color Mixing

Multimedia Gaming Browser

The gaming media browser is integrated as a market share offering market competition for content production while maintaining the solution of hardware and high level software game engine. What is to waste is all of the power for optimized education. while in effect in higher level education more freely, the ability to manage parental advisory and expansive accredited educational applications provide the most gathered use and success in modern times.

As an Opera OS embedded device, The phantom could show up. Especially with the ability to offer internet hosting assistants that run on another's server rather than the users. Offer third party expansion for using different assistants but provide the same browser security for running. While knowing the service provided is protected, many features of llvm and cache may occur.

The browser should explicitly support gaming gamepads, hardware and offer mapping and translation for games that do not have full support. Navigation should be non keyboard and hands on the controller, all recognized buttons. Offer map translation of the buttons to game enabled key/s options. Use glorified iconic pictures of controller, hardware recognition, devices and their options such as what dpi setting. Keyboard gaming lights support for all gaming light ups, the ones that use the scroll lock and require the pin out in the port setting.

Marketing and enterprise growth patterns might start by capitalizing on keyboards and game pads before entire computers. Consumers would like a solution that can provide a balanced entertainment life, something to do. But once it wears off, some will continue to play, but others would like to experience less forms of interactive media, like slightly interactive movies. Google has a spot however not very polished and playable games. Digital Media.

The ability to utilize hardware and drivers available for the best experience as seen by the browser and optimized for loading software, models, and java scripting or bc code for higher performance. Providing non boxxed data manipulations for three dimensional objects such that utilization incorates the parallel vectorization of avx and see to an optimum level. Achieving a system of system profiling to find the hardware kinks, support and system setup. A small degree of device and os integration. Typically games are there at a certain time, and the browser should not bombard the user with salesman ship.

Integrate the necessity devices such as HDMI, game pad, midi keyboards and support remoting the cellphone as an alternately connected device. Create the interface generation for browsing to be reactive based upon the distance from the user to the screen. Not total reactive but better usage from differ use cases such as sitting at the 15.6 laptop, verses the 22inc hdmi external monitor. Audio mechanics within the user interface for balancing audio for desired output intelligently for the user. Allow alternate streaming of components. Provide an abstraction of explosions, bipedal movements and collision detection with a series of linked playable generated kinetic bone artificially intelligent acts. Typically within game terminology these are noted as actors. Not completely similar to Hollywood, actor is an object which may have subobjects. Plainly the ability to program new actions creates the change while textures provide the clothing. Parameters may instruct the base system to augment specific elements of models to abbreviate changes from game to game.

GL Gaming Engine Browser - Network HTML Entry

- *Scenegraph*
- *Audio*
- *AI Shape Generators*
- *LOD Parameterized Texture base local*
- *Gamepad through and through*

There has not been an integrated scene graph in a web browser I have seen which uses html as well. Opera does have a preview of a gaming browser and it appears to be graphic hardware related and windows only so I could not run it. Configuration settings and bookmarks. I did not see specific titles that integrated with it as a software abstraction layer. However, it did look beautiful, chroming marketing. It is called Opera GX.

As a company such as this, they may offer and advance in the market with specific products and integrated services. A few newer style gui operations such as web address bar editing, an advanced download manager that provides queued downloads and scheduling. Device integration beyond the normal offer now with applet generation for their own remote controls and personalized app interfaces. Many I know would love to have a cellphone remote to netflix on their tv with audio adjustments, pip, color, stretching aspect ratio, and better intelligence and selection than netflix has now.

As a data provider of user personalization services, most browsers in the market are constantly interrupted by ads at inappropriate times. Offering better integrated advertising with whats going on in the browser, and user's life - mood- etc can provide a more effective e-marketing device where necessary. However, as well, commercialization of products that relate to gaming and the style, genre while linked to electronic upgrades and technology that are specific to their machine is a nice feature.

Sales men once tell of the products that are compatible, and offers are in a specific page that is controlled, organized and made to be selected from. The style should speak of an authentic store while the secureness of it is intensified. Most developers will dedicate time to breaking this since it is an ecommerce responsibility. The best experience says that no breakin is best. Cards such as the google card are nice, but have a third party to provide distribution and maintenance. Offering the cards as a no fee service promotes the brand and supplies in a specific and competing market. The most viable market is by allowing properly placed market goods to be for sale within the browser. Within the one start button, the menu should have an option to load in the card value. Browser gaming card is most likely a good candidate. Secure spending often comes at the price of

handling banking using software integrated. However, banks will want to upgrade their software much more often due to security concerns.

Reverse engineering is a confined process but at times very difficult especially with intel's newer memory protection algorithms for secure memory access. Hopefully a newer bread of economic driver software will occupy the OS interface to manage the credit card as a greyed out payment window pin with confirmation. This gives the opportunity to offer unique usage of a binary blob stored within the machine for an encryption key. Simply when registered as a browser user, both sides are responsible for securing the communication and encryption algorithm selection. At most times algorithms are limited by communication standards meaning that consumer ones have to be crackable.

The surprise is that their competitors list their source code. With specific changes, two modules, they could potentially increase their speed to be one of the fastest in the market - skia - v8. These two products can even be improved, skia for example has many features but little in the way of utilizing those features for application interfaces. As well is too bloated for usage on a users specific device. With the correct dedication to physics modeling, incorporating a system like panda 3d, orge, blender gaming engine, and profiling the system mechanics to reincorporate and design more performance the system, which has not been designed yet. The list of game engine is quite extensive and architecture development is different between each. List of Game Engines has a list of them.

Taking into account steam stores, offloading the production and content engine to the pattern makers such as unity. Part of the api is utilizing the quota management and as a management ment of the specific browser mode, database facilities and sound of the amazing browser effectively. Perhaps a domain category or new protocol could identify the server's ability to host distributed gaming hardware and artificial intelligence centers, game play planners and reusable collective gaming networked resources for world generation, story formats, voice script production, and mixdown audio services.

One of the biggest benefits of modern software os that it works for most people. This came about though the diligence of useless robust testing procedures. Movie and media fields incorporate the acquirement of creative stories to remake and publish using distinct assets. The ability to incorporate critique, improvement, and second hand improvement comes from selection of a specific group of people and focus where their focus is maintained to be scope limited, time based, and tainted with the build procedures of gaming. The asset production system and package system should provide for these quality creations to exist but exist within the realm of the trained providers as a secondary usage. Therefore evaluation and publishing will likely have several types of projects and procedures for composition.

Providing a base of textures and high quality model libraries as a compressed base will achieve very highly rated loading speeds. Along with texture include textures which can be repeatable and also used for blending and shaping interface technology. Image processing with defaults can offer creates more options to choose from when their program runs. Lite keyboard control usually keyboard and other things have lights which are controlled by intercepting keys and controlling pinouts. Provide accessibility controls which may incorporate numeric or simple menu selections. Newer technology utilize high resolution cameras for both gaming and accessibility interaction. Provide a sign on screen which applies a device integrated process. Provide account retrieval and limited account storage.

Terrain based engine and object tessellation profiling after modeling abstraction and build whereby models can be constructed using 2d and three dimensional vector and polygon operations such as union and subtraction. Of notable value is the ability to create objects where the result provides the necessary intersection clipping and shaping bevels. These aspects are typically coded in python because the necessity of using them were made by someone that was

modeling and could write the programs to provide joins of varying styles. The ability to lathe a two dimensions object to provide a faced solid three dimensional object is another useful operation. Control materials, generate height maps using noise functions and integrated realistic repeatable bump map reliefs such as granite and other difficult noise.Metallic surfaces, skin and skin like surfaces should also be producible from the base system.

While typically a thing of yesteryear, every once in a while a game with a story comes along. But of course there are many games and how long a person plans to use them. Maybe it is for ten minutes, which can modify the selection. The books and interactive stories should be supported where a storyboard engine provides game and mechanics but new story data, maps, puzzles, and interactions. By incorporating the science of game play, there are specific areas which make the game playing a success for most such as a goal and reward/accomplishment. So creating types of storyboards where segments are generalized in time but still provide free play is a good solution but should not be a requirement. The most comprehensive performance solution can come from providing a detail generating base that may be expanded by download models as well. Typically AI may be expanded and based integrated for emotional changes and adaptability. Based upon difficulty level settings are often not as granular as necessary.

From the Microsoft Windows interface, the system UI technology is still old while other programs have been updated so a total UI experience with the once pristine comprehension has given way to even better leverages Microsoft can employ by investing in their quality and trademark. With the window system, it is the component architecture that has excelled and caused application ui to become outdated. The edge browser seems smooth as it uses the newest methods of fifth generation design as os integration. And they have the compiler people on board. I always remember the first versions of Java Applets ran the fastest using the microsoft vm. It made my experience of debugging and writing code for it much easier. At the time, most of the byte code engines were not as optimized as they are now. The debugger for it was a step up in hardware from the normal for a few years until java became pronounced and platform adapted. I am always amazed at the durability of interfaces that use Java. A big issue is 100 % standard without leakage to their leverage of browser distribution.

To understand why this happens is not a negative thing however. It is really a mixture of the audience that complains. Most people that care about computing standards are the corporate entities needing it for commercialization of web information systems. As the argument with having a platform filled with extensions is inviting for multimedia target that require the leading edge integration. Plus it locks them out of exterior controls enabling their advancement to transpire.

Solving the issue is very simple and even more agreeable to security concepts of a browser for standard browsing that adhered to all W3C standards with nothing more. And having a game browser with a specialized electronic commerce system that is set and store controlled as a parental or control panel method. That is spending small amounts from a managed electronically liquid fund separate from main banking institutions makes it an attractive feature. While having two code bases, the advancement and progression unbridled in the gaming desktop browser would be significant. While the XBOX will become more trident in its game ai, content availability from the newer content production methods and available marketeers would become indie based.

Compared to the xbox titles, which have more time in them as a unit, the web gl games excel in their simulation features. And there are many more creative ideas. However, they are not as filled with story concepts related to higher game design. Because of platform integration, the unity system is not as polished as a localized c/c++/asm/python scene graph. And this trait echos throughout the highly polished Javascript GL, asmjs selection, to be as fast as it can. But it appears different than games that run local. That is the frame rates are more constant and graphics have more detail.

The firefox web browser has a nice block update that makes the render of the full html page very quick. Typically without wayland, a small section will be noticeable that has a jagged ghost. It is very quick, but you would have to look intensely so see. The page from one to next shows decent page content that is most likely cached, but the render time is significantly improved such that there are no block copy artifacts ever noticeable. That is on both systems you may have measured 700 fps using sdl blitting, but it is still different somehow like the order of the pixels seems to be occurring in more precise synchronization with the oled/lcd/led display technology. The weight of such importance has not been found as continual operation of such loops then have the same upgrades in performance.

The reaction time from the UI occurring as the result of HID device input is viewed as a top trait of system acceptability and performance evaluation by laymen. When linux is used in the new wayland configuration, without xserver, the input utilized arrives from another high level abstraction software module named libinput and evdev. These provide fast generalized interfaces that are file stream based. That is, easily programmed with standard c++ library routines. As well, there is also a python interface for the system module.

The mix of web browser layout engine layout technology by just the language definition alone has achieved a plateau in development publishing productivity and quality. Advanced features are always swarmed with review and notoriety until the lengthy cross browser supports stop having to be introduced as browser specific tags. Its a task whereby one might have spent a length of time writing the descriptive names and making the code readable, in its natural form, to the newly pasted length block of multiline lengthy code.

It is apparent, as with all technology, that data growth has also reached its tolerable visual limit in text quite easily such that the network medium is wasteful compared to other interfaces such as ssh. Text is something that the browser is calibrated to achieve instinctively yet very little emphasis is placed on advanced rendering such as texture, path filling, and general cinema quality animated textual effects. Most film producers and writers would love to sell a seat to such a high definition resolution device for their story. Much of the functionality of the computing device's potential is hampered by both man's limited design scope and engineering's principle desire of efficiency that it is a group muted direction.

The amount of text that is never read on pages is most likely a high statistic. On average, the big blocks of binary media data are what consumes large amounts of bandwidth. Part of the media server's purposeful architecture is quality of service (QOS) for streaming. So with a text buffer clipping provider which also takes into account the multiplex object mixture of game data, less resources would be used initially and second, third trips to network requests for linked media is less. It seems the slowest medium - networking - is not as planned for streaming use as the devices of sixties and seventies were for serialized tunnel access. A stream would be used in conjunction with the preparsed inlined frames of media. This would revolutionize network application usability multiple times over especially for gaming.

Stella, CPM, DOS and 32bit

By virtualizing the machine code from multiple cpu instruction sets, the process of emulating the cpu within the llvm architecture is similar to assigning values to register names that are variables. And handling the logical operations for instruction addresses as jump label positions within llvm without worry of jump style or l2 cache branch prediction on output. So, 32 bit code may be run within the bc virtual machine input. One of the biggest problems is knowing the architecture specific internals such as how exceptions occur at specific areas. This encompases the PE format for windows. As well a perfect disassembler, IDA pro was the last I heard about being perfect. Wine provides these services internally to run programs. It would be nice to sit through about twenty minutes of translation time to have applications run very quick natively as they would in gtk, gtkmm or qt. Browser.

The biggest problem is how the program runs within the environment as tied to an operating system. There are methods of building such a system that initially are internet based that find the applied api call from the dlsym call. DLL, exe, or other format. The parameters are different abi using the registers vs stack, and order for types of languages. These are handled within the c and c++ language with a prefix in the c function declaration that make it accept the style.

By running the program as you step by step study the stub and local tie-ins of the windows api to linux api, applying a direct translation eventually through the llvm with a type of package or translation output. This may be illegal for some software to be accomplished on. But as well offer a type of newer “sandboxing” technology that can progressively improve and upgrade its efficiency on a trust policy. Sandboxing is typically used for items that are going to be depreciated but here it is used as an initial policy to determine if the component is safe to be run as a potential security analysis.

The potential to modify the data storage mechanism to a select and more powerfully integrated device storage or network storage from older ms access, dbf, flat file, paradox, isam to berkeley db, relational database or modern storage exists through analysis.

See Also

- Recompiler

Off scope and Necessary

Compilers produce code that numerically adhere to principles of a standard. That is conversion from smaller types to larger types and vice versa. A few other principles are also applied when mathematical operations between two types occur such as sign conversion precedence within the c++ compiler standard. Data size operation size within the registers are 32bit value on intel for char data when it is indexed as a byte, but this does not affect performance as other methods of string data operations are handled with avx instructions by some libraries extending the local platform. The integer representation uses two's complement for signed numbers and hence binary sign data (bit) is repeated when the width of the field is increased (movsx, movsz). There are also select instructions which operate on types of each variation. Modular math, sign extension, and rounding are applied as well and can be seen within the machine code separately for avx and fpu. There are some discussions about this, however the best way to view these principles are from the object-dump disassembly. However producing optimized assembly is also another world in itself with short cuts and instructions used as per side effects like two math operations in one.

There are some cpu machine instructions that would be handled differently like port commands. They should not be within the decode stream for most programs. Methods such as total emulation can be employed if the output is necessary. Carry, sign extension, zero extension, while skipping some that are irrelevant nop, jumpnear, jump far, specific floating point ops perhaps as the desire to optimize by application domain is necessary. For floating point operations, typically they are intercepted as already optimized. Recognizing the signatures from specific computer language compilers can reimplement an applicable series of floating point operations to provide the same or better output from bc back to native form. But most consumer products utilize the mmx 64bit floating point which may be studied or analyzed for pure input hopefully. Vector floating point functions would be important. Coverage of the OpenGL set of functions is important. The memory addresses encoded within the instructions may be of varying sizes as chosen for various reasons and the size (byte, word, qword) is encoded within the instruction packet.

Decoding the entire instruction set, you see the upgrade path was really a paste of new forms of instructions and recategorized to mean something, the new pin wires. It is a might to live the history upgrade path within the binary is as simple as matching a few bits difference between sets.

It's quite a read, Mod/RM etc at the Intel manuals. Typically the mov instruction is from a location to a register, 64bit while movsd to floating point. The stack frame initialization set by _start is also important for assembly programs calling c++ functions. Expression reduction and even computing ahead of time values that can be are what is expected to be reigning down on you as an asm programmer. One or two good routines perhaps an arguable necessity. Pixel routines are these in some older libraries. Not recommended for productivity but a good study of about three months, of even a year with intel. It was nice seeing byte in formulas on my unfinished jit. Text formulas with all the data types, not an ast based like it should be in the pro version, but a post fix stack to produce linux gas assembly of an expression grouped. The next generation in which I stopped would be division. So that c++ style with gcc style assembly similarities for each data type (short,int,float,double,long double (signed/unsigned)) with conversions M D AS. Division did work but there was some bug I cannot remember now I was lazy with fixing. The output program used a stack allocation width allocation and alignment of 64bit c style program. And c style expression conversion rules applied. And in the process of linking it up with the std library. When encoding the instruction was the leap in research, as all of my projects go.

So, after a quick study of the machine code mecaniques, without a thorough understanding of the intel manual meaning, I coded an implementation that provided the mov instruction, add, and basic math instruction, and a method of instruction variation study by program generation to have a system that is completely data storage based with instruction indexed and parameter shifted in. The templeOS does this shifting for register reference for its c style jit but it is a very simplified model. The instruction set of my jit is compressed and stored in byte form as a header file with a memory buffer object interface. The object's methods are the same instructions. About two percent of the intel instruction set.

The best jit formats are also components such as java oracle jvm, v8 javascript, byak, asmjit, nanojit, luajit, and font technologies which implement the newest instructions and complex options of branching on the local processor. Typically the stack, memory, calling in both intel and risc plus several more are often supported. The input mechanism for each of these is as a layer model whereby the language or platform has an input, an api library of routines that are used for data types of programing language/os features, parsers, communications, and platform. The best integration branching points for most are as a unit whole because of the maintenance value. Byak is just a header file, but can be used in specific areas where upgrade is not necessary. Pixel, Raster, and Font technologies. LLVM is nice to study and has a lot of sourcecode. GCCJit is also an applicable technology but different to use.

I noticed that the 32, and 64 bit mmx registers are handled differently than the stack based fpu 80 on intel for different compilers, gcc vs clang. That is llvm has further expression evaluation to remove floating point stack operations by using different instructions and stack order. While gcc has a few instructions more. As well the reduction of some operations depending upon the numeric type, signed or unsigned, are applied using modular math amounts such that conversion from floating points of different sizes to integer are bound to integer domain bit sizes. LLVM emits the instruction PUNPCKLDQ while GCC uses a different method. The MMX Assembly instructions were used to a small degree better with LLVM. There are a few magic numbers encoded within the machine code to achieve modular conversion from 80 bit floating point to integer. At most times, it is spread as part of the instruction encoding.

After seeing the risc instruction set, it is encoded much easier but offers less potential advances in distributed hardware. But intel still does not have this yet either. That is instructions just take longer within the cycle click rather than happening as part of a distributed system request. Providing a virtualized set that is at a distinct level where optimization can occur on modern platforms is BC. I may have a misunderstanding, but it seems the JavaByte code is too granular and generalized in areas of vectorization. I am unsure if it produces vectors of parallel operations for floating point, but BC is geared for this by instruction. The oracle JVM does produce very optimized machine code. I have looked at and studied many of the JIT as well as assemblers, YASM, BYACK, asmJit and v8. I think the best formed and provider is the v8. Way down in the system is the intel 64bit jit, great model, it used to be the sun implementation however the architecture changed. It still looks

readable but less generalized in the x86_64 intel encoder as some functions of numeric processing are inlined. The shuffle ones I am still unsure how they are used. The source trunk for the back end cpu encoders are here. The layer above is in use by web assembly and provides nurturing such as applying spill bound register selection and abi application to the local platform.

Plus the jit has different levels to provide formatting, like optimized numeric and date formatting. Working with the base data types, the engine provides a layer for a math library for intel - ieee754 intel. These routines are mapped and invoked from the assembly. Can they be changed for even faster, Making sure that fast math routines are used. So the engine could be adopted to also have a module for specific vector communication types of open gl structures while supplying tuned routines for javascript. At A technique that lua scripting uses more regularly but with a more in-lined approach at time of machine code emission.

The major points of platform and language integration seem that they may be linked to the builtins-call-gen.cc. Here is where the call sites for c++ does not use boxed arrays and has a const ref ptr with compile time checks so data is not copied. I am unsure how often the information is copied for boxing and unboxing data, even numeric doubles. Most likely they have selected a code path for threeD.js to choose one that does not. However as well, the parameters of most function calls of a GCC or LLVM within the 64bit abi utilize the register directly for the parameter passing. Inside the routine it utilizes the same register thus accepting it as a routine variable. Most likely the javascript engine provides the same level.

Wasm has its own abstraction of assembler opcodes. A series of memory storage, conversion functions, its own data vectorization, container types, and it is strongly typed to size for platform native usage. That is it provides all the base data sizes and types. The level of assembly is not as modular as BC however. To create proper high performance JavaScript llvm bc, It is a newer invention. While BC can still house the web assembly, there are many functions that appear to be state related to floating point hardware such as the simd instruction inclusion. Where BC is a much higher level and has the vector data type.

v8 dev and the source code is located at v8 github source. The tree and names are well organized. Platform has items that are threading based and some utilities. Uses pthreads and threading directly and not std. Are bit fields incorporated to use instructions that test for bit like BT or xor? It would be interesting to set up a test where the assembly from two decent programs, or several compared. Analysis might find the points of optimization where certain steps, or other peep holed optimizations are evaluated.

Functions and modules provide the implementation as the templated oriented c++ assembly produced by full compiling would be noticeably different as its array handling and data type from std are template based. Lua provides this type of capability in the way the production engine works, but it is not an inline template either but its numerical processing is. They may have changed to llvm, I am unsure. At this level of data and hardware programming from a high level api is difficult to provide, and JavaScript functions fairly. However, providing this functionality using a template processor for data containers makes legitimate hardware communication and data traversal more efficient.

There are questions as to where to integrate the gaming and base platform. Code duplication in this arena is not a good choice with the code bank already being huge. Since the most new additions are the video processing, perhaps using a linkage to a module of cross platform c++ but compiled natively would be to utilize the apparent functions of the macro assembler.

A fine measurement can be seen on Agner Fog's website agner.org. Optimizing Assembly and micro-architecture contain important information that is useful in not translating various aspects of optimized assembly to provide functionality related to numerical processing. Or non benefitting loop input such as branch prediction would be filtered out. These aspects of the machine code would be produced more specifically for the device being used. As well, the binary will not contain code that is not within the realm of the executing cpu model.

I decided to go with LLVM as my path and pull from existing language input such as c++17-21, python, rust, haskel. It is necessary to have each of these operating with the same llvm. GCC is used for kernel building primarily due to some code not working with clang. As well, optimizing at -O2 vs -O3 on varied modules with intermingled clang gcc compiling is what I desire. Probably can happen all by one emerge command to rebuild after setting up the gcc fallback to kick in whenever the clang issues an error.

IDE Desktop File Browser Integration

There are so many features that are in line to be designed. For example, would it be nice to have built within system integration a common framework for file system manager and its many forms of application usage such as file name selection, file choosing, directory path, etc. That is the common file dialogs handled system wide.

Integration of development functions contextual by file and directory provides a free style but integrated approach to project management. The selection of editor is nimble process and often many plugin enhancements per file and project. Most editors I have seen are heavy weights even though they claim to be light. Atom, Eclipse, VSCode, Geany is a little lighter but not as integrated for debugging. So, it seems that the component orient approach works but little has been done for system wide components that are native plugable smart gui components.

IDE File Window

As a system user and developer, the IDE integration within the file manager flow is another key idea that is inviting. Building, project contextual operations, and syntax editor inline from the file dialog is nice. The project file is morphed from the file manager browsing window. Or program doxygen and lint analysis tools, hyper linked code paths, and analysis of code cross application domain as seen by the index feature is a place where the clang parser may be useful as input into the system index. Of course as well as IDE and other aspects of file based usage. Ohh, how about a git diff mode that shows within the software window of the file manager.

Ad Hoc precompiler

Ad hoc precompiler without the process of formalizing the post preprocessor. Build language additions that provide enhancements to c++ such that the language and programming features are preprocessed to provide benefits of multimedia usage and gui configuration. Data input for nested data structures. Sloppy input correction with suggested corrections.

Databases are typical inclusion for this as embedded sql. Provide language bindings where tooling would be required to run programs in different languages. And provide module data passage with the c++ std namespace allowed for data communication. Google protobuf is a type of system that allows this but is instantiated due to the simple fact that the software base has an unknown internal communication std compiled in version. Gentoo and source level building system do not have this problem but provide it through native compile and link options. The cstd++ library is easily

controlled but the template library which is inlined has to be controlled and changes between compiler versions and compiler for all platforms. This is the reason that glib, gio and the linkage to the scripting boxing data systems exist. Provide automated packaging and build system integration for developers to receive the package. If it is a library, provide usage.

Diagramming, and wysiwyg within the window

Documentation, books, pdf, latex, man pages, saved material information from stack overflow, diagramming, chm, docbook, c style commenting, external file linking/ internal system documentation, boost gdk docs, doxygen, xml commenting and markdown (md) are methods of providing documentation. However integration within the IDE is often not very well supported. Typically the highest level of support is syntax highlighting. The most advanced ide support provides hover for documentation that is stored on the system. Context sensitive help is differentiated between visual studio, code blocks, vscode, eclipse, and even vi. Vi supports a headless eclipse in a text mode, but I feel it is very difficult to use. Not at all like the Borland Turbo Pascal 3.0, just an editor, and compiler. But after adjustment it is usable. Most advancements are held within the GUI context switching ability of copy paste formatting with multiple selection methods easily accessed by mouse, keyboard, touch. Perhaps would be served with gestures of shaking the cursor and then outlining the textual dimensions.

The syntax highlighting serves a fine purpose of logical separations, while some modern open source forms lack extensive documentation and rely on conceptual names for teaching. That is more self documenting code. However better font usage with mixed font usage, video, diagramming, and iconic indicators for routines.

The use of proportional spaced fonts in conjunction with publishing tools that provide externalized file management and automatic publishing are well sought in the industry. Perhaps at some points, diagramming may be structured such that it is at the file level by standards. However other uses and third party integrations may provide usability in routine maintenance.

Imagine a routine with a type of maintenance where data tables could be edited with a row and column editor. This is a standard practice for engine type programming. The editor provides the aspects where auto insertion of “”, and items like that occur but on a protected cell editor style with the appropriate c++ coding helpers involved.

Indicators are very nice within the eclipse IDE such as information and warning indicators. The information is within the editor and readable. However the integration is not as seamless as some times longer messages are not text wrapped. And there is no controlling the font. But there are further uses to the minification icons have with the unit concepts routines and modules have.

Indicators may be served well in the display of coding analysis suggestions. Linting and statistical profiling analysis views that consider the view from the routines perspective. Providing a tabbed view that is well blended with the textual mode of editor to distinct documentation and maintenance tracking units along with git commenting blocks.

Extensive writing tools as an inlined viewport scrolling window that also pops out to a publishing interface, provides the render indexing and editing of the object oriented commenting as part of the development environment. Within this scheme, placing HTML and CSS can be enabled as an externalized file tag from within the current source stream comment blocks is a good way to implement without burdening the source file but providing the visualization and gui framework requested within the documentation header for the function or module.

Providing integrated cross platform building that is linked within the same way, a source file from the project, yet is tailored with a gui for package confiruage as well as having well as having build system support will be well liked and increase productivity. Like a gui install for source libraries that are within the ide. Package config is how this is done in a particular system library way for linux. However, there are other methods which are integrated with the build system.

Facilities, or module level component controls can be a summarized and glued together project control panel where modules and their usage may be known about from other libraries. Module level and object support from separate binaries is often an advanced feature.

Providing drag and drop component software interfaces that is cross platform should be a reality already. There are libraries that support this, yet development level knowledge is required to make extensive use of the facility. As well, some interfaces use a generate secondary source format to achieve the handling f dialog and help title information for the methods and attributes. That is the javascript and box types operate in a fashion where they are named attributes, properties and methods. The glib system provides methods to automate and save as a group settings to these as well as standard methods for accessing and control. However at times these API calls do dirty the source. Providing facilities to organize this with dynamic modules and scripting would be a good benefit.

The ide scripting sits atop meson. Provide large scale building tolerances such as generating os builds, gentoo for example. provide usb flashing using etcher or dd most likely. true type fonts for commenting and book editing style comment blocks that fly out into books. Print epub materials, and pdf. Integrate the ide so that it may be broken apart within the context of the file browsing. Minimialization.

One issue is having options and then having many options. The user should have a common interface to control the groups of options, toolbars and menus for the application integration experience - visibility. And if not in primary - secondary organization. An option is providing this from the system configuration from the hamburger options that is sorted within the tree depth by usage. Meaning that it is statically positioned but undetermined at this time the position within the UI.

lldb symbol insertion

Debugging with lldb as well as gdb. lldb currently has little support for its feature rich debugging. There is a binary interface, a textual interface, and a python interface. It is also much faster in its stepping process and finds easily variable information. There are no guis, but it has a tui interface. For large programs, the monospace font makes it difficult to really debug as well as the eclipse debugger. The formatting and structure dumping is also robust.

AI Assistive Debugging and Unit Test Iteration

An intelligent debugging system that arrives to same point using the taught mechanism.

So ai can be used with direction in unit testing. Most likely, at the same level of monitoring and guidance, learn to correct bugs within systems much faster.

AI Code Analysis

Once the intelligence of the system provides enough, analysis of the complete tool chain to find where algorithms and processes are repeated and assemble to code to used shared code. While at times, maintenance value would change, the performance provided by such abstraction and order would offer benefits of using the fastest algorithms to solve problems of searching and data indexing. As per LLVM and historical usage, the types of algorithms chosen may be based on data amounts. The compactions of the human form of a kernel to minification and code reduction, algorithm search sharing, and other fashioned operations. Most likely the biggest benefits would still come from manmade or perceived generalizations of struction.

From Windows Message Queue, MFC and GTK to devux TUI/GUI Object multicast visualization

For example, consider all of the GUI code and mfc c++, windows, gtk etc. If a program could remove all of the dependencies of the system and provide an object which methods are the menu and operations within the program, it would be very valuable. The program could be ran as a cgi service inside a web server like nginx or apache. That means every program would be capable of running natively on machines where the hardware was handled correctly if it is a program that need such access. A program directed to study the mechanics of a format it knew about, such as finding the message queue. A clang parse gives the ability to achieve things like this. Code that may analyze systems written previously and update them to newer standards of architecture and style.

DSP MIDI Composition Server

Networked

Musical accompaniment is often a selective and proposed process that can be tailored, however often too weighty to be placed on the client playback for real-time playback, integrated midi controller and sequencing software can drain a plethora of cash your way because people want to play games and people want to play music. The ability to encapsulate the processing power necessary for VST management on a professional level comes at the price of large hardware because the expressiveness always grows infinitely. Ultimately other processes in music are used call track bouncing or merging, mixing and rendering are separate functions. The interface is completely parameterized. Midi compression and real-time playback would is not possible. However their are very well designed minification providers to dsp algorithms and sound card capabilities to provide a place holder market and at most times a decent instrument rendering. It is only when multiple layers are provided such as chaining that multiple synchronizations in real time processing have to occur.

Most of the DAWs I have used are very wasteful in the underlying technology platform in that the systems are so large to write, interface, dsp side and feature side, that integration within a workflow is deficient. One of the best methods is bouncing, mixdowns and temporary mixdowns are typically not supported in the sequencer. The sequencer software has to be placed in the render most to get an editable audio signal. Therefore, the data and time the creative element arrives within the user's workspace allows for such processing to be distributed. Utilizing the browsers communication resources. There are moments when the state and signal would be changed, but over all the midi output, vst parameters, audiomation shape generators and midi controller input would be transfer and able to be monitored, modified on an incremental basis. While focus, or track, is typically the mini milestone for some music formats.

The ability for the state to be compressed both in lossless and lossy tagged midi provides a select number of midi information measurements to be transferred perhaps using a type of numerical range scaling. Just audio compressed can be very specific and as a mixed resource arriving from

the server, using intelligently the storage and playback capabilities of the local dsp setup is the most useful. Such that resource usage is minified by focus such as track operations, effect multiple changes, note taking while rest of system is queue of local mix play using the newest best near real-time downstream mixer. Technology for these types of playback facilities are greatly increased by offloading the mathematical wave production and utilizing the adsr and wave table pcm mixing solution.

Local Music Web Server Box

Controlled Wireless High speed

Another solution that is appropriate and durable for musicians working is a local web server that integrates the tasks such as out-line above for the necessity of powerful hardware local. This would most likely be durable and be a local served by minimal points when roaming away from the local server. So it has a wifi-card that is high speed not relying on any hosting provider but is a provider of these communication services via an integrated modem. While technology is limited on the provider side (internet service provider) as far as roll out speed, it is not on the local device services. The hardware and ability for local near field communication is legal and abundantly available.

The impressive differences between the large scale normalized and powered motherboards ultimately are geared with high frame rate producing cards. However, for audio devices, the field of visualizations is more 2d graph ranges and measurement specific dials so the display does not have to be three dimensional but more supportive of graph visualization real time. The box does not have to project an interface at all, of only gloating lcd lights in a small window.

On the client editing device, A 2d renderer is suited for optimization however most modern systems for network interoperability provide the unnecessary vectorization system for rendering. For embedded devices, much the same way icons are compiled for modern desktop interfaces, the appropriate resources are provided for visual resolutions dedicated to the terminal's capability. Thankfully modern monitors support two standard resolutions, while likened to televisions. For integrated OS interfaces projection of interface is organized to be readable because the resources and control methods at time also provide motivation.

As an audio engineered OS for DSP local server with an acceptable client mixing technology to utilize fully the playing devices resources for midi input as an instrument increases the portability of the soundbank portability and quality. As a box to be transported for musical equipment inclusion, the armor and connection setup has to be different, more durable spill resistant, self component part repairable / long term usage and investment policies known for practice and performance use. Five years would be the expected shelf life of the audio server. It sounds like a large investment in hardware but it is really cutting the bottom end out, so a machine build with a slender double cooled vented and very dust proof case. Leather/ vinyl/ pvc exterior with thick plastic polymer on inside. Metal carriage cage with luggage handle. Ports consisting of quality plugs typically female, extendable, replaceable and maintainable for audio, sound, lights, video, balance, audio level, panic, network LED and phone applet tie in as a product developed by publishing to the device for a secure connection for a show.

Speaking Computer Speech Advances

VST Music Hardware and DSP sound Coprocessor

As a user of VST, I know currently some of the simple functions of the wave production and how buffers are gathered in chunks to mix. Although not familiar technically, I perceive that both vst wave production technology, ffwt, and the production of sound be midi based while effect processing be a distinct modular function unit of a multi core audio processor. Can audio DSP be productive as a dedicated multicore processor? Yes and audio deserves dedication as well as video.

In some circumstances having a sound generator geared towards parameterized production of gaming objects to add dynamic realisms is very acceptable for quality. With this technology there have been numerous chances such as the atari 2600, ti 99, sinclair z80, apple IIe, etc which had fm synthesis built in which operated on a port basis rather than wave form production through main cpu instructions. A gaming audio engine.

Voice Print Audio Text To Speech with Emotion and Realism

Voice print emotional speaking has intensified in twenty years time, as much of an inclusion to what is said of real technology. However, the last ten years have only been hampered by processing limitations concerning the model of MRI based data. While resolution reduction into a working horn flow system while using physics modeling on polygon reduced data should be comparable to the same wave technology used in ray tracing. Basic material reflectivity, depth and then mixing.

By animating the sound production parts to be alive with synthetic breath, the cavity, tongue, and rubberized inner texture of the mouth can be simulated. The cords, tension, and other aspects generated as a type of model generation. However in current implementations, the production usually is provided by a severe approximation and often unrelated way.

The chain might be better approximated by providing summarized motion path data and other data relationships suited for packetized production perhaps, but often the best quality will be had by applying the full model first. Articulatory MRI based speech production technologies is a field google search term for related research, a PDF about these techniques.

Further DAW Advancements

The integration of raw midi data packets as midi is designed seems that it is encompassing and includes PCM sample data processing. Yet the typical passage of PCM data is from SF2 and WAV within PC based DAW software. Automation data can be lengthy as a function of time so typically syncopathic and quantized to smooth shapes can be a product of the midi synthesizer processor on board as well to provide an accompanying gathering of technology used in music production. As a co processor within modern desktops it destiny is on the motherboard to be used by the suite of existing and future software.

As a SOC chip, with intel or amd style video, the necessity of a provisioned instrument can be manufactured. The least faulty times are often had for instruments of this type to be dedicated software and kernel bases. The hardware driver interfaces as well as the style of audio processing DAC and memory mixing is accomplished using a different sound card driver. FLAC and other lossless codecs as well as functional lossless are included within the audio processor.

Application API as base platforms for musical DAWs are often best suited for embedded designs and predetermined resource allotments. Has a built in digital amplifier with 15 watts of output. A typical usage would be production studio where that’s a lot of power. Standard audio outs, 1/4 I /r and sub crossover. 12 gages spring clips for home consumer connection form on board active powered amp (+ - stereo + sub) clamps.

For low amp standard output there is optical, bluetooth, usb, rca, built in qwerty full size print as jelly comb. Hard case but without a LCD screen. HDMI projector onboard that projects audio visualizations. And it offers wireless midi controllers. At most times the midi controllers such as the keyboard have the interfaces that may display information as part of the computer display.

A HDMI connector and monitor sold separately provides the musical composition desktop. While in certain modes can be used for live performances but is used primarily as an advanced programmable sequencing DAW. Often the parameters and edits are used as loops within other sequences during play. The large amount of interface editing within the VST, LV2, DSSI, and newer technologies occurring with lua would be fine.

I often thought about writing one, The operations seem rather simple in the fact they modify data in a linear rate based fashion. Linear meaning that it is sequential usually related to previous and latter values by small amounts. The overall production while distributed in audible perceptible meaning can be consolidated to form a vector of processing within a continuous loop whereby a cycle is formed.

The greatest benefits in real time production are often measured by seemingly little but highly noticeable because it is continuously looped. Each unit in form firing events and queried for chunked audio data. The operation of the chain becomes adapted in user interface space to be a modular element or object based equipment rack. Often the granularity of an audio module is obscured within the synthesizer technology, but in the common user domain it is perceived as elements such as reverb, flanger, echo, multiband compressor, eq, and limiter. The version often considers both sound type, style, number of parameters, and quality.

Often never achieving the total vision without a dedicated focus, the necessity for a large projection of a DAW that is suited for work has to have a dedicated GUI desktop. A select type of software and none accepting of other various and destabilizing formats. The promise of wireless, browser, writing tools, syndicated and generalized membership publication from modern web audio provides companionship for the producer and dj.

Intelligent server based audio production of remote music equipment and hosted environments.

Opens the desktop market with newer interface inventions while providing a uniquely and economically efficient device interface. The ability for customers to other devices that are preprogrammed of size, quality, vinyl coatings, is also a major part of the market. In the next five years, the flexible oled technology will also provide amulets that wrapp on the wrist, transmitting jewelry perhaps.

gentoo-desktop Build with rc-service

Intel i3-8145U 12GB 256gb-nvme-1.4MBS 1TB-5200rpm-250MBS 128GB-120MBS 1TB-10MBS 8x32GB-10MBS 4x8GB-10MBS

With a gentoo build rc-service, it's bolted onto your hardware precisely and mostly compiled with too much code. But chrome starts in under a second as well as many software. While clear linux is

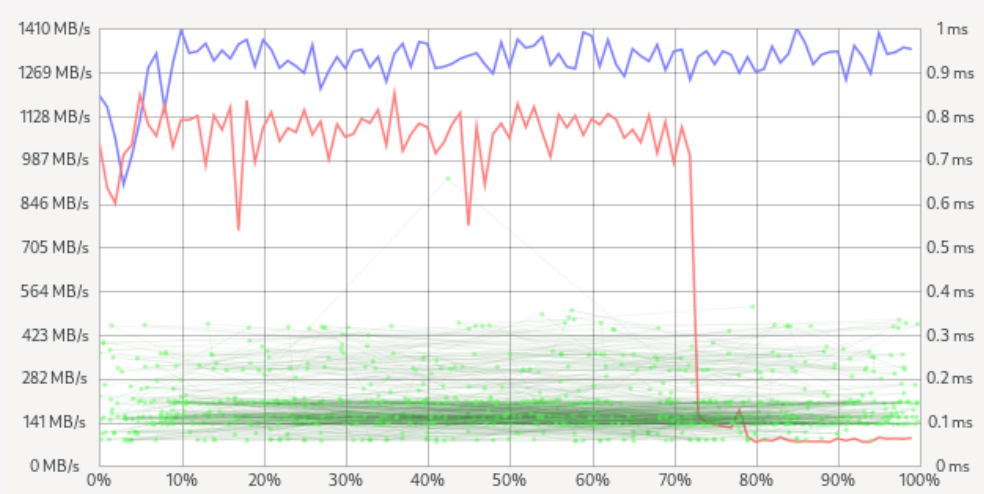
less than two to three seconds. Clear linux and gentoo are gnome desktop environments. But LXqt on lubtuntu boots faster, yet lags in video support tear free. Gentoo is the low latency high cpu performer. gnome Battery, not as good as lxqt lubuntu battery.

question: Is there is something you should do with nvme drive? , but after formatting and partitioning, deleting, partitioning, moving, reordering, and rebooting loaders, efi resets, mistags, formats my nvme runs at about 1.4. I saw the numbers 2.4 on clear linux but the numbers changed. I am unsure if there is some command to run. I switched to btrfs for my gentoo partitions and set the compression level to 6 and 4 for my development and audio environments. btrfs seems the best overall performer for software and system usage. That is you are loading blocks of data and writing stuff. The system has a great memory queue which makes writing so fast. How big is a file you are editing, its already in memory too, so it becomes a compressed stream to the nvme. Something that will be built into nvme drives of the future. The algorithms are performance tuned with drop off block segments for the numerical ranges of 1-15 for zstd. zstd is the btrfs optimized file system that seems to outperform zlib. Quick is inusage and save a bunch of space where source code and binary exist due to its adaptable compression scheme. Even items that cannot be compressed at the risk of performance are placed into the pile of non compressible. These files are already compressed and unique binaries. So about 64 percent of a base system install is uncompressed. Yet when you start adding source code and user documents, those are compressed automatically by what they are detailing. You can tell that the memory buffer is blown at about 70% of the testing cycle. While the read decompression is quick and only takes a small percentage of the drive output capacity, that while you are using the system. When the system boots, its about a second and a half longer or a little more from showing the boot up as the large series of libraries are loaded all at once from decompression. Worth the savings on such little time when most likely you are still unloading the mouse or setting up the pad. Thats only if it is being rebooted as most time it is plugged in and turning on from a standby.

lenovo /home/anthony # compsize /run/media/anthony/gentoo-primary

Processed 617996 files, 355651 regular extents (355677 refs), 353546 inline.

Type	Perc	Disk Usage	Uncompressed	Referenced
TOTAL	64%	16G	25G	25G
none	100%	10G	10G	10G
zstd	40%	6.0G	14G	14G



Disk or Device	Partition 4 of 256 GB Disk — JAJP600M256GB (/dev/nvme0n1p4)
Last Benchmarked	Fri 22 Jan 2021 10:49:06 AM MST (Less than a minute ago)
Sample Size	10.0 MiB (10,485,760 bytes)
Average Read Rate	1.3 GB/s (100 samples)
Average Write Rate	805.3 MB/s (100 samples)
Average Access Time	0.14 msec (1000 samples)

Quick at block level loading those huge files of program startup. It seems that the system and directory node caching with all of the b tree stuff pays off in the long run. The task switching it seems that google chrome freezes in gentoo for a period of time when video is playing in several windows. This seemed to change behavior from a previous instance of gentoo, so I think something messed up in the install-compile. I was impressed with the newest wayland version of firefox and crazy games performance, not lagging at all behind. However it seems that in most cases the chrome v8 jit produces the fastest experience when gaming.

Gentoo Audio Building and System setup

Setting up the audio environment with gentoo is a long list of compiling setup. Multilib should be enabled as well as the ming-64, library. Sound processing libraries, software, wine, and other things carla, lmms, calf. The great thing about these builds is that they are much shorter, less than ten minutes. While the OS build can be days, weeks, months to compile. Still working on audio build and cross windows build system. Thats producing windows binaries on linux, running windows software on linux, compiling code while running in a windows environment to run in emulation in linux, and sound calf studio processing. Kernel scheduler and those little glitches. Changing the screen brightness via the control panel.

Another goal in the future after finishing some more interesting items would be to re optimize the system as well by building, recompressing the system build backup, clean unnecessary builds, rebuild for two kernel types via chroot. The build default allowing applications to be optimized at O3 would be a starting testing point. And then selecting kernel modules for O3 and then comparing the resulting differences in binary versus source code output intermingled listing for an overview to see what types of changes are made. The important functions are knowing the libraries and architecturally how the functions are used within the context of the system.

Desktop integration of audio files, audio channel mixing application as well as beat bassline and other facets of the file browser based sample and midi recording process would be nice morphed in from the file browser. A sequencer drop down and quick doodling setup for creating tracks and beats without leaving the file browser. Text input as well, “X X X X X”, that a good high hat hit or a bass bd drum lesson for free.

Musical track indexing, midi indexing and preload rendering. Instrument mixup quick selection file browser ui.

Gentoo and Qualcomm Snap Dragon Portable Designs

Gentoo is relaxation after the build. A sound piece of mind from developers. I have never had so many successful compiles in a row - look at her go like a proud princess. Everything you install is a natively linked compile. But you must specify that the kernel be compiled nativel as well, choosing the right kernel parameters can be tedious. But a learning process.

The keyboard. Half inch rubber folded and stiff enough in the lap. Next gen gelly keyboard. The laptop offers a mat workspace that protects the screen fhd oled. The keyboard attaches to the CPU nvme pcie 2tb 32gb snapdragon 10core with a next generation video system. 2d accelerated low power up rendering with true type accelerated vector filling, texture mapping, and matrix stack glyph. The display processor supports intelligent rendering of layout upon the 2d and 3d plane through multiple visualization and computation GPU cpu hybrid running in a risc architecture. The shared memory provides less data movement while interrupt style callbacks schedule populating the circular world buffer when camera changes. Onboard physics emulation and collision detection when desired.

The 2d and ui composition are instantiated by not the means of a video buffer with planes of raw video rgb. Rather the signal is controlled through chaining of data structures similarly composed as a risc instruction. That is identification, parameters, and instructions as the same b8size. The parameters offer the memory locations of the data. Text, font, xy locations by numeric. Float or int unknown. Browser translates to new screen driver format.

Memory locking and access provided at driver level such as vsync yet operates on a parameter shared memory access request as a new type of memory protection. As well the screen program which is located at the video memory proper. The ports of the video card may be changed for backwards b800, a000, a000 text ttf scaling mode, and a000 new GPU mode.

Legacy bios such as 8x8 text bitmap and von Neuman boot up obsolete. New bios has 128 mb reserved on nvme and has access to llvm. A distinct feature is that the processor can utilize bc directly without translation. That is risc and bc combined. All soc with male input usb d.

To find the type of hardware I want, I would have to live in the tomorrow. But, I will be happy with a new Lenovo mobile workstation. But the process of invention allows other markets that are subdued in financial backing that they provide research rather than reliance. And of course research according to a team dedicated in competing in a market that is dominated by other chips for major computing devices.

The Qualcomm snapdragon series with numerous operating cores and greater power savings enable along with screen technology to achieve day hour rates. That's 24 uptime. Like most technologies, the progressions take multiple paths. And with bios of IBM compatible machine and architecture filled with years of legacy upgrades, the time has never been better to make the next gen leap.

Gentoo build. Wayland, firewayland touch desktop. Next generation thematic design for video desktop system supporting high energy minimal state iconic defaulting to magnifying focus floating thumbing browse w preview.

Usefulness Decided during The Life experience of User

A better thousand uses for the cellphone, tablet, laptop, network, and smartwatch tie in. The commonality and portable nature of each device offers a selection weight to it when the user is progressing their day. The ability to adaptively generate connections between these devices to provide remote functionality and information carriny is nice. The usage must also be built within the projected device interface. Such that it offers convenience in reading, long usage, and quick placing it into the pocket. So in modern terms this means that cell phone app production or swift generation and linkage can be a part of server and local OS production. That is, the interface matches across devices and has the same performance enhancements.

An eInk touch remote mouse that works with devices in the capacity of a remote for many devices tv, phone, speaker, and force feedback haptic can ring intelligible life style devices within the users every move and provide home automation. An onboard mic for controlled computer voice recognition. Most likely a solid rubber coated device that is void of finger prints but may display in its form. Wheel controls a scrolled menu on the direction found within the control to simulation a context change for a specific device.

For channel watching this can be a tool where the primary interface is by using the wheel and the two mouse buttons as a clicking input. Most often the distance of the television is great while the channel surfing interface provides hard to read selection. In this circumstance integration of the mouse remote and its display would be fine for a detailed small set of functions on the mouse eInk display. Packages and linking to user devices provide the capability to use the clear bottom surface an other clickable areas. When it is sitting there, it offers the finger scroll control and tap. The integration of functions change their display when the mouse is not within the palm.

That jelly comb smooth, slender and tender touch mouse, feels like a bar of bath soap. The rgb edge for tron single curves but with less lines, keeping the same curves as my model. The wheel glows but the dpi is gathered from the fine slit between soft click buttons. Perhaps it can smile and reflect the beat of the music or of its personal mouse voice. The impending click had as a rampant undertaking of neurological expression only derelict of the next one. Controlled.

Robotic Mouse Find You Or Shouts for a lift

The motorized terrain climbing robotic version will travel seemingly mountains to come to you for a click. A new type of input and short term display companion, the mouse will be a friend to introduce as small talk with something to say new each time about the person it sees.

Often the varied digit communication of a touch device is as not as such as a soft determined click. Through a keypad that can be tossed about with a small one line lcd, mic input for name recognition and ocr scanner for business scan as a digital camera and a mouse, what fun the three inch device will be. Most importantly, artificial intelligence is not atominious but more as an

assistant robotic technology that is computed and processed by the user's home system. This makes it much more expandible such as google's platform.

Hopefully the cat will not get involved with the robotic recharging station. The ability for a docking station and transforming docking station technology that gathers collectively devices used together will spring to life and assemble themselves, travel over and morph into the desired workspace. A desk space with elevated keyboard, mouse. Or midi piano, gamepad, drum pad, with sub on wheel barrel rollers. And of course the walking talking speaker with face and audio expression lights. It walks on five legs and can dance.

Entertainment Audio Visualization System

Most people consider speakers and the ingredients involved in production of sound. As also a dimly lit individual, some knowledge is found through experience in the audio domain with crossovers and basic cabinet design. Through the years of modern living the texture of things become suited to evaluate as naturally wave forms become attached to our tactile input. The rubbing, grinding, submerging, burning and other aspects which sound is manufactured from the inert, become a part of our processing nature specifically with our digits and our understanding of material composition for predictability of air wave disturbances.

Stephen King and the sharp teeth of that clown from "It" springs eternally of a balloon being pinched. We are more heart filled in our approach not lending a hand to Freddy Kruger either, in preserving the balloon's exterior architecture. We know that the size of the balloon matters, the pressure of the balloon. And just blinking, intermittently before the pleasure of air compressed escaping its happy home in shrieked terror. That valve play and burning rubber fog horn squall can be transformed into a great instrument also. The left hand operates a defunct apparatus like a single fluted bagpipe. And with the grit of the teeth, and continually increased strength grip, pow, and kablooie - latex splinters dabbled in a peaceful shower on the wood floor.

Audio placement within the sound stage as a three dimensional space occupying object can be applied but often decreases the signal strength and clarity precision such that leveling compression should also be applied to provide the same amount of detail. Most often these functions are not as detailed but integrated within the sound systems of game engines as the scene graph elements provide the storage of object coordinates. However material composition and reflectance are often not detailed or accounted for. Speed, sling slot direction (trajectory) account for types of pitch attuneinations often found in the real world as a train passing.

With integration of these systems as a complete SOC chip and a much larger application memory cache for program space instruction will yield tremendous amounts of heat that are appropriately cooled and designed. The multimedia controller is very sophisticated yet operated as a unit or combinationation of systems on an integrated timer basis. The system produces audio, video textured as modern intel, amd, nvidia, elevated 3d audio with motion, 3d processor has integrated motion path generation, compression playback and real-time control. Haptic bass and music accompaniments. IKenetic skeletal physics and parameterized environmental object based physics simulator math coprocessor. Haptic technology output for multimedia experiences haptic furniture, haptic clothing, and the vibrating movie sombrero.

Midi, and audio track visualization expression studio provides a fad free way of syncronizing automation of midi with objects and their intelligent ai animation. As a form of communication, once video accompanied music, the bipedal forms sprang to life to show their workings of instruments studied. Media diction provided more artistic inventions found and ever progressing. While knowing of the wave form being produced and visualized as a line graph is common form, and

brightly changing as it is inclusive of the entire signal, some visualization find attributed of the music by sound spectrum frequency domain. The beat.

As part of the studio, a more adaptive form of the algorithm which indexes the midi instructions with the audio chunk analysis to a component based interpreter provides the intuition to gui design to align distinct musical operations with boundaries of the object meaning within the field of view. Providing high quality server rendering while medium preview editing buildup in real-time, the musical visualization suite is suited for the monkey glue one click operation wizard for generates and uses a knowledge based artification intelligence system to deduce musical operations and sound acoustics to tell a story. A story of the song in which is generally directed by image input or video input. Input clicks of seconds, the motion, frames syncopated and used as textures within the story. Offering a poetic text word style input of artistic direction which each word can be numerical weighted providing the easy connectivity to modular function type by word to produce a story.

YickYak music art studio provide intelligent video story board output for musicians using an integrated commercialization where designers may work together remotely by networking instruments together through the internet.

User's Local Cloud Publishing

integrated applications for common functions. Archival, git, remote data drives, device syncing and mtp. As well, several devices operate with information to form the user's data cloud held within their devices, allowing for storage of sensitive information within their home environment only.

Boot Loader

The device does have a customizable boot loader which is at a layer which the user may control the startup mode. However, as a permanent fixture is etched, the flow of device initialization and memory state, programs and kernel will be image base loaded. That is, the device also provides a staged architecture approach where consumed memory is accounted for within the freezer state memory build. Organize the footprint for compression and NZIP nvme drive dedicated partition for device freezer state. Ahh, an inexpensive 64gb should hold the base os, and 30 complete system freezer states. This is not the user drive.

HMM Voice Recognition

An exciting feature is providing the newest hmm voice recognition as a dedicated system component so that context and operating parameters of the desktop are controlled per use and desire. A full screen guidance. It seems that using the technology as an add on to control the interface is not effective as conversational operation and desire. While at most times the input will be dictation. Specific methods for all input classifications comes from the application at most times. For example, given that there are many programs and many types of information formats these applications accept, and of course the most important feature is the desired look, appeal and operations. Publishing.

LLVM Publishing

Within the publishing layer, many forms of performance gains are granted by configuration context and llvm. A desktop setting will reside within a permanent compile state after it has been accepted. The machine, having jit compiled the bc, along with providing sandboxing for security, will use the modularized component as a jmp instruction or call within its round robin discrete scheduler inlined. This provides a nice feature where seamless application is had. That is, setting the desktop without rebooting. The underlying linkage system, while at the forefront of embedded design, has to be performed on a computing device with a multi core modern bogo mips rather than an official power saving device such as a smaller cpu tablet. But a definitive decision will be had by applying the interface to an android device. This is wh one component of the user's system will be used for the compilation process and a private communication to the device will change its firmware if needed, or kernel boot, plugin, component, application or backup file. The update can be tailored for data operations at a block level rather than image by using gpg, auto correction, data hashing algorithm, and binary comparison technologies.

Publishing is where it is at, and font control is the finalization of such a desktop technology. Path rendering is computationally expensive while the duty to fill and stroke add more. The modification and traversal of path points, when mapped to pixel values or float values within a vector provide the best method of distributing edge effects and modulation. Hairy letters, and vines that intertwine within a compelling organized display of fine art, while computer generated, and controlled by the parameter attribute input, shall ring brightness in likeability. Compositing and layer system along with the standard Porter/Duff compositing operators supported in graphics hardware accelerated state.

Commonality of Splat Interfaces

Modern desktops have grown to support the commonality of interface usage. Lying down, propped up, from a bungee cord, on the cars, as sun glasses, on a wall, around the wrist, upon clothing and of course from the hip shot. The biggest problem is activating it when not necessary. The size of the screen, interface, and available application have to be under the control of the user with appropriate measurements from the environment.

Those games that run anywhere, and use the total capabilities of the hardware and input devices. The cd collection. I can squeeze much more from my android I know, that lonesome gpu and overused quad. The code a streamline concrete, branch, peep-holed, expressions reduced to native math operations and loop optimized to consider memory, native risc without the unnecessary bloat and much more considering llvm such as cpu cache operation which yields much greater performance. It's a binary mass of device specific integration generated by user production on the host system. Low, tight and fast. And, yes a device and control itself.

OTG is a technology that provides many types of usages within the user life. However the price often comes at the price of power. Ultimately, power charging from a centralized spot, while not the main functional purpose of the device, offers selective attributes many find compatible. The sport model also comes with a none data active charging port away from other centralized data ports.

USB D does appear to offer the necessary throughput for nvme portability. Those nice aluminum cases that toss around, Perhaps the ability to simply read a nvme descriptor partition block could show within eInk paper displays as a great labeling system. Allowing for newever embedded devices that work directly with plugged devices.

The system should be self maintaining offering the most personalization in application and desktop domain areas. System software is accounted for the standard developer mode, as it should

operate finely without a user focusing on the preknown device hardware setup. OTG is the known expandable part of the OS realm where it communicates to the bus.

Skia is a major rendering technology that is easy to use and has many features such as the ingredients of rendering animation, particles, 2d lines, filling, function based filling, gradient of several function types, antialiasing, vector path communication structures, binary implementations of vector base images suited for continuous animated rendering update via structures (skpicture). The system is well ordered and namespace, nouns, verbs adjective formalizing the api space are angular, very readable. But the documentation is medium level, these are not tutorials and there is a lot of system to consider. You will have to invest in books for this as anything else. The font rendering system is fabulous as it uses international layout systems by harfbuzz. Thus supporting ttf and legacy supplemental formats.

The tool system of wayland such as cairo, pango, (c and c++ versions) are not as fluid in their encompassing design. As a fish merely swimming in the sea of coded source volumes, all systems are privy to the eyes of after thought to summarize, organize, and structure the blocks such that usefulness, accountability in stability, and sustainability can be pursued. So I currently think that from my experience with all of the GUI technologies, that most of the libraries to date, and tui based ones, plus new designs of small screen usability not within my scope yet, a better model may be established that 1) first priority - accepts text input that is nlp corrective to gui form but through one call to buffer. buffer can be several formats, text, parsed markup, binary stream yet offers easy building and transfer of events for gui application to handle systems encompassing the realms of the browser. image format, dsp, midi musical rendering, raytracing, scene graphs, files and their formats such as epub, pdf, video, etc.

Providing these many applications within a separate architecture is what exists now. And this not only is bloat which is small enough that it was the yester progression to working speed, but the efficiency of switching between usage of the resources necessary to view or edit the data. By applying the plugin format and the needed design for memory communication using standardized c++ mechanisms that are shared access, memory usage will be applied at the right time, and of course at user command waiting much less. Each format has all types of software, list vector implementations, architectures for controlling the gui app menu, and its plugins, their own utility and cross platform support libraries and the bugs they fixed.

By applying the file system mechanisms, indexing system, and application server plugin with the virtualized data in compressed rolled form, perhaps an input mechanism within the gui uxdev api, controlling the separate systems will utilize cpu resources at necessary times much better I summarize logically. Ideally the same architecture library style accessible in uniformed form as a data layer only. Handle art files specifically offering the blend to gimp functionality, but plugged into the interface as per design. Audacity has major audio functionality that is encased within its binary body but also, tons of ui code. The ability to operate on all data from a modularized user interface context while being a headless program to provide flexible and extensive program tie together is the major direction of the application production engine. So the argument is headless design. The style proposed, and the necessity of including interface designs too specialized to break apart such as blender.

User Blender Clipart Adjusted Lighting

render farm produced three dimensional animated clipart with information panel composition / text - numerical graph visualization. Marketing materials for business store front.

Concepts arise in companion to modeling technology that change some graphical modeling usage scope into the user domain more. Imagine an artistic blender that supports building a scene within

the file browser to pose. Art blender ai posing via menu selectability may provide model specific parameters and applied as a time weighted and time stretch stack of verb adjective motions, movements. Providing them as a stack allows the variance and modulation necessary for adapted emotional movement. Audio, text to speech production of high quality. The models include numerical object identifications, much like GUID numerics that supply their persistence within an inventory of the world defined. Other scripts may interact with them in selected ways to provide scaled animation suited to the user's control. These scripts, generated dynamically using artificial intelligence, are python scripts that control the models, their meaning of symbolism upon interaction. Scripts can be applied using python scripts and skeletal IK connectes. Scripts that are produced independently as at times modeling and animation are focuses of different visualization perceptions and logic. Allowing for these creative persons to work more together on separate functional aspects allows their strengths to be utilized more.

Modernize Computing Hardware

The computing hardware device needs updating as well. To modernize the computer motherboard technology is of great benefit and in many ways a reason non standard embedded devices exist, alone. The Qualcomm snapdragon is a risc based SOC chip with video decoding capabilities and numerous cores. The production of RISC instructions is less complex as well the operating tempretation and power usage considered less than intel. It is LLVM compatible.

Soc video with open VG perhaps added in the future for desktop plane graphics. By supplying a opengl series with heat accountability for moderate gaming the product offering has shadowed the sony playstation market. Yet relying on networking for complete game play is unacceptable, thankfully nvme fulfills many purposes of designing such a device. In general, there are too many expandability options to discuss, but comparable gl with modern features is acceptable for visualization providing that more usage requirements does not gobble too much battery.

With touch and without touch desktops and expand to GB ram level of laptop level hardware, 32GB. 16GB as integrated manufacture of motherboard.

Pcie nvme only with nzip drive as discrete functionality. Performance on pre compressed data are the high vol items movies etc. One expansion slot for another drive like.

Video supported decompressor video audio.

Better cooling system for all components.

Newly designed bus removing legacy and modern portability. Power reporting system all components use. Pcie Nvme flash boot f2fs bios.

Signalinaling smart stream multi channel dma controller for linear TB addressing. Protected mode with memory subsystem integration with predominant devices that utilize it. Video, audio, market devices. Integrated wireless hide hub supporting multiple and many. 2.4ghz wireless mouse keyboard, mouse and audio unifying receiver onboard. Multiple connect bluetooth.

Five channel soundcard optional.

Networking.

The laptop only comes with a wireless keyboard and mouse. The bottom is its flat metal portable desk. Hinged multiple swinging. Can be removed with a bolted hinge lock. The thing can be placed within the bezel and mesh type inter compartment. Center, top or bottom. The screen joints and stand joints may pivot lock and tighten by finger. The telescope arm also uses the three locking positions. The stand locks as a screen protector during transport. Leaving hid components selectable, some stationary, some ultra portable. As well portable projector tech. Onboard someday via lazers short throw.

Keyboard first locks into stand and can be removed.

Stand optional for locking on to a telescopic memory arm to a table or other fixture.

By now, everyone has seen the awesome folding laptop computer that has an oled display. There are affordable HDMI monitors and also portable powered lcd monitors. The portable powered if used as an extension or clamping essentially doubles your screen size. An interesting addition would be to have an extension for the top that is half height. Essentially made for laptops with a 15.6 display, portability. The extension vertically from above would be very small, more portable, less energy. However, locking onto thin bezel screens is difficult. Ultimately a locking clamp that fits along the back of the older screen attached to the laptop. Essentially grooved into place.

More advanced is a sectional approach where the laptop screen could be extended. both vertical and horizontally with a gap in the middle where the older laptop screen slides into. A very weak joint, perhaps to be supported by the overhang by the slot nature. The system uses the HDMI and also reorganizes pixels to expand as much as seamlessly possible, given that the bezels will be present. This is the most useful in that the problem of miniaturized reading and font size is solved. The concept rearranges the space to use the original laptop as the primary but extends the usage to partnered window manager and software display technology. It is possible in this configuration to have laptops with a usable 19” screen with less power consumption and pixel requirements than a full size second screen. Two inches all the way around as portable as possible is a very organized solution that is economically competitive and very useful. People will use the technology for many years even though large screen laptops exist.

Another fit could be that the seam distance is decreased by allowing the bezel portion of the laptop screen to slide behind the new screen. The chance to integrate more seamlessly is there and perhaps an aspect of competition. LCD expansion sleeve that uses the HDMI port.

This is also an excellent place to put various speaker drivers - left and right on the overhang. There are basic ingredients which cost a base of money. A technology like this, really portable, smart space wize, and usable deserves the attention to another key component which is typically lost due to much miniaturization and low quality parts. Audio drivers, cabinet space, power. Low frequency , rectangular, elliptical woofer. Medium quality two way stereo. on the bezel with a low frequency brick for boom. Typically located away and powered by the crossover amp. The housing of the screen is now much thicker, and perhaps extends full length of the laptop vertically. The bezel mounted two ways are useful as a ten watt digital amplifier. The drivers are more distinguished than typical, in that they are capable of a decent frequency range. When the sub woofer is not charged or plugged in, the drivers try to hand the full range. At most times, loudness at the range of the screen is not as important than the detail and frequency range. As well personality trains suggest technological audio equipment and sound balancing is a related field. A dome tweeter may be too bright but offers a cost effective detail solution. Typically dome tweeters can handle many types of frequency cutoffs and often vary technically by the sound accuracy measurement to balance the midrange/midbass optimally. For example, a pair of Tang Band 1/2"

Silk Dome Tweeter and two FaitalPro Midrange, two balanced Peerless Passive Radiators. For the optional sub woofer block which as its own amplifier, 50w, power and battery, 60000, is very heavy but fun, The Tang Band W639SI will make a good sub woofer, 38Hz, with feel it sub bass. Theatre experience. The cabinet can be a radiator one, but a live cabinet would work the best as a port design to lesson the needed power some perhaps. The important aspect is the volume of air matches the acoustical width and the tube's length equates to the dampening harmonic frequency of the measured setup. That is the presence of the cabinet is kept from vibrating by the port filtering the vibration. There are different aspects to the enclosure design that an audio engineer would provide much more technical information and perhaps much cheaper drivers and power requirements. However the main problem has to be solved as to portable mid grade sound with sub and LCD expansion. It is loved by the masses I am sure. Sell it with game controllers and some games.

Conclusion

In summary, the progression of the desktop has shown to shine with very little glitter. People where happen with jagged lines, and flicker because it worked. Hopefully, this will be the same with the stuff that does not strobe on the screen and the text is readable. The new features will be the accepted normal, such as tiger letters with fur reacting to the pet of the mouse cursor. Some forms of lighting can be achieved but not full 2d vector text screens. The accoutrement of polygon reduction within the 3d realm should be incorporated within the rendering of objects. How many triangles does it take to make a button, gui elements with a 3d scene graph element can provide the real time lighting. The most effective for these types of screen elements will be the bump mapping to provide texture relief.

See Also:

- gentoo
- clearlinux
- wayland
- NZip Technology
- Qualcomm Snapdragon microprocessor
- Snapdragon Technical Presentation