Self-Optimizing Smart Cities Employing Decentralized Ledgers

Speculative Futures

By Joseph Juma

July 2026

July 17th 2026

In the future, smart cities (SMCs) could be thought of as a node placed on a map, connected to other SMCs both digitally and physically, forming a sort of triangulated mesh over Earth's surface. The physical connections will often be to material resource hubs, or other nearby SMCs if the resource isn't geographically constrained like water or food. Each SMC will have a self-managed local commerce which, in order to track its net economic activity, margins and inventory, will need a digitized central database that has a high guarantee of being tamper-proof. There's an argument here for block-chain based ledgers on consensus networks being natural applicable here for a few reasons.

Each municipality ("muni") could run their own block-chain consensus network (BCCN) with a local cryptocurrency managed almost entirely by automated systems, rather than being used by people. Human commerce would happen on a different "layer" than this cryptocurrency (CC) network, and any interactions with it by humans would be at very specific transaction points. This sort of thing enables a central system of the smart city to see the exact transactions of every autonomous unit in the system, and allows it to accurately calculate its own efficiencies, surpluses and shortages. The margin of the city over its operating costs are then dispersed as part of a securities derivative for bond-holders, then the remainder is used as a local UBI to establish a base standard of living without individual costs.

I believe this is preferable over a singular federal UBI, because when you have large, centralized items in the federal budget they place the lives of every citizen in the changing hands of a handful who might get bad ideas over many decades; actually, that's a statistical inevitability so long as a country can become dysregulated. It's also far easier to embezzle or launder money in large pools than smaller ones; so smaller ones become unattractive for grand schemes by well-resourced peoples. If one of these SMC systems fails, it also can request partial help from its neighbor systems, but they don't necessarily fail, nor do other far-away SMCs. This overall produces greater resilience of any UBI system.

Nearby SMCs may then communicate with each other, and employ a higher-order BCCN to conduct automatic commerce between each node on the larger map. This block-chain can be decentralized across a region at scale. The idea is that each SMC should use basic supply-

demand curves to balance surplus and shortages against those of other nearby SMCs, allowing for resources to automatically allocate between each node in the graph of the SMC network, while registering the commerce as economic activity of the SMCs; producing “mercantile cities” – cities that are themselves, engaged in mercantile affairs on behalf of their population for the sole purpose of stabilizing the local ecosystem’s homeostasis.

Extend once more, and you reach national level, with the national BCCN handling transactions that are primarily inter-regional and bulked via lightning-network techniques. By using this kind of graph-based SMC network enabled by BCCNs, we can expose resource databases, and net economic activity of automatic systems to SMC cores, and establish a cross-country network, allowing for automatic network updates and per-muni balancing to produce improved living and overall better economic efficiencies for the function of the country. Favorable countries could also do limited network connections to other countries, to allow for automatic commerce between foreign states. All being carried out by adaptive algorithms whose sole goal is the correct allocation of resource allocation for establishing a better underlying ecosystem for living; both environmental, and socially.

In addition, though less savory, graph commerce isolation can be leveraged as an adversarial soft-power dynamic, and hyper-connectivity as a beneficial one. In a future where every city, state or country has to choose between an idyllic life and sudden resource starvation, rebellion is unlikely so long as some threshold of SMCs are given a good life. Because of this, SMC systems **must** be controlled by a benevolent system, rather than by shifting elected officials, or private corporations.

The Block-Chain Consensus Network Design (July 18th 2026)

Diving a little bit deeper, I’ve given some thought to the BCCN design. A consensus network is advantageous because machines can freely join and leave without issue, and generally without harming the network, and the vast majority of the machines on this network should be smart infrastructure, which can undergo spotty connectivity for a vast variety of reasons. The network also doesn’t allow just any network to perform “mining” (i.e. the proof of work process of computing hash salts for generating a valid target hash for block encoding). Instead specialized mining machines (“drills”) would be the only ones allowed to mine on the network.

Drills would be custom hardware, with each unit being federally or state certified, and coming from no less than four different major manufacturers, who, ideally, don’t share a supply chain. These devices would have on-board ASICs, and although they might run an operating system, it wouldn’t be something most non-programmers could call one. Each of these machines would be tracked, and dispatched to either government agencies, private companies, utility providers, or other established trustworthy institutions, who would need to place them in a secure location.

Each drill would have an internal binary-tree with ten-levels, and 1024 leaf-nodes. Each leaf-node would be an asymmetric key-pair, but only one would be unencrypted at a time. Each

parent would have a symmetric encryption key on it, and every level above the leaf layer would have increased security. This would allow a compromised machine which had its keys dumped to only have it's *current* keys dumped. Some security measures could even include requiring hardware devices to be plugged into the drill, that each share fragments of the encryption key and so it's literally impossible to unlock the higher order parts without having multiple specialized physical devices all at once.

These drills would use their current active key-pair to *sign* a given block, endorsing it as a candidate for the new canonical block in a block-chain. Also, every drill would engage in a medium-low difficulty proof of work mining operation. A block would only be accepted once a correct nonce (i.e. hash salt) is guessed, and also that block has received signatures from more than 67% of the high-trust drill nodes on the network.

When a new drill joins the network, it has a reputation that is increased so long as it's actively contributing, and it remains on-network. If a given block is confirmed once every two-weeks, we might make it so a drill requires 4 cycles to reach full "reputation". Signatures by partial reputation nodes are not considered as strongly as those from high-reputation nodes. This is why proof of work (PoW) stays around at all: if a malicious node (i.e. a stolen and compromised drill) joins the network, it has to survive on the network while being active for two-months to gain authority. Because every node on the network is identifiable by a federal certification identifier, and all nodes must be physically in the same regional, this and the other security practices make it very annoying for an attacker to deploy malicious drills.

If drills are deployed by the Federal or State government to a muni that also include adversarial government organizations (i.e. orgs who are naturally adversarial to each other), it can improve the odds that they won't tamper with their drills to aid each other. This creates a sort of "muni-level checks and balances" through the block signing vote.

The block-chain ledger then becomes an easily queryable ledger that gains tamper-resistance proportional to the number of blocks ahead of a given record. Trying to recompute the hash for 1 block might be feasible; but recomputing it for 10 blocks can be extremely expensive. This allows for the multiple adversarial organizations to each act as ledger authorities, while not being in agreement, and also produce a ledger that can be almost automatically audited by anyone; including by AI or other software.

Now the cryptocurrency is distributed by some program or other to every node on the network that isn't necessarily a drill. I've not given much thought to how this works – but the idea is going back to good old fashioned barter economies. A given system can pay another system in-exchange for some amount of a resource, and those transactions are logged in a single unit (the cryptocurrency) on the ledger. This allows for standard conversion rates (e.g. MuniCoin to watts) to be published in each block, allowing for software audits on the exact behavior and commerce between nodes on the smart infrastructure "grid".

Even if a node goes offline, every ledger-holding drill still tracks its operative behavior, because every node is assigned a wallet ID, so its net operations don't require it to be online except to communicate its transactions. If other nodes are doing transactions with *it* and it's offline – then it's net transactions are still visible. This is pretty nice from an uptime perspective.

Next Steps

The idea is that once we have a BCCN that can look at everything in a muni, we can start treating a muni as having an "inventory" of resources, but also identify inefficiencies and balance things automatically using software and smart contracts (i.e. programs that can be deployed to execute and rate limit behaviors on the infrastructure). Add in future scifi ideas of job automation now.

What could happen is, we still have labor happening but it's being handled by machines. Those machines produce some kind of economic product, and then muni-bond holders are paid out of the robotic laborers they funded; with the remaining going into a UBI pool for that given SMC or muni. That's then used to establish a UBI by funding things like community housing, roads, infrastructure repairs, public services, and so on. All enabled, in-part, by a thoughtful re-application of block-chain and consensus network technologies.

The really interesting thing about this is that it's 100% possible today. Besides robotic automation, every piece of technology that I discussed already exists – or could be built within six-months with off-the-shelf components or open-source software libraries. The real issue is human incentives, but also that nobody has "had" this idea. I'm not sure if the economics check out yet, but they make sense going into the future – having *every* muni be able to make sure it was operating efficiently, deploying self-repairs and balancing its own budget with the efficiency of a program is pretty compelling as an abstract idea. The question is where the asperities lie.