

Ideal Properties of a 3i Candidate

From manabars.com, 2006. Information taken from many public sources.

Categories

Security

Security is a very broad description for protecting user content, privacy, money and identity. Security should be addressed throughout the execution stack, starting from hardware for both authorisation and computation through to the delivery of content to protect user information.

Convenience

Historically computers have been difficult to deal with. Many features of content radically simplify how users interact with information to save them time and increase their productivity.

Privacy

Information that is owned by users is considered private and cannot be accessed by others. There are many technical aspects to

achieving this vision such as preventing content from disseminating the information that it has to work with.

Reliability

Traditionally, computers have been notorious for breakdowns. Content on the Network delivers a new level of reliability for mission critical software.

Economic

The entire Network and all the content it holds should be founded on an economic system where all resources must be paid for. Content should guarantee royalties, but should be forced to pay for other resources that it uses.

Advanced

The advanced features should bring new innovative concepts that raise the bar on what information technology should deliver.

Security

Zero attack surface area

The MIC hardware utilises Set Theory to achieve computational completeness at a level of scope. This prevents software from attacking new software. There are also mechanisms to prevent software from taking control of the Operating System known as the Governance Layer. Set theory prevents the Malicious Client Problem, and the MIC hardware prevents the Malicious Host Problem by eliminating the host altogether by running as hardware directly.

Introspective scope of execution

The foundation unit of information utilises a principal known as Perfect Encapsulation to allow a set to access only its own references. There is only a single machine instruction to gain access to a new reference and is called GetChild and this is an inward looking function. There is no instruction (at a hardware level) to allow external scope that may be used to locate any parent set.

Protection against advances in factorisation algorithms

A private network that is secure should prevent the need to utilise mathematical mechanisms such as public/private encryption that is vulnerable to new discoveries in mathematics

Protection of encryption keys

Since the grid network is always on, systems that utilise old passwords can be automatically updated with stronger passwords over time as computers become more powerful and put weak passwords at risk.

Alternatively, archived encrypted data images on static media pose a risk of possession or control if the decryption key is compromised. On the grid, old images should be automatically updated.

Prevents Distributed Denial of Service (DDOS)

DDOS attacks are a result of the poor architecture of the TCP/IP protocol. DDOS attacks should not be possible on the Network because all software should operate within an accountable

framework. Software should pay market prices for all resources consumed and be unable to affect resource contracts that have already been purchased by other software.

Parasite Free

By forcing all information into an economic framework, the Network should prevent parasitic problems such as Spam or Splog.

No ID Theft

Portals should uniquely authenticate the content that it uses so that no other Portal can access content that is private to a Portal. Only one person should access each Portal. Authorisation to the Portal should vary from pin codes to biometric. When Portals are accessed from an insecure location the user should be locked into a sandbox and sensitive information should be contained in digital safes.

Brute Force Attack Free

All authorisation systems should require at least a 128 bit key and a financial deposit for each key attempt. Regardless of how powerful

the computer, it should be impossible to brute force attack the system because the amount of money required to attempt even one billionth of the possibilities would cost trillions of dollars.

Backdoor free

Developers of content should not be able to access the Network or the broader internet. Instead, they should be forced to communicate with predetermined services operating from within sandboxes. As a result, they should not be able to distribute any information that the user may have generated with the content. Nor should they access any other content already in the Portal, so it should be impossible for software to subvert a user's computer against their knowledge.

No monitoring by host

Although software executes on an open grid network, designers should be assured that the software will not be monitored by the 'malicious host problem'. Administrators should specify the exact cluster of computers that the applications may execute on.

Provided that these computers have been secured, then the applications should not be able to be monitored.

Convenience

No backups required

All information should take the form of a Digital Business (Mobile Agent), which should manage its location and persistence on the grid network.

No software installations

All Digital Businesses (Mobile Agents) should be already configured in an executable form and should not need to undergo an installation when moving between Portals or Nodes.

Binding of application and data

Digital Businesses should be always bound to the data that they represent. Since information is always available on the network

there should be no need to separate these two components. Both application and data accessibility should be guaranteed.

No hardware required

Users of information should utilise any device to access their information and should not need to own any hardware.

Forward and backwardly compatible

The Emulator (Virtual Machine) should be a complete and unchanging specification that is capable of describing all information and therefore should ensure total compatibility for all software.

Device independent

The Emulator should run on any hardware for which it has been ported. Since the Emulator should be fairly small and utilise limited libraries, porting should be a fairly easy task.

Data accessibility guaranteed

The Digital Business that is used to render and manage data should be bootable at any point from a fixed image. This means that if the rendered image becomes corrupt then it should be rebooted. Rendered Digital Businesses should also be automatically shutdown and sent into hibernation by a user to save resource. The image should then be used to resurrect the hibernating Digital Business. The boot image should also be used when transferring an application to another user (see Content is free to distribute).

Stable operation

Objects or content pages that link to other content should not face the risk that subcomponents on which they rely may change or suddenly disappear. Once content has been created as a Digital Business it should not be changed - instead, new content should be released, however, the source code or documents used to generate the original content should easily be used to regenerate a new Digital Business. Since security patches are not required (see Zero

attack surface area), information should continue to provide stable operation for any process that utilises it.

No Logical Limit to Portal size

Portals are virtual containers that hold a user's information. The Portal should be distributed and exists on a multitude of computers. The Content that it contains should be itself distributed, so there should be no physical size limit for software - outside the physical size of the entire network itself. This should be useful for photographs, music and movies or doing system backups.

Native Collaboration

All Content should be shared by any number of users. When users make changes to the Content, those changes should be reflected in every copy on the network. Developers should not need to be concerned about coding this functionality into their software as it should be inherently part of the infrastructure.

Zero Download Time

Applications should be accessed instantly, regardless of their underlying size, as they should be executed directly out of the grid. A Content item should also be copied instantly no matter how large it is. Unlike email, messages should be received in the Portal immediately.

The only time delay should be the transmission of the image data to the device presenting the Portal

No information loss

Sometimes when software has been programmed badly, bugs can cause unintentional information loss. Content in a Portal should record every change that it has undergone in a similar way to “undo”. At any point when information loss is discovered, content should be able to be rolled back or pieces copied from a rolled back version.

Privacy

Spam free

References should be used to send and receive messages.

References should operate in a private address space should not suffer from leaked fixed addresses used by the public. If a private reference is used to transmit Spam, then just that reference should be deleted while all other relationships should remain unaffected.

This should allow a user to remove all public points of entry for information.

Impedance of advertising

Information or content should automatically charge each time it is viewed, so there should be little incentive to include advertising banners with content pages or objects. Revenue should be gained directly without having to introduce a parallel advertising mechanism that is likely to introduce unrelated content and reduce the perceived quality of the information.

Unauthorised data dissemination impossible

A Digital Business that renders content for a user should be unable to transmit the database or store of data that is intrinsically bound to the application to any other object on the grid network. This should provide users with the security that information is safe from Trojan like applications. This is different from Introspective Scope of execution that already prevents a Trojan from affecting other data on a system. This mechanism should specifically prevent a Trojan from disseminating information that it is required to have access to for normal operation.

Absolute Delete

Users of content should be assured that information is completely destroyed. This function should be supported by special hardware instructions that should literally rip references from every parent set that contains them. This is different to garbage collection that only frees up memory once it is no longer referenced.

Mission critical applications for the military or government can be assured that the information is destroyed digitally.

Reliability

Grid network

The Network should be an open and unbounded grid with no grid administrator. Anyone should be able to contribute to the mass of Nodes that are protected from the mobile agents known as Digital Businesses. Digital Businesses should be able to choose which host they wish to reside on.

Realtime platform

All resources should be guaranteed and should be made available on demand. Resources should include memory, hard disk, bandwidth, electrical power, computational process plus specialised resources that relate to the architecture only - such as capacity for garbage collection.

Guaranteed supply of resource

Resources should have guaranteed availability, whether physical or as Content. Physical resource should be purchased in advance to guarantee supply or contracts purchased that should provide a level of cover for periods of unknown demand.

Not Encumbered

The Manabars III Model Candidate (MIC) is a patent pending hardware architecture meaning that implementations are less likely to infringe on other patents.

Economic

Generates revenue from hardware

Owners of hardware should generate computational credits that should later be sold on the open grid market. To do this they should allow Digital Businesses to utilise excess processing power on their machines.

Monetary API

Digital Businesses should utilise software within the Network that should allow them to transfer computational credits and ultimately money.

Prevents reverse engineering

Digital Businesses should be configured to be inaccessible directly by their users and should only release the outputs to a protected and hidden internal process.

Royalties guaranteed

Digital Businesses should automatically charge every time they are used. Software or users of a Digital Business should not be able to subvert the payment mechanism. Digital Businesses should not be able to change its pricing once it has been released.

Content free to distribute

Globalisation is characterised by organisations focusing energy internally and competing externally with products. The patent system amplifies this dynamic, which creates barriers to entry to markets and inefficiencies.

Embedded Charging should be the opposite. Users of Digital Businesses should be able to use any content they encounter without limitation because all content should be free to distribute. This should radically increase the uptake of useful software and content. Users of the software should still be charged during its use, but if they own hardware they should be able to offset this cost with money earned when renting out idle clock cycles.

Economic Foundation

The Network should preserve the economic principles of asset ownership, asset protection, the regulation of Content, the taxation of transactions, the capacity to set a price for an asset and the capacity to transfer asset utility. These principles are derived from

the old economy, but should be supercharged in the Network to increase efficiency and productivity. For example tax evasion should be impossible, royalties should be guaranteed and reverse engineering should be impossible.

Advanced

Avoids the von Neumann Bottleneck

The Emulator (VM) architecture should be optimised for a silicon implementation. Any number of simplified von Neumann architectures should be installable on the chipset. Further to this, each Emulator should operate independently across the grid network.

Globally compliant

The Governance Layer (Fielden) should allow the system to integrate seamlessly with a single interoperable solution such as the 3i ([Interoperable Information Infrastructure](#)) as proposed by Dr. Kay Fielden. This universal Operating System should tax earnings from content, ban illegal content and trace money transfers.

Each Node in the Network should be installed with an identical Operating System that can be upgraded so as to comply with various regulations.

Sensitive transactions such as money should be monitored or intercepted across multiple Nodes. The universal OS should ban information that is illegal such as child pornography or illicit drug shops.

Extensible hardware architecture

The abstract architecture should provide an opportunity to develop hardware improvements without having to "climb up the stack" such as is occurring in the aging graphics pipeline or with multiple cores in a CPU.

Reconfigurable computing architecture

The hardware architecture should be fluid and should be extended dynamically at runtime when implemented with Magnetic Logic or Field Programmable Gate Arrays without shutting the Node down.

Zero dependencies

Compiled software for a Digital Business should require zero (library) dependencies for normal operation in which information should take any form and should be future proof. However, in the event that a hardware plugin is added and a library used to connect to this hardware, that particular Digital Business should only operate correctly on a Node with the correct plugin support. The library that supports the plugin should be attached to the Digital Business (and hence the grid network) - simplifying integration. This is equivalent to new hardware automatically locating its supporting software across the grid network.

Encourages Renewable Energy Technologies

The Network should transmit power between Nodes as well as bandwidth. As a resource, power should be governed by the same resource model as bandwidth that allows software to purchase resource contracts between each Node and enable it to guarantee a fixed bandwidth link between its endpoints.

Applications on the Network should purchase power on behalf of households, factories or even appliances. Such a power system should not be able to be hacked by terrorists, prevent cascading power failures and should be fully distributed -- allowing power to be injected at any point, provided that the phase and voltage are implemented correctly.

The Network should be renewable energy friendly and should handle a large ratio of renewables to thermal generation. Normally, thermal generation must account for the vast majority of the total generation capacity. If the sun does not shine and the wind does not blow, thermal generation can always meet the base load.

In a renewable energy economy, there is normally insufficient base load for cloudy and calm days. In the Network, applications should trade power contracts to reduce the base load when necessary. Furthermore, applications should purchase power directly from the source from where it was generated. Compare this to a coal power station that pays a carbon tax but is still able to generate a profit and operate. On the Network, the power users should simply not use power from dirty sources if they choose to do so.

Network Intelligence

In the same way that brains use fear and pleasure to govern their structure and behaviour, content should financial cues to modify the application behaviour.

Content should embed other content so designers are encouraged to chase the configuration that invokes the most usage from the user. Collectively, as thousands of Embedded Charging content items search for maximum profit, they should naturally reflect an intelligence that continues to sharpen over time as more content is generated to take advantage of human behaviour.

This is similar to the way the old economy is intelligent when it already has manufactured a selection of foods, exactly when you need it and near to where you live. By rewarding companies financially, people can purchase the items that they want. In the same way, content should become intelligent by rewarding the designers when they produce behaviour that is useful to the user.

Fixed Transmission Latency

The Network should operate as a transport layer with fixed latency between source and destination endpoints in the network. Unlike TCP/IP where packets are shunted chaotically around the network, the Network should guarantee fixed bandwidth for each leg in its journey. As a result, other processes on the network should not be able to interfere with network transmission.

Transmission Accountability

Software should purchase all resource that it uses on the Network. This innate accountability should guarantee the fair payment of resource. For example, since TCP/IP is not economically orientated, Nodes move traffic through their routers for free. As a

result, Nodes that are strategically located on the internet backbone (eg CDNs) pay less for traffic, where users on the fringes pay more.

Accountability should be important for the stability of the entire network. Should a computer on the internet backbone transmit vast amounts of traffic destined for the fringe of the network, it will cost the fringe users dearly to receive it. In an attack form, packets with a bogus return address leaving the internet core and headed for the fringe could easily clog these outlying networks.

The Network should force equitable payment at each Node – preventing any form of economic attack.

Interoperation with the patent system

Embedded Charging should not only solve most of the issues encountered with the patent system, but provides a smooth migration path for this mass of legislation. In cases where the content has been shown to infringe, the revenue should be redirected to the legitimate owner or in severe cases the content should be banned.

Hot Linking

Hot Linking should allow class implementations to be modified without having to shutdown an application.

Supports Network Coding

Network Coding allows more information to be transmitted across bottlenecks - provided that data is specially encoded and alternative routes are used. On the Network, Network Coding can be managed by applications directly because applications should be able to purchase fixed bandwidth contracts between Nodes and should manage the raw data feed between each Node on the network directly. For example, Network Coding should be managed by Routing Agent applications that already use the raw feeds to switch data.

The economic foundation of the Network should provide easy billing options for the shared encoding technique used by Network Coding. The abstract nature of the Network with homogenous Nodes should simplify implementation across wireless and wired networks. Furthermore, the problem of determining which communication

channels must be mixed should be simplified because the entire flow of communication that defines the exact route for every stream can be known by the application in advance, prior to message transmission.

Finally, the Network should provide a solution to catalysing the network which would otherwise prove daunting in attaining a global deployment in today's TCP/IP dominated network architecture.

Unlocks Network Monopoly

Historically, network orientated businesses such as electricity transmission and telecommunications have suffered from a monopoly imposed by the owner of the network.

The familiar dynamic is that new connections to the network are controlled by the network owner. A lack of clarity on billing and how the owner of the network is to recoup its costs fairly has often led to a monopolistic approach. A monopoly can easily lead to pricing abuse, often ending in government intervention.

The Network should break the monopoly by empowering each Node owner. Anyone should be able to join provided that they have the permission of the owner of the Node that they are connecting to. They should be able to charge anything for their own resources and compete in an open marketplace. Charging too much to transmit bandwidth may cause traffic to be routed along alternative paths.

The Network should have no grid administrator and no owner that can abuse the network.s