2023-10-31st ACeS - Resource Engine problem - How to remove the Rules for Rulers

This has also been [posted to Github](#) and also available as a PDF.

> TLDR: This proposal outlines an "**A**bundance **Ce**ntered **S**ociety" (ACeS) for a Mars colony, where resources are managed by a computer based "Resource Engine" rather than traditional politics or market economics. The goal is a post-scarcity society without money where everyone's basic needs are met. However, a challenge is preventing the "Rules for Rulers" effect where leaders become corrupt.
>
> This proposal suggests open source development, transparency, distributed decision-making, and other measures to reduce centralization of power. It poses thought-provoking questions about preventing corruption of the Resource Engine, such as whether leaders are necessary, how to structure voting, and balancing regional differences. In conclusion, it invites discussion on the best structures and safeguards to create a fair, resilient Resource Engine not exploited for gain.

Update:
I've been thinking about this for a few weeks now and there's a few things.

Firstly, I made some assumptions in my original thesis around how humans like to gravitate towards having a leader. I do think this is partly the case (and I'm sure Jordan Peterson and the crabs would agree with me). But maybe all we need is some system that can break a tie breaker set of votes. That could be an algorithm that simply flips coins.

Secondly, I've read almost all of the updated Dictators' Handbook (well, listened to the Audiobook) and the good news they point out is that there's not been a strong democracy that's turned into a dictatorship in the last 150+ years. A democracy is stable and often tends towards becoming more democratic. There has been some notable attacks on democracy with Trump in USA, Duterte in the Philippines. But most seem to have weathered those attacks.

One obvious thing I would say is that **no Sociopaths** / Psychopaths should be allowed to gain anything approaching a leadership position. We've seen what happens when that occurs and it's an attack on the system in order for them to gain more power in a system destroying way.

With such thinking done, I've got a first draft proposal for how to minimise the Rules for Rulers effect on an ACeS Resource Engine.

1. Have a large number of people who can understand and submit code updates. My suggestion is to have a minimum of 10,000 people or a reasonable percentage of the population who are explicitly trained and capable of understanding at least important parts of the system and what updates will do. For most of them, maintaining the system should be their primary focus. They would be an "**ACE**" or Abundance Centered Engineer

2. Have **a multi-stage testing, acceptance and rollout process for updates**. This likely starts with the pull request (a technical term for merging in new code) triggering a large number of acceptance tests. From unit tests to simulations of past data, expectations of the future and handling of likely disaster scenarios. If it passes those then it's soft-notified to anyone listening and a hard notification sent to a handful of ACE developers, some might be randomly selected and some might be specifically selected because they've dealt with code changes in that area and should understand it (they might also have some feeling of ownership). The ACEs would be given 48hrs by default to review the code changes (but can ask for more if needed). If enough vote for accepting the change and there's none that outright block it (causing a formal review processing) then it can proceed to a 2nd round of change reviews whilst also being deployed in a small geographical area (if appropriate), allowing the equivalent of an A/B testing deploy. Should the data show there's no obvious regressions, then it can be rolled out to greater and greater areas. This process would especially apply to better dealing with edge cases, emergencies or maybe something like tweaks to a transport algorithm. Maybe it's better able to determine if cargo should go the fastest path (e.g if you are an ambulance), shortest path (if heavy freight) or safest path (especially if chemical, radioactive or biologically dangerous).

3. If it's a large or **breaking change** e.g it triggers **something you can't easily reverse**, like major building code changes and thus ties up a lot of resources which others might have other plans for. Then it should be tagged and flagged as such and go under a lot more scrutiny. It might go for an ACEs vote or taking a **participatory democracy** stance of **those most affected should have the most say**, then it could require a broader vote, including everyone, before a rollout is allowed.

The aim is to work a bit like Wikipedia, as a community of people working on a commons, for the greater good of all.
The aim is to allow the smaller, more obvious and easy to accept updates to be

accepted more easily and for anything more important or not easily reversible to have more scrutiny and also public acceptance when needed.

There's a bunch of feedback steps whereby people can make suggestions on how to do thing better. You could start with a proposal first, then have code itself and both can change and be refined.

Anyone can listen to suggested merges and look at their process (including who's reviewing them) and can also proactively review them themselves.

There's no single leader. It's what some would say is a leaderful system, more than leaderless system. Each ACE should uphold a set of morals and ethics. Ideally we should be able to test when those are being breached in large ways, or are simply shifting with the times (e.g as Mars gets terraformed or if we develop the ability to digitise our consciousness and live inside a computer).

There should be ways that anyone can make suggestions or proposals. Especially a feedback mechanism for pointing out issues and problems.

Pretty much anyone should be able to do a course on being able to understand the Resource Allocation system and to make suggested changes and work towards becoming an ACE.

You might have people who have those who make a code proposal and also those who champion them, sometimes that'll be the same person, sometimes it'll be different. This is probably the area that needs the most scrutiny. You don't want changes which will negatively affect a large number of people but help a very small number, especially those associated with the changes, to be able to simply convince the reviewers it's a good idea.

There should be some way of having a general consensus on things that matter and what a mutually exclusive direction would be and why it shouldn't be pursued.

I like the way Alex explains how there's attractors and shelling gradients In Episode #84 of the _Boundaryless Conversations Podcast_ titled: **Gardening Platforms and the Future of Open Ecosystems with Alex Komoroske**. The idea is that if something works well, then people will gravitate towards it and to think of it as **Gardening**, not building.

If there does need to be some sort of structure to the system then I'd suggest the **Constellation Model of Collaboration** as explained by Mark and Tonia Surman. Having a stewardship group that can set a shared vision. But they can't control the resources.
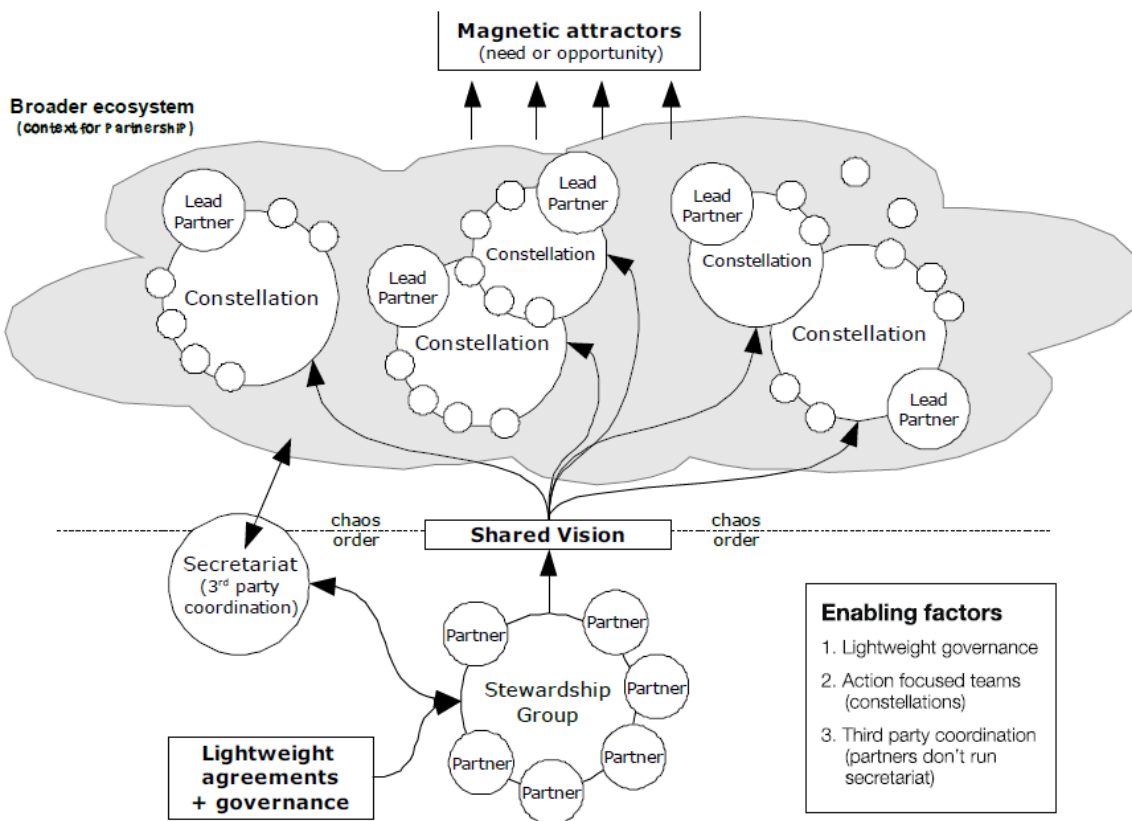
Note that obviously the above proposal is very basic. As we get closer to actually enacting such a system we'll want to come up with multiple other governance proposals, select the best and then detail and build the system, refine, iterate, etc..

Original challenge / post:

I am currently grappling with a challenging problem which I think will affect a lot of humanity in the future and I'm looking for some help.

The problem is:

How do you prevent or reduce the effects of the Rules for Rulers in an Abundance Centered Society

Especially in regards to the Resource Engine.

In other words. How do you prevent corruption?

Some context.
I explain what ACeS or an Abundance Centered Society is in episode 1 of the Abundant Mars podcast but I'll explain it shortly for those that aren't sure.

Rules for Rulers

The Rules for Rulers is the term taken from the [CGP Grey video](#) which is based on [the Dictators Handbook](#).

The rules are:

- Rule 0 : Without power you can affect nothing
- Rule 1 : Keep the key supporters on your side
- Rule 2 : Control the treasure
- Rule 3 : Minimize key supporters

There's more nuance and detail in the Dictators Handbook which I found out recently revised their audiobook version to even include Biden getting elected and the reign of Trump, the Arab Spring era and other things.

Really it comes down to this. If there's a group of people who get together and pool their resources in some way, then we usually allocate a leader to decide where those resources go.

This applies to large governments and companies down to tiny local councils or even neighbourhood watch programs.

Some of the key points:

If you are a ruler then you can't rule alone. You need people to help you. Those people will want money, kickbacks and favours for helping you.

If you are in an autocratic dictatorship style country or company then only you as the leader and a small number of people are needed to keep you in power. In a country that might be the head of the military and the head of the treasury.

In a company it'll be the people on the board of directors. Those are the people who can fire you.

In a democracy you will usually have people vote for you. Depending on the system that might mean you need to get only 1% of the votes or 51% of the votes from the population. That difference will affect how you reign and what happens to the majority of people.

If you have only 10 people who you need support from in order to stay in power, then you are likely to screw over the remaining thousands in the company or millions in the country to appease them, because if you don't, someone else will.

So overall, **the more people who need to vote you in, the better for the majority of people who you can affect.**

The less key supporters a leader needs to be elected, **the higher the amount of corruption there likely is.**

ACeS

An Abundance Centered Society or ACeS is my term for what you might know as a Post-Scarcity society or RBE (Natural Law, Resource Based Economy as described in [Zeitgeist Moving Forward](#)).

Through the use of things like a Cradle to Cradle, Circular Economy the resources we use are kept in a continuous cycle. Just like the water cycle or Carbon cycle, we have cycles for all the materials.

This means we track technical and biological materials, keep them separate and reform them into new products over time.

When your phone gets old we should be able to break it apart into constituent elements and reform them into a new one, or maybe into the circuitry for a drone or space ship.

You drink water and after it's passed from your body we'd have systems to process and filter it, extract the urea and as they say on the ISS, [yesterday's coffee is tomorrow's coffee](#).

Through Access Abundance we make things built to last and also shared. So you need less things built and are sharing by default.

This leads to enough abundance that there's no longer a need for money.
So it's a post-monetary society.

There's a bunch more to it, but I'm also applying this to a base on Mars. Mars has a lot of extremes. Very cold, barely any atmosphere, not a lot of energy (e.g only 43% as much sunlight), no existing ecosystem. It takes 7 months to travel there, you can only go once ever 2ish years and it costs a lot to get there. Those extremes help highlight constraints in the system you want to run.

Resource Engine

In the ACeS system I'm proposing we run on Mars (and eventually Earth) we will need to deal with resources in an efficient and effective way.

Right now we have politicians and politics. These are affected by the rules for rulers and aren't very effective and often bad behaviour and bad policies can be good for the leader, even though it can cause the whole society to collapse.

That's why I thought we could replace it with a Resource Engine.

My recommendation is a next gen distributed ledger system.
It's transparent, anyone can read the history of the decisions it's made and maybe some info about why. It's able to make decisions with large amounts of data and it ideally should be somewhat neutral.

When we lived as hunter-gatherers there wasn't a lot of resources to worry about, just those you could carry on your back and where you as a family or tribe would travel to next in order to get food.
Then we developed agriculture and it used to be that kings and royalty would do a lot of the important decision making.
Then it transitioned to a market system.

As per Yuval Noah Harari's book [Homo Deus](#), we have transitioned between centralised and decentralised decision making systems over time and as per Dataism, it looks like we are on the cusp of changing again.

The Soviet Union attempted to run a centralised resource processing system. All the information went to Moscow, processed and the orders would be sent out.
This meant that someone in Moscow was making decisions about how much bread a bakery in Siberia or Vladivostok should be baking. They'd send orders to the granary to have them send the bakery the amount needed.

But they were dealing with human processing on old data that was often intentionally manipulated by those sending it for political reasons (see the Rules for Rulers at the heart of the problem). The end result was many starving people living in the USSR.

USA, the UK and now pretty much most of the rest of the world on the other hand uses a market system whereby the bakery makes it's own decision on how much to bake, based on the expected number of people who want bread versus cakes, the price of grain, how many people they have who can run the store, how experienced they are, the quality of the grain and more.

Whilst the Bakery hopefully takes into account holidays and other regular special occasions. That bakery is unlikely to know that maybe 10x more people are expected to come visit tomorrow for an unusual one off event and so it should work extra hard. Or that people have just watched a special warning about how carbs like bread will make you fat. Or that maybe with shifting demographics there's now a greater demand for rice based foods, not wheat based ones.

The ACeS Resource Engine is designed to take over the current market system to some extent and replace it with algorithms.

The way I envision it is that IoT (Internet of Things) sensors, plus other data is fed into the system and it can then make decisions based on that.

Some Mars base examples:

In order to grow food you want a Vertical Farming, Aquaponics system. They are a form of hydroponics grown food in multiple layers but which also uses fish. The fish, usually in tanks on the lower levels eat the left over scrap foods and other things. Their poop then makes for good fertiliser that can be sprayed onto the roots of the produce. Be it lettuces and tomatoes, Apples and Oranges, wheat or rice.

However if a Vegan community forms then they might collectively decide they don't want any animals (except special pets) within the boundaries of their area nor involved with their food.
So they want just a plain hydroponics system, no bees, no fish, no chickens, nothing like that.
The nutrient flows for that hydroponics system will need to include a lot better processing and use of human feces and separate, non-worm based composting systems.
You now need to route different foods, different pipes and different systems to different areas of the base.

You might however have a Vegetarian cohort somewhere else that will accept their food having nutrients provided by animals, but still don't want those animals in their area.

You might also have scattered throughout the colony people who want to do extreme endurance activities. Be it trying to climb the highest mountain in the solar system, being Olympus Mons, which is about 3x taller than Mt Everest. Or to travel for really long distances in a space ship.
Those people need calorie dense, long lasting foods. They'll not actually spend much time at the base but want a bunch of food that can last years. So you need to food processing and storage facilities for them.

You'll want those anyway for feeding the people returning to Earth, for storing spare food for emergencies and so maybe most of the endurance athletes can eat food that's being rotated out of the emergency storage supplies.

There's other types of people with different requirements.
Families versus solo. People who want to live in trees instead of easy to clean, rounded rooms. The water and resources needed to grow trees on Mars is incredibly hard to get. But if the normal rooms need plastic, then that's not something you can get on Mars.

So we have a situation whereby we have different cultures, groups and identities have different requirements.

There's so many more examples I can think of that the Resource Engine has to deal with.

Just getting Oxygen to breathe is a lot of effort. You can extract small amounts from the atmosphere and larger amounts by mining ice from the poles, melting it and splitting it into hydrogen and Oxygen. But that's energy intensive and getting the ice from the poles to the main base I would assume would be near the equator of the planet in order to have better sunlight would also be a lot.

So if it turned out that a reasonably large amount of Oxygen was being lost through leaks from windows and doors then you might want to update the building code to triple glaze the system and add a whole new metal shell layer around the buildings in order to capture more O2.
Whilst there's a fair amount of Iron and sand on Mars, it's still limited and trying to mine it, process it, melt it down and create steel beams or glass panes is also very energy intensive.
Maybe later on it's found that special coat of paint is all that's needed to keep the O2 from escaping and the extra shell layer is removed and the steel melted down and used to make new buildings.

Of course balancing energy needs will be one of the big challenges for the resource engine.
So that example covers building codes and a tiny touch of trying to balance different resource constraints.

Another example is changing human behaviours based on what there's an abundance of.
Maybe there's a system malfunction and 80% of the Aquaponic fish die off. There's also a bit of a heat wave. So there's more fish than can be kept in cold storage. You

used to get a fish a week and now ideally everyone will be eating multiple fish a day for the next few days.

But then you won't be getting fish for a while. That's the sort of thing you'd notify people about.

But what about general seasonal variations in fish stocks?
One option is that when people go to open up their app to order food the fish meals will be either big and prominent when there's lots of fish and much more hidden away when there's not very many fish.

Another option might be around transport.
Ideally an architect could define a residential and industrial or commercial hub and specify that people should be able to get transport between the two that's relatively fast and they shouldn't have to wait more than 5mins.
The resource engine can then work out if that should be done using a road with a bus system and occasional self-driving cars which are waiting like taxis. Or maybe it's done with a hyperloop or something resembling an elevator you get into.

You could also have building codes also specify that pedestrian and cycling access should also be placed along.

If there's recently been a crash, then instead of suggesting people take the train, you might suggest they take a bike and ride there.
No, I don't know what riding a bike on Mars will be like or if it's really even possible. Maybe a zip line will be better.

But the idea is that the Resource Engine would replace the need for politics and politicians. Instead people would argue over which algorithm is best....
But it should be the best for the society as a whole, not the best for a small group of people.

Overview of the Resource Engine

So there's a few different things the resource engine has to do.

We've got requirements regarding how many people are on Mars and what they are doing.

We want to provide at least the necessities of life, for free to everyone. That means providing them with air, shelter, water, food, healthcare, transport, education, entertainment, etc..

We've got building codes. Everything from the minimum height of a doorway, or size of a room, to how the filtered air is dealt with, the size of the water pipes, the thickness of power cables, if you use Fibre optic cables or Wireless internet with wifi extenders. How many people a transport hub should be able to handle. All sorts of things.

You've then got regional exceptions. The Vegans or people living in trees, or families wanting day care and family friendly parks.
No, I don't think there's really going to be people going to Mars to try and live in the trees, it's just an extreme example.

It's also got the ability to influence human behaviours to help smooth out resource access and prevent people hitting certain limits.

The Resource Engine will need a lot of smarts and algorithms.

Some of the algorithms will be based on fairly strict logical coding that's easy to reason. If there's more people, we need more places for them to live and more food production, water, air and other treatment facilities, power generation, etc..

Some will be based on weighted preferences. Say we need more power.
Should we build more Solar Panels or Nuclear Fusion reactors?
Maybe there's a limited supply of nuclear fuel, so we need more Solar Panels with power storage.

Ideally we could look at the distributed ledger and see what decision it made and why. It had these constraints and so this option was the best.

When it goes to build is it Solar Thermal or Solar PV? Where is the Solar farm built? Did it take into account expected base expansion so it's not right next door. Was it up higher for better solar incidence?
If it's too far away then what about the electrical transmission line losses?
You can't spend much time as a human outside on Mars due to the cosmic radiation. So we need to have robots doing the work. Ideally they'll be programmed and mostly automated.
This means that the Resource Engine could identify the need for more power generation, propose a solution and if accepted, go and task robots to make the solar panels, take them to the site, set them up, wire them up and connect it all to the grid.

If we find some Geothermal vent then we can see when it makes sense to build there, but might have to design and build the parts ourselves for the first couple of power plants.

Other aspects might require some AI models, most likely around predicting human patterns. Like how many people does it predict will be in the next wave of ships? When will the highest load be on the transport system? Then we can look at implementing ways to stagger shifts to reduce that load to prevent having to build more transport.

AI algorithms are usually a lot more opaque. It's hard to know why it predicted a certain outcome. Hopefully some work can be done to help provide at least some idea of the weightings involved.

A lot of the Resource Engine needs to interact with humans. It's unlikely it can actually design and build a new shell for the buildings all on it's own and we probably don't want it to. We'd want it to highlight issues, maybe provide some suggestions but we will still want humans in the loop.

There's a bunch of stuff I've glossed over. Like if there's limited studio facilities then do you schedule more time for creating educational or entertainment content?
Do you create new kids shows or new blockbuster action movies?
A lot of that can be derived from the actions, behaviours and demands of the people in the colony. Some of it can also be inferred by their demographics. Obviously if there's a lot of people starting new families then there will be a greater demand for Martian based kids shows that can help explain things more specific to kids growing up in such a life, versus what you'd learn on Earth. Dinosaurs are cool on Earth, will they still be on Mars? Knowing how to tie your shoes is important on Earth. Knowing how to get into a space suit or use an emergency oxygen mask will be important on Mars.

Heart of the Problem

We can now start getting to the heart of the issue.

Obviously with all the capabilities that the Resource Engine has, that comes with great power.
As we know from Spiderman, **with great power comes great responsibility.**

How do we ensure that what it's doing, it's doing in a fair way that's not exploiting the majority for the gain of a small minority?

How do you ensure that the Resource Engine is helping the colony to thrive and will help it keep thriving in the long term?

But also, how do we know that there aren't people programming it to do things in corrupt or manipulative ways, like would happen by those affected by the rules for rulers?

The way I imagine it is this.

There's **people who program the resource engine itself**.

The codebase should be openly available to everyone and anyone should be able to submit code change requests.

If the changes are better, then they could even be automatically accepted, but only applied in a small region first before being rolled out over the planet or even solar system.

Ideally changes made to the resource engine go through a battery of tests. From Unit tests to simulating it on both previous actual data and future possible data, plus potential emergencies.
So changes aren't accepted if there's a regression. Although it's likely that someone's attempt at getting the transport system working faster means trade-offs in other areas, like a reduced priority for ambulances or other emergency vehicles.

But as we know from lots of commercial and especially open source projects it's likely there will be a core group of developers who look after the system.

There will be **entire teams who help develop different AI models** and other teams which **provide data** that the system uses.

Then there's the people who set specific location resource constraints. The Vegan boundary area or those who influence cultural change.

There's also those who set the building codes, the architects who designate areas and design new buildings or engineers developing new systems.

The developers and all the other people involved have different ways they can influence the system.

In some cases, they might intentionally manipulate things for their own gain, in other ways they might do it unintentionally because of their own biases, or because they are being influenced by other people.

Humans often gravitate towards having leaders or key decision makers, even in most group settings. There's a few different biases at play, like letting people specialise, being lazy, information processing limitations and also the fact we are a mix of pair bonding and tournament species.
There's also people who are psychopaths and thus lack empathy, although like nearly any human trait, that's somewhat on a sliding scale.

So even in groups where we try to let many people have an influence, we'll often still have a group leader.

This person inherently gets influence and power.
Now you get into **the rules for rulers**. If they want to keep that power and control they have to help give the key supporters what they want, within their capabilities.

Now, I think that a Resource Engine running in Capitalism would be rather bad.

The people who program the Resource Engine and accepted changes would likely be fought over or themselves would make changes that easily give them lots of power. They'd want to maintain that power so would make it harder for other people to see the code, to propose changes or have them accepted.

They'd become a monopoly, controlling the resources of the society.
If they don't do that themselves, they'll likely be forced to by someone who will come along an force them to do their bidding.

In an ACeS we are, as much as possible, guaranteeing people have the necessities of life, for free, to everyone on the planet.
The aim isn't to accumulate money.

Although it's likely that people will still attempt to accumulate power.
So again, the Rules for Rulers will apply. Even if the the outcomes are hopefully less dire as they still want to maximise thriving for the society to some extent, but will still put in exceptions.
Maybe they give themselves greater priority than even emergency vehicles.
Maybe they ensure they get the best locations to live and work.
But maybe they also try to hoard access to resources. They might reduce the amount of emergency supplies, and allocate the resources instead into luxury items for themselves, to the point it means the base might not survive a Mars quake, asteroid strike or other disaster.

Like [Michael Albert explains when talking about Parecon](#) (a different alternative economic system) a big question is **how do you prevent people from undoing all the safeguards and changing the system to benefit themselves at the cost of others?**

That's the challenge I'm facing.
That's the question I've got.

At the heart of it, I think is a question around how do you prevent the Rules for Rulers. Or, how do you reduce the effects the most?

Some thoughts.

Firstly, we would try not to have a single person who's a leader.
We would want a relatively large team, like over 100 people who can be a part of any primary dev team and could make changes.

This makes it harder to manipulate things to your own advantage.
If there's major changes to the Resource Engine, then they can be taken to a vote by everyone. If you need lots of people to vote on the changes, then you'll need to

pander to the majority of people, not just the 100 or so primary developers.
We'll need good systems to explain what the changes are and how they affect people. Does it mean more people but less amenities. Or less people, but more work, or some other trade off?

The current status of pretty much all the systems should be something anyone can check. Energy, Food, Water, transport, computing, etc.. What is the load, the capacity, the storage, etc...

I talked about Unit Tests. What if people manipulate the system by weakening the tests or changing their expectations to be too high in areas that doesn't matter to deter people who would otherwise submit changes which fix key issues causing stratification?

Do you now have to have a vote on changing the testing code or simulations? How many people in the society are going to understand properly what they are voting on?

To reduce the likelyhood of problems, we might decide that people should vote for changes to the Resource Engine. Maybe there's a few different proposals for how to deal with problems, what the tests should be or more likely, what major projects to work on.
Everyone gets a vote, or maybe even 3 votes, but you can also allocate your votes to someone else who you think is paying more attention and is more likely to represent your interests.
Now we've just re-created representative democracy in a way and now those people would be key supporters that the people who want power would attempt to influence. There's ways to reduce this, by not letting people know that others are allocating their votes to them. But then you might have vote loops. You allocate your votes to someone who allocates their votes to you and then what happens?

Lets change to another example. That of projects to build.
Imagine we are beyond being a self-sustaining colony. There's millions of people on Mars which is a thriving civilisation.

You are on your way back from work and you get a notification on your phone. It's time to vote on a major project.

Lets say there's 3 main projects:

1. **Terraform Mars**, including bringing in new water via redirecting asteroids into Mars' orbit and then carefully bringing pieces down to the planet
2. Build Space ships and the systems to **colonise the gas planets** of Venus and Jupiter

3. Build **a massive generational ship** to go to another solar system. Like the Behemoth / Medina station in The Expanse.

They all involve developing a space industry and might also benefit from a [Space Elevator](#) or at least [Sky Hook](#).

Firstly, who proposed these?

Can anyone make project proposals? What's needed to get them accepted as a potential project to be voted on?

Who benefits?

If you are a core developer or someone who is trying to rule and you have a strong preference for one of the projects, it's likely you could sway the votes, or manipulate the system so your option is preferred. With enough power you'd have the decision made behind closed doors and your selected project simply announced to everyone without anyone else knowing about the other options even being a possibility.

I hope you get the idea.

I was hoping that a Resource Engine would stop politics. But I'm not sure of the best structure which would ensure that and couldn't be easily manipulated to have the various safeguards removed.

Maybe each equivalent of a town or city could fork their own version of the Resource Engine and have it's own system running. You might have some dictatorships in some areas, but hopefully overall you can get the diversity needed to have resilience, whilst also converging towards good enough.

Maybe you can let people vote to roll back changes. But what if there's an important set of changes because of a new threat, like a large incoming meteor and we have to build defences, but people roll back the changes which propose building railguns, nukes or any way of defending against it because they think it's fake news designed to make it harder for them to work? Like in the movie [Don't Look Up](#).

I don't have the answers.
I'm guessing it's mostly about having some decently good enough structure to spread the power out enough. Like no single person has complete power and instead a large enough number of people have to sign off changes. Have very high amounts of transparency.

Also having some active measures in place so we can prevent corruption, have the ability to change those who potentially have power, etc...

But, maybe there's major assumptions I have wrong. Or structures I'm missing.

I'm looking for quality discussions and debates.

Should we use a Resource Engine?

Is there better ways? What do they look like?

What structure should we have? What safeguards? What systems in place to update and correct the safeguards?

Do we have to have leaders? Are the negatives of leaders almost always worse than the positives, and is that inevitable?

What are the better options than having leaders and how do you prevent them from being corrupted?

If you have any ideas, contact me.

Contact details:

Name: **Michael Kubler**

Twitter: https://twitter.com/kublermdk

Email: michael@abundantmars.com

Website: https://www.kublermdk.com/

Keybase (identity validation and also encrypted comms): https://keybase.io/kublermdk