~~Electronic Cash~~ Bitcoin without a payment intermediary. An essentially Peer 2 Peer electronic cash system could allow online payments to be exchanged directly from one party to another without the hassle of exchange using a financial intermediary. Digital Signatures offer a part of the solution, but the primary benefits are lost if an intermediary base on trust is required to ensure that a transaction cannot be double spent.

We propose a solution to the double spending attack vector based on the universal broadcast of transactions over a tightly(?) connected P2P network. The network of commercial nodes competes to from a timestamped record of transactions that are ordered cronologically.

Competitive systems act as nodes that record transactions in a sequence or chain of hash based effort proving the economic cost spent in creating the WORM form (farm?) set of records in a manner that precludes altering one value without returning to calculate all proof values in the sequences. The longest chain of hashes serves as both a cronological sequence of ordered events and a public record of all events. It acts as a witness to the recieved of order of events and is a set of proof and acts as an evidence trail showing that the majority of the systems acting to validate transactions hold the longest economic investment and expendature via the most "wasted" CPU power.

Whilst honest nodes are operated and controlled by the largest amount of CPU burning over the connected network of validator companies, these competing systems generate the longest chain and outpace any attacker.

A highly connected small world farm network that broadcasts to all connected nodes (see notes on NIPPA @ ASX) only need a basic structure and self farms and is self healing and resiliant to attack and/or outage.

Transactions are sent to all nodes using a broadcasted basis by best effort. Groups and systems competing to act as a validator and to earn revinue are not required to be a part of the ongoing system and can leave and/or rejoin the system at will picking up where the last value was validated. The nodes use the longest valid chain of work evidenced hash expendature in order to validate ~~on~~ a commonly accepted ledger of occurences that can be validated even though some nodes come and go.

<u>Intro.</u>

Commerce on the Internet has ~~come~~ become almost totally reliant on the use of risky, costly, financial payment intermediaries.

Micropayments and small online token payments have been promised for decades (Tim May → Black net → Spyder). The trust needed is too high + transaction costs are too great. Disputes cannot be arbitrated without costs that preclude small casual payments + any web pay by page system.

Fraud cannot be avoided (see Granger re Fraud Triangle and role(?) of triple entry ledger)

Merchants hold PII – adds to cost + risk.

Non-reversible – small games – WOW tokens.

Web has no physical currency token like a coin as noting can be done for under 10¢.

An ectronic payment system that is based on costly to cheat audience is required – (ecash failure as Double Spends and IAA are too costly but anonymous. = trolls and cowards.

This shield of cowards always leads to the worst factor in society collapsing the system (Plato – Ring of Gyges).

Q → why do ppl think privacy and anonymity are even close... Anonymity is the tool of criminals and the dark state actor.
Privacy is honest.

If a system is anonymous it cannot be private – Ever!

Privacy is to Anon
as Good is to utter Evil

Always Doubt spends – how to have a system with no intermediary.

Answer – the ASX (NIPPA) was a central hub spoke.

[hub & spoke diagram: ASX, P2P eCash]

If we take the centre and have this removed as a complete broadcast like many P2P ~~piracl~~ pirates are using to avoid honest payment, we can use the tools the criminals advanced against them.

P2P File theft networks do not care about double-spends – the information in "free" or has no value (socialist). We just need to have ~~an~~ a time ordered economically enforced ledger of all transaction messages that all parties can validate.

The ledger + economics solves the (drive?) of P2P (criminal) sharing networks and defeats these parasites. We create a distributed commercial P2P timestamping service where parties compete to validate a separate IP 2 IP peer exchange network exchange ordering these in a FIFO structure.

If we take the use of code as a predicate we can build a TTM or as Sipser called it, a decider.

T is valid = True

is False or not Valid = Reject.

The system is economically secure. The chain of Tx's mean that a Red queen game eventuates where an attacker is aways behind the honest group.

→ unlikely that a criminal majority could eventuate.

For the criminal groups to gain a majority they will need to run more systems → sustainably than the honest ones.

→We would expect the creation of Data Centres + large server farms. If the timechain concept works. A data centre is easy to find and as it takes a capital investment over time is difficult to Sybil.

This – we can expect that criminals and other dishonest groups who have resulted in the collapse of all other digital cash systems, will not be attracted to my system.

(Lynn's law tutor may be of use here and has a contact to the KPMG Legal guy in Melb who was on ecash tracing paper).

Transactional Message

A digital coin is formed using a chain of digital signatures.

The User transfers the Coin to the user they want to exchange with (see Northumbria notes).

– If User to User, we can say this is a discrete exchange.

– But → as a backup we send to the network + they hold a transaction for a defined time (*?) to be sent to the recipient when they check the network.

(Ozemail – UUCP analogy)

(Northumbria – if this is not User 2 User, it would be likely a Postal Rule exchange in UK law)

In the past systems, users cannot directly verify that a transaction has not been double spent already. This is why an intermediary exists
– to report + stop double spends.

What if:

It is a self-help network. That is all users can see the chain of validor messages and can check if the coin they are offered has been presented previously themselves?!

The ecash (90's) process needs a mint + acts as a depository bank but this is not efficient.

→ again cannot do small payments + kills web commerce. If we have an overlay structure where all transactions are public, this means that any user can instantly validate the state of the transaction that they are to recieve.

An intermediary has always been necessary,

Annocing to all parties means all can check – but not all need hold all.

The time chain is crucial as all parties must agree on the order of a transaction and a single history results. (No more Enron + many books – Wattyl)

the User recieving a payment thus needs evidence detailing the true time of every transactions as the majority of validating companies have agreed to state (in open competition) is the true history of what was 1st received – FIFO model.

(Tracing law seems to lie heavily on FIFO. – see case)

Also Modex + Smart card tracing orders.

Timestamp and ordering Server.

We start with a distributed time and validator (or order reporting) server.

The server is a distributed group of peer systems – commercial entities act to compete.

Each seeks payment by transaction (as earnt in validating a batched set of transactions.

(see Brooks as well)

We ~~b~~ batch a set of messages into a single block and then hash this (order now matters)

The hash of this block of batched transactions and/or messages – This is similar to what S Haber/WS Stornetta (1991)
(1997)
proposed using newspapers and UseNet. We use this public source of hash (one way) information in order to timestamp a batch of digital messages.

If we link the hashes (see the Lampert (?) chain) so that a hash also has the hash in the batch from the last Batch – we end with a chain.

like Lampert (?) (Lesley) and his signature scheme, the chain is now more secure as it extends.

Network

The network consists of a P2P user layer on top of a P2P (commercial) validation layer. This will evolve if the system works.

1) All new messages are recieved by nodes on the network and are then broadcast to all other active nodes.

2) Each node is the commercial validator network orders the messages into a batch.

3) Each node seeks to solve a difficult puzzle.

Aura et al have an interesting idea on a hash based puzzle where you seek to find a partial hash collision.

See more Aura. Seek code examples to use.

This puzzle can be matched against the hash of the block formed of the last value hash and the batch of time (strictly) ordered (chronologically) transaction messages~~)~~.

4) When a validator in the Peer group finds a solution to the hash puzzle (see also mod square options). It seeks to broadcast this to all other validators as fast as it can

→ Note these commercial entities compete for the batch fees (the guys QSCU use + D have can act as a model → even if single entity + Bank)

5) The nodes validate the batched Block, its contents, and the solution to the puzzle seeking (is zero sum game → set reward per period) to invalidate the competing nodes in any error (as this increases the overall revenue of each node with a good + honest approach).

– batches must only have valid ϵ transactions that are not double spent.

6) After validating the batch – the reward is paid to the winning node and the nodes in the group accept this – demonstrate acceptance – in then using the discovered puzzle value to move onto building the next batch.

The hash of the accepted block is included in the next batch + block.

– Note – Nodes will work on the longest chain (should keep other – in case) Seek to extend the longest chain. If two (or more) fork exist, then as the world is large and this is distributed, we end with a race and fork at the chain – this ends through Competition.

Users – Do not care Users have SPV

This is a validator issue. (need to have validator rewards unspendable for a significant time to make certain that forks do not lead to invalid spends – the only issue – maybe have validators need a day's grace before being able to spend revenue?

See Small World – we would expect the validators to be densley (?) connected.

This means – we can expect any transaction volume → Terrabytes at a time to be sent → commercial entities seek profit and the network will scale as these companies invest to capture profit.

Faster sending of the batch + the solution = more revenue.

The system goes to an ~~equip~~ equilibrium where the revenue and cost balances ar(?), where the cost of scaling leads to a system that costs more than the fees to maintain.

– DDOS + Fault tolerant.
– Dropped Transactions etc are OK.

– Inefficient nodes (companies) are going to earn less and be dropped.

Incentives – This is an economic model + companies compete. We will jump start with a subsidy → see Singapore/Korea.

→ reward miners + validators.

→ like seeking Gold.

→ As more – value per subsidy decreases to wean the validator off of the uneconomic payment but to allow them to see that they need to compete and hence professionalise and this means to grow the transaction volume AS LARGE AS Can be by any Legal method.

– IOT

– EDI(?)

Any message + timestamp will add fees and make the cost of fees to the micro payment use less
→ More transactions
→ larger batches
= Stronger more professional network

Companies Build + Create a competitive system.

Explain + detail Fees.
TBD.

Note it is the incentives that allow the system to professionalise as the message value grows.

→ Moor'se Law

There is no limit to scaling the P2P digital cash system once it has been bootstrapped.
(even pay for Page – no more Ad's)

Dishonest validators end
→ In Law!
→ In being rejected by honest nodes seeking to grow the network,

Size + Space

Storage Doubles faster than Moore's Law. But We can prune old values when the chain is deep enough.
→ likely way back chain as an archive.

Memory vs Storage.

User Payment Network

The user acts on a simplified network – Only need batch + manage headers.
Alice (A) wants to send to Bob (B)
A → B.

A sends the path + hashes. (ie, A proves her message + the fact the coin she has recieved has not been spent.

Merkle vs other trees.

→ to check more.

Attack only ends as a double spend and this is like cheque fraud with signed evidence (law).

Perceptron

Fan in/Fan out.
need to cover the Minskey stuff.

One/Many Inputs
↓
One/Many Outputs.

(change as a concept).

lots of coins
– like minting + re minting

but all users do this themself

Simple Payment is for Users

Users Do not need to check the entire "Time chain".

→ Validators are paid to order and check.

Calculations Sections

→ Weller/Stats.
→ Poison/Reliability Model.

TBD later.

Need to model propagation

– SEIR

– see virus + epidemics.

Privacy

Not and will not be anon

Those criminal cowards are the bane of most systems.

–Must kill off crime.

Privacy is not Anon.

We do need to isolate PII.

Not like Visa

HIPAA
etc.

look at Firewalling the system.

Users are able to exchange

identity But like the ASX (NYSE etc.) ticker – The public validates anything but without recourse to seeing the ID of the parties.

[diagram]

Alice	||	Chain
↕	||	Batched Messages
Bob	||	Transactions History.

Firewall

Known=Shared	Private + not available

[end diagram]

→ Not like PGP.

Keys are Not reused.

$P(A)_0$ = not shared (ID on PUT-CA)

$P(A)_1 = P(A)_0 + P_{AB}$.

P_{AB} known by A + B?

$P_{(A)1} \leftrightarrow P_{B(1)}$ exchange

A+B can prove the other

But A+ B not linked publicly To $P_{A(0)}$ or $P_{(B)0}$.

→ so ID is private

→ Need to test(?)/chart(?) the finances of this system.

How to Boot strap it?

When large – Easy

Small → Urgh – Vulnerable.

So much to do.

Alan → will BDO find systems?

Lasseters → Would be a means to stop what occurred.

Would Centre bet + Sporting Bet use a token?

Need to see GGS + all other games co's also – WOW.

also seek ideas in SCADA

– IOT – power etc.

LLM topic – Internet Intermediaries.

We need to write how the Intermeds are – covered under law. How they can be constrained in have a legal framework for crimes online as money needs controls.

Need Protocol Versioning such that we can add layers.
– VER 0 (TCP)
– VER 1 (UDP)
– VER 2 (ICMP)
– VER 01 (HTTP)
– like IP → layers in message.

[diagram]

Freezing + Recovery is Far too big an area for my LLM. PhD later?

The system needs a way to add Rules later – This needs to be distributed – Money as a base – allow a token for a separate country.
-UK, China, USA, Japan +
→ This will be encapsulated and able to be exchanged in Bitcoin – See P Wrayner (?) – SWAPS – The atomic swap ideas in his game book / likely need OT – Oblivious transfer would allow private transfer of assets on the system – Coin (A) Coin B swap.

The Rules will need to be International at the base layer – See internet intermediaries + liability.
-Common Carrier
vs
-Action.

The validators will need to act to freeze (WFO's) assets – easy as FIFO with ledger.

→ Alert the system.

We need an Alert Broadcast to stop the system moving a coin if it is validly frozen – Fast – maybe a separate alert node system. A Court can sign a key – if valid court order – the coin can be tainted as invalid in jurisdiction X

Rules would cover restitution.

As the Validators can add code – code can be altered if the majority of validators agree – the commercial nodes are bound under law – with an audit record.

Tracing is simplified (Not like Blinded systems).

Validators are a lot like ISPs/PSPs – Payment validation as a distributed clearinghouse – Valid and honest as all these guys are Public.

Public vs Private – child Porn/Terrorism Funding issues.

LLM Disertation

Must cover IP + Internet law. The proposal is ok. The structure is ok – Need to get a {draft
{ outline
by EOM Oct.

———

Alert is simple – send a message to all nodes – if a node or a user sees the alert – hold and Freeze – who wants a tainted money!

We can use the same broadcast to send such messages but later can expand this and add other non consensus rules.

The rules are important.

What is needed is a means to act on a WORM.

Oracle – SANS courses.

WORM. Talk to Alan re WRite once read many.

US clients are doing accounts not (Enron) on Oracle with the flags set to stop detecting(?) records – they could use a WANG B tape.

– Once in
Always in

Set change → a new record adds to the tape – the chain is altered as a valid reward sets why a change is made.

So – $A_{(\phi)} \rightarrow C_{(1)} \rightarrow A_{(1)}$
$\rightarrow B_{(1)}$

Here A has asset (ϕ)

C issues an order and this is written to the chain.

The order allocates $A_{(0)}$ to $A_{(1)}$
$+ B_{(1)}$

So – the coin is now split by the terms of an order.

Too much for now – Not POC. But must get a means to allow the validators to follow court rules or a court will simply rule the validator illegal.

The timechain is a law system.

The Rules are enforced by the validators – who act ~~are~~ as responsible Corporate citizens as they are Public

Important users are Private
Validators are Not

Validators are utterly Not private as this scales
– all they do
– all they earn
– where they are.

All is seen – nothing a validator can do can be hidden – they have no clothes.

So – Validators will follow rules

OR they will be FUCKED!

This Solves the issue of freezing a restitution

Validators are under the Thumb.

But Global law –
→ few crimes are global in nature.

→ International enforcement.

The rules allow restitution (where economically justified).

The validators just alter the code.

→ Version 2.

Need Lynn's Paper + work.

"Corporate Social Responsibility"

The validators have no PII as the reciever generates a new key pair and provides the public key to the sender just prior to signing.

Soon PII + Privacy are no issue

[A + B] Identities
↓

Template address.

Old Fashion law enforcement + tracing still will work and we save all the costs of managing private data for small transactions, the cost of an attack is low but the cost to the validator are high.

The only double-spend is where the validator acts to add the double spend – either they are a party or they are complicant. Instant with small risk.

– but a signed transactions

– this means the reciever has an action and if not small – law

Attacker is cheque fraud – but with a signed Public audit + evidence trail.

→ A can be dobbed in by B.

Law can then freeze assets.

A will have signed transaction to two parties – this would be saved as an alternate chain history.

→ not forever – but long enough. As the keys are not reused as attacker cannot plan the attack in advance.

More if B sends the message. Then B wins as the probability of A's message getting to C and outpacing B's is based on an exponentially decreasing race.

More B could see A's message to C, know A is a cheat and stop the payment – keep the goods but maybe get paid – Crime does not pay with a time chain of blocks of batched transaction messages.

The cheaper the Batches – the more secure. The reciever is alerted to the senders cheating in mere seconds – this could be better as the validators improve/professionalise. The only real attack – An attacker could try and change one of his own transactions to do cheque fraud and take back a payment of money he has just sent.

No Fed/Central Bank infinite printing either as this is

1) ~~Globla~~ Global.
2) Publicly Auditable.

This idea(?) could allow DAATS
CAATS

To be AI + automated – Audit will be about advice + classification again not the crap we are Doing

Need to discuss with Nevil + Alan. If BDO starts to act as a Validator this would get around early issues as an "issuer". They are tokens – commodity but with exchange value – not legal tender – But could open up a Path to have banks use the chained block system in order to have perfectly honest auditable Fiat money.

– Batched Coin

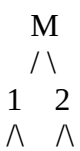
↓

Fiat negotiable instrument.

→ tokenised USD – by Fed.

but – Central Banks are now publicly auditable – all the people see the games(?) – no more secrets.

Extended – a log could be in messages.



each log in a merkle tree to collapse – but eg root logs are not collapsed – By value.

IF this works

I will get Both.

{ Partnership @ BDO
{ Professorship.

– Set!

Aura, et al 2000

DOS Resistant Auth. with Client Puzzle. T. Aura and others have a good idea of a client puzzle. They do not understand economics. They have all systems grinding which skews this to botnets + makes criminals have an advantage.

The idea is sound – the puzzle – really good – SHA256 version (replace SHA1) and this is the evidence to show work in our chain.

The resources are best in a market, a user can best show by buying a proof of work done by another party who sells to a (global) market so the price is reflected as the use – not system by system – In having a fixed token supply – the value of the token will be traded and a value will reflex market demand.

[diagram] logistic price.

The client commits resources in the form of money – not just as an effort at time. A user will be expected to have One PC. a botnet can be millions – so the hypoteses of Aura + even Dwork + Naor(?) (98) is flawed.

But a user can purchase a token at a ~~set~~ exchange valued/market rate and now – any dishonest/criminal host is incentivised to sell tokens to honest users. This kills off the negative externalities of the free Internet model.

→ Need to get contact for Aura – No reply to email.

The time ordered solution to tokens will allow these to be sold if it is able to be exchanged by supply is limited – Cap

→ Gold is a Good Model.

In a manner – Validators are analogous to Gold Miners.

→ The subsidy could degrade in the same way.

Set in blocks – but a planned removal(?) over time. like Gold, lots – easy at the start, laying about on the ground, then over time, it becomes more and more difficult.

Aura's Auth Protocol. (S4? 54?)

Better if we have my idea as a market token – also – a ledger is a log – so attackers can not delete logs.

Difficulty Adjustment.

→ Set as Stag Hunt

~~Ned~~ Need more on

– Stackleberg Games.
– Red Queen.
– Oligarchy.

If share based – leads to Rent seeking. Arestocratic – what work was done Not what is being done.

Share based system results in lower work + security. Rent seeking → crime.

Botnets → mining to get future share. Must have miners – aka validators. earn each batch of transaction messages.

Information IS Value.

→ Token could link to selling Access (see extended Aura ideas).

This use of a partial match is simple to implement. Aura's idea of the initial zero's in a hash puzzle Being matched is also simple to adjust as the difficulty is modified. So, far better as a work measure. We can use the market to adjust the evidence of provided work. The use of a subsidy does help (I hope) bring parties into the game. The validators act like gold miners getting Gold off the ground at first then, as this is all found, they need a harder + harder effort to stay in the game.

This should force the low preformers out. As the system grows – small players with low powered machines and slow networks will be made redundant (though they are still users).

Only high end corporate systems, data centres and companies who are willing to invest in this Red Queen Game will remain. The end – A few very large players – (Pareto) and some small ones that discover a neiche means to mine profitably.

→ The system can also support specialists

A puzzle of a puzzle may be used.

This is achied with a hash of A hash (solves some other security issues as an asside).

Puzzle = H_1 (H_2 (x)).

Now – P_1 = H_1 (H_2 (x)) needs to be solved in full.

But Party A can order transactions when the system is large.

The batched messages are hashed and handed to B.

B solves P_2 = H_2 (x).

→ As we have used a tree hash A can have all the details bar a small amount of info.

B can know they will be Paid By A. → We can ~~ar~~ setup systems where A + B specialise

A in Propagation + ordering
(validation).
B in Proof of Work

→ Over time, this can become (evolve) into a far more specialised system. Not A+B → But many layers of corporation (as we see with the Internet + Intermediaries) → This can fragment to all the layers of Redbach(?) + Spyder as I could not ~~sold~~ solve in 2002-4.

issues with China + other slow net's can be solved – area specialties + low cost labor US high end systems etc.

Ultrafast Consensus in Small-World Networks
Olfati-Seber(?) (2005).

– The transmission of a new block of batched messages is a phase change – Validators move from the last block to the next one and start a new ~~bl~~ batch.

→ 1000x more efficint than a regular complex network.

"Random Rewiring" – Watts(?) + Strogatz (1998).

– If we restrict the size to a small characteristic length – propagation is robust and efficient.

We end with a solution to Byzantine general problems. As long as the honest majority agree – then the system is robust in its simplicity.

Consensus is defined in this manner – and the idea has been touted as a way of solving
– sensor network consensus
– load balancing
– swarms + flocks
– sync. of coupled oscillators.

So, logging (SSH etc), SCADA Control and the overlays using this will be defined solutions and not mere ideas, but do not see that I can ~~pl~~ patent the time chain idea – BDO Capital Team.

Talked to Judith Ryan and Sebastian Stevens in Corp. Either I am not explaining my idea well (at all) or the base timechain is a ~~mu~~ not something that I could patent. They also have concerns – banking and issue licence – and do not think that you would be able to have a set of validators who are nodes that can come + go – partnership defined by a probabilistic lottery.

Alan Granger remain supportive (but seems to be politically isolated).

Roma Pala (?)/ Sidney Lim → Also do not think that a signed ledger of batched blocks as a timestamp is able to be patented. Blocks are discovered in a puzzle such as Aura() in a poison distribution. Each validator finds at a rate λ_i so, if we take a system of independent poisson processes – each finds at rates λ_i for $N_i(t)$. Once the system – ~~all~~ a validation node finds a solution to the Aura() puzzle – the winner sends to all other nodes – at the point where a batch block is solved and the proof of work puzzle in completed. The next is to broadcast to all other validators. So there is a small difference to a pure SIR model → as propagation is instant but validators will lose if this is to big → so they will expand to minimise the time

→ λ_i = solve ~~bact~~ block puzzle proof
p_i = propagation time

$p_i \to \phi$ as the validator invests and should be small. Where p_i is large the λ_i is reduced. So, for an ineffecient propagator ($\lambda_{i\ d} = \lambda_i + p_i$) where λ_{id} is the batch block as it is discovered and propagated to the majority of other nodes – not all. But – the more the other nodes – and the faster – the more likely that node(i) will win the puzzle prize (fees for validation of the block or batch or messages).

So, if $N_i(t)$ are (j) independent Poisson processes, $N(t) = N_1(t) + N_2(t)...$
$+ N_i(t) + ...$
$+ N_j(t).$

$\forall\ i \geq 1 + j$ the total # of validators.

j likely to end between 1000 to 100000 nodes.
→ all depends ~~of~~ on the propagation + network tech at the time.

Note – When (i) has sent/broadcast the puzzle to all other nodes (or rather a significant number of them) the process ~~seeks~~ ceases to be ~~inp~~ independent. As any other node knows of the solution, the choice is → mine and validate the old or new block. If the validator has more than 50% of the network – they can treat this as if they own it – but then they are also bound under strict law that will bind more than as a part of the group consensus.

For each miner (i) the process is thus (when the rules ~~w~~ are followed) a 2 party Poisson Process.

$N(t) = N_1(t) + N_2(t)$ where

$N_1(t) = N_i(t)$ original (discover Block puzzle).

$N_2(t) = N_j \neq i(t)$ all other validators combined.

$N(t) = N_1(t) + N_2(t)$ + thus $\lambda = \lambda_1 + \lambda_2$

So if (1) has 1/3rd of the CPU power and $p_i \cong \phi$ (fast propagation) then λ_2 has ⅔ total CPU. So if (1) discovers at a rate ~~>2~~ $\lambda_1 = 2$ and (2) discovers a solution at $\lambda_2 = 1$ ($\lambda = \lambda_1 + \lambda_2$) = 3.

So for each solution / it works on an average over time.

Now – this means if p is large the λ we obtain is a combination of CPU and propagation time → so we again have validators seeking to minimise p_i as the probagation time increases – so when we talk about a Visa size system of 50.000 Transactions or messages a second, we need now to think on the time to propagate → but Validators will have most or all of the messages – the hash of which (with very negligable collision – assume zero) rates is small SHA1 (160) SHA2 (256), SHA384(?) etc.

So, even a large (10MB plus – or even the XP windows limit of 4.3 gb) could be ignored with the batch read for only the hash → 4.3 GB as 160 to 256 bits
(8bit – bytes)

– See – Splitting (thinning) of Poisson Processes
– Merging independent processes.
– MSTAT Notes.

Note must add section on competing processes. The proess is more of competing SEIR (epidemic) models when blocks are discovered at around the same time.

lightning Download.

Have tested (see 2006 Ridges Notes) the XP downloader – Lightning Download
by Headlight Software.

See – www.lightningdownload.com.

Order (5696-2)

licesence name – CRAIG S. WRIGHT
licensenc code – WRIGH-J388U-GAHiR-P93JS-UPX2A

Will not end in PoC code – but in my Alpha test would be a good thing to add.

The software allows –

Splitting up to 10 natively – but I have changed this to 250 on a server Dell PC with multi CPUs + lots RAM

Recovery – parts, files + thus messages can be resent + broadcast from a stop/start.

With more RAM, a better network (or even to split pipes – see checkpoint load balance + Firewall software) I think that a server farm could handle a large batch today. A 4.3 GB (limit to Windows) message can be split ~~ing~~ into 250 Parts and on a 1GBe pipe or even (10x 100MBe) this could be sent in 10 seconds – at data centre levels.

at 50k messages a second, the hash matching on a CPU will be more important than the network.

→ So, – Matching Batches by Hash.
– Sending Messages.
→ Validating
→ Store/order.
– Puzzle solver.

→ Hash based databases will save the messages (see Alkami) and serve these but the chain of hashes can be created(?) on large fast host as needed – so many specialisations will occur as the system professionalises.

I expect this to be analogous to the ISP industry – Pegasus/Ozemail etc and how these became a few large (Professional) Systems.

If the system works as expected the validators and as a distributed Clearing House – a mutual where profit is earnt probabilistically. In effect – the system has a undefined-floating number of corporations acting as the nebulous competition based distributed Intermediary. Need to tie this into Mutual, association + early Partnership law as this is really a form or early corporate structure in the UK etc.

I do not see a scale limit in this – unbounded → as the system grows it should be able to handle IOT + SCADA messages and even logs (overlay network – need to add VERSION to code + script)

Using CASE statements – not efficient – But PoC is just PoC. when I move from the Alpha towards a more corporate model, I can look at something more efficient than CASE (threading vs. race conditions) – not sure how to have all the script constructs run and be tested in parallel ~~order~~ rather than Sequential, but Some of the ~~SASI~~ SANS guys know good coders. Planning to take GSE (.Net) exam next year. This covers security – but need to learn to make my code scale more.

Can look at the system as a series of micro-services + archetect this as web platforms are doing – as we scale – more likely to be able to gain from this approach. As ISP's started as home systems the distributed Intermediary network will specialise. Seing even ISP's now become more specialised (DNS, Email + Web specialists and not ISP's doing it all).

– See STAR Web Application Security Notes.

GREM – still like using the C+C model of Botnets for this. IRC is a simple method to bootstrap + can use ~~co~~ some of the malware code we analysed in this for good – I like the irony of creating a system that makes crime less profitable using code from criminals that is designed for cybercrime. Serves the dishonest bastards right that they are helping me destroy them and their scummy industry.

Honest nodes profit more! Can link this to GCIA + GCFA stuff as the timechain is an evidence trail + could end used as ~~a p~~ evidence in court.

Gerardo + Chowell(?) + Carlos Castillo-Chavez. (2003)

– Ch. 2 – Worst-Case Scenarios + Epidemics.

– See 2.2 Fully Meshed transmission network. The validators will seek to maximise connectivity

– So, merchants will want to know they are connecting to validator nodes – likely this will lead to the top "miner" getting more but this will be Balanced as some merchants will send to lower miners and others to a random mix. Best with a random ~~rand~~ range → pick 8 say and check from others → That is, send as the merchant to 8 validator nodes – and if 1000 miners these are 99% likely to send to all other miners in a single hop.

[diagram]

So Send to Some at Step 1.

Send a validation at Step 2.

– Now, the miner knows if they cheat the other miners will report and the merchant

The sender A sends Not to a PKI form address of the merchant → ~~Bult~~ But a new key.

$P(M)_1 = P(M)_0 + P_{(MA)}$ eg. So, the merchant could use a PKI identity and send a value (such as HMAC (X_A)) to have A + M generate a unique key and this is only a one time address or template. A cannot send a "double-spend" with the merchant not knowing and as A signs the template, they know that the "change" addresses are ones they set. In this A + M isolate + firewall identity completely from the validators/miners but the two contrasting Parties have evidence (admissible in Court) that is valid and can save the PII with less necessity to have all PII we do Now in a Visa world.

The "Worst Case" Senario in Epidemics is the best case in our network and a giant network forms. Miners are naturally incentivised to connect to as many other nodes (validators) as they can as this helps them get a solution out as well. The need to solve a puzzle of batched messages and to be paid to not find a solution to the puzzle but to have the majority in Consensus that you have a solution before the ~~others~~ majority of others is the key. A miner who has a majority of other miners will likely aslo

have miners on an alternate block that is just behind the 1st one defect as this is the ultimate Game Strategy.

t = 0 → M_1 finds block puzzle (50% have it)

t = 1 → M_2 finds a block puzzle (25% of miners)
M_1 (60% of miners)

t = 2 → M_2 gains some (30% of miners)
M_1 gains (70% of miners)

t = 3 → M_2 – loses some support (20% of miners)
M_1 – Gains (80% of miners

→ So as the miners see support for Mi → The SEIR model will allow the mathematical ~~any~~ analysis in time of what Blocks are still coming in a "Competing Epidemics Model"– see ~~Influence~~ Influenza Studies. (InfA vs InfB). We can show, if M_1 has 1% CPU power + M_2 has 30% CPU power that it will be in M2's interests if M**i** has managed to propagate blocks fast + now has > 50% support. So even M_2 with 30% CPU power fails if it does not ~~imp~~ immediately get a block ~~at dis~~ puzzle discovery out very fast – so no withholding strategy works – the validators are not going to end as the start and this forces professionalism + corporate nodes to start to form.

Not all nodes in our graph are equal. More – it comes to the economics of the system – The cost of the node is a mix in time of capital costs – fast internet links + servers to hold all the data (and in time this could end as exabytes!) and the immediate costs – staff, power etc. and as power changes (day by day and hours in a day) nodes will globally shift in strenght coming + going to maximise not revenue but total profit. This is an important aspect. The proof of work token now is an outsourced version of Aura's idea + one that has a floating value on some market.

– Need more time on "Bond Percolation processes" → studies of subsets of nodes

SIR is a simple idea but the propagation and validation time leads to SEIR models being more true.

As an overlay → the merchants will seek to maximise connectivity. At first the model PoC is Best as a single node structure where one Alpha software does everything but in time we will need to move towards specialised systems.

likely that separate Protocols for sending a message (merchant to the "miner" giant node structure), merchant validating the other (not sent to nodes) nodes have the transaction (and as a separate check to having sent to the miner directly – this is a separate protocol as

M → V_1 (send transaction)
send the full message

M → V_2 ($V \neq V_2$) is the hash of the transaction.

+ V_2 to M is a True/False.

True = V_2 has the hash of the message.

False = V_2 does not have it.

This forms a Bayesian belief network and the merchant can check (Really) fast if the message (with very low pre-puzzle batch) probaility will be in + accepted). We could have a miner attest → and this now adds legal liabity to a miner who is not honest. So – V_2 signs (may be for a fee) an attestation →

S_1 (No Tx) sig v_2
S_2 (Tx Good) sig v_2

→ Some Alert network for Validators could be created in time to have a have a dishonest Validator ostricised and excluded and this will "punish" ill behaviour better than a "legal" process meaning only the worst cases will need to have law ~~in~~ enforcement action – self regulating industry.

Need to find the ideal total(λ) or batch puzzle time – Note Poisson as an approx. So, with all the propagation → to do this and minimise races and forks of this chain, we need a large enough time (even on batches of 100's of Gigabytes as this may end in a decade it if works).

Must run senarios.

→ Cannot be under 5 mins.
→ 20 to 30 mins is too slow.

→ Simple is best. so simple way to have the parties know.

Values to test – 6 min | 10/hr.
– 10 min | 6/hr.
– 12 min | 5/hr.

Must also look at how we can set the puzzle to remain (→ period) about the same.

→ long game period

→ This is a Game. a zero sum game. if V_1 cheats. The same # of batch puzzle $ rewards must be done in the set (far larger ~~that~~ than puzzle period).

– try ~~2~~ 1 week
2 weeks
2x Month.
1x Month.

→ if you have 1000 blocks a month every month, then if V_1 cheats and loses a reward, there should still be 1000 valid batches that month.

→ Herteroscedastics – MA, ARIMA.
GARCH.

– Need to check models – to have a difficulty that remains close to the same.

Also, we can expect the system to be able to form a base layer Protocol – like Internet with separate messages + templates doing all forms of non-cash transfers.

This can be a separate overlay that is linked by a set of hashes or other puzzles in code. Forth should do this well and allows us to try the WANG B concept.

The time chain acts to stop all double spends, order (chronologically) messages and to (most importantly) pay the validators for this while being a micro payment system – in time, it could be that channels of thousands of a ~~se~~ cent at a time are exchanged where trust is limited to 0.001 cents per exchange and thus the

___

SANS – GSE Malware as a target.
– Need to add this into my study schedule
– AIM for a year away – 2008/exam.

___

GSE-C. Done – Can I do the (3) in 3 years – ?
Goal
– What will BDO allow for time/costs?

_____

SEIR Rayner ? others in MSTAT medical Stream.

[diagram] offline _also_.

So A + B can Send + B can check after a time.