redhat.

When You Shouldn't Use Blockchains – or Smart Contracts

axel simon, Office of the CTO

axel@redhat.com

OpenFintech Forum
2018-10-11

Presentation agenda

- What are blockchains? (and what are they useful for?)
- Are we getting overexcited here?
- When not to use a blockchain
- What are smart contracts? (and what are they useful for?)
- What to think about when considering smart contracts
- What about *trust*?

What blockchains are

(an attempt at a definition)

BLOCKCHAIN: THE NEW SOLUTION

(TO ALL YOUR PROBLEMS)

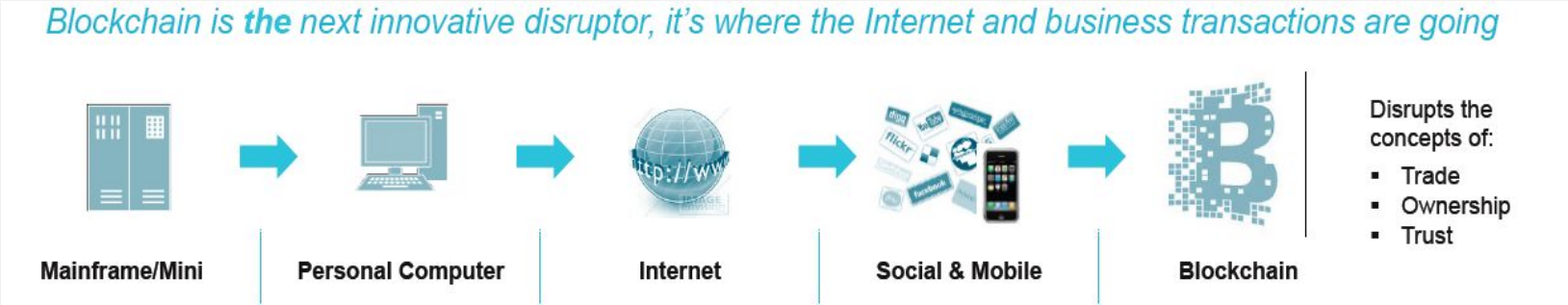
A **blockchain** is a

- distributed
- replicated
- peer-to-peer network of
- databases allowing
- multiple non-trusting parties
- to transact without a trusted 3rd party intermediary and
- maintains an immutable record of time sequence records.

BLOCKCHAINS:

THE INTERNET OF VALUE?

- A hyped-up concept, with some real properties.
- Blockchains (for different uses) rather than blockchain (one to rule them all).
- Ecosystem mostly open source.



What are they useful for

HOW ARE THEY USEFUL

REPLACING A (FALLIBLE) AUTHORITY

Replace a (fallible) authority with a distributed system where every participant reaches agreement regarding "the truth" and no one can take control.

Distributed authority & consensus mechanisms.

HOW ARE THEY USEFUL

A COMMON RAIL

Blockchains can also be envisioned as a common information rail (*or a message bus*) that connects various entities.

Each organisation latches onto the rail and follows it.

This idea is at the heart of most of the powerful BC use cases: reconciliation with a single entity (the blockchain, the rail, the message bus) rather than a multitude of different ones with their own approaches, protocols and quirks.

Another common rail: the Sky-Line in BioShock Infinite

HOW ARE THEY USEFUL

TRACEABILITY

Immutable series of time-stamped events.

Great for traceability.

Useful for logistics, healthcare (one patient record easily shared), financial transactions, etc.

CC Twentyfour Students

Things to keep in mind about blockchains

THINGS TO KEEP IN MIND

A FEW (INTERESTING, DESIRABLE) CHARACTERISTICS

Fundamentally, blockchains are peer-to-peer networks: this has consequences (good & bad).

Everyone has a copy of the data.

A solution looking for a problem?

A SOLUTION LOOKING FOR A PROBLEM?

Blockchains are not a silver bullet.

What's pushing for adoption?

(and are we getting overexcited?)

ADOPTION DRIVERS

NOT ALL DESIRABLE

- Any application with multiple users which stores data can be a blockchain-enabled application!

ADOPTION DRIVERS

NOT ALL DESIRABLE

• Any application with multiple users which stores data can be a blockchain-enabled application!

• Conferences and “gurus” saying “embrace blockchain now, or go out of business within six months”

ADOPTION DRIVERS

NOT ALL DESIRABLE

• Any application with multiple users which stores data can be a blockchain-enabled application!

• Conferences and “gurus” saying “embrace blockchain now, or go out of business within six months”

• The tech is hard to understand, those who do tend to be techies
(and don’t necessarily care about business impact)

If you only remember one thing:

before you ask “which?”, ask “why?”

Reasons not to deploy a blockchain

WHEN TO SAY NO

IT'S EASIER THAN IT SEEMS

- when
  - it's just you: no group/multiple (dis/mistrustful) entities, no consensus, no point
  -

WHEN TO SAY NO

IT'S EASIER THAN IT SEEMS

- when
  - it's just you: no group/multiple (dis/mistrustful) entities, no consensus, no point
  - it's a group, with a central/third-party authority, but it's doing its job fine.
    Also: when the process is a single/few step(s), not multiple steps.
-

WHEN TO SAY NO

IT'S EASIER THAN IT SEEMS

- when
  - it's just you: no group/multiple (dis/mistrustful) entities, no consensus, no point
  - it's a group, with a central/third-party authority, but it's doing its job fine.
    Also: when the process is a single/few step(s), not multiple steps.

- when using a centralised database works fine
  - think “where is the data” (location), not “who owns or controls it” (authority)
  - in some cases, consensus (not control) about updating it is enough

-

WHEN TO SAY NO

IT'S EASIER THAN IT SEEMS

- when
  - it's just you: no group/multiple (dis/mistrustful) entities, no consensus, no point
  - it's a group, with a central/third-party authority, but it's doing its job fine.
    Also: when the process is a single/few step(s), not multiple steps.

- when using a centralised database works fine
  - think “where is the data” (location), not “who owns or controls it” (authority)
  - in some cases, consensus (not control) about updating it is enough

- when you're the only one who benefits and it will just be costly / painful to others: you'll never really convince them

WHEN TO SAY NO

OR AT LEAST THINK LONG AND HARD

- Privacy by design: if you are going to store personal information, think twice.

- GDPR compliance team will thank you

Things to keep in mind with smart contracts

“SMART” “CONTRACTS”

NEITHER REALLY ONE OR THE OTHER

You are probably thinking of contracts like this:

Think programs.

The programs are sent to the blockchain network.

They are executed by blockchain nodes.

“SMART” “CONTRACTS”

LOTS OF NEW PROBLEMS TO CONSIDER

Smart contracts on multiple nodes, lots of ~~complex~~ interesting legal problems:

- legal validity of code
- responsibility (initiator? executor? third-parties?)
- jurisdictional (where is the program actually running?)
- oracles
  - timely?
  - trustworthy? (assumed)
  - reachable?

“SMART” “CONTRACTS”

EVEN MORE PROBLEMS TO CONSIDER

Smart contracts on multiple nodes:

- how do you keep them updated?

Lots of complex computer problems lie here:

- atomicity
- race conditions
- secure multi party computation (not exactly solved)

Smart contracts on single nodes:

- SPOF, central authority?

“SMART” “CONTRACTS”

JUST REMEMBER THIS

Not saying “don’t”.

Saying: think about “why” and “how”.

A note about trust

TRUST

Trust is not a commodity.
In fact, the original bitcoin whitepaper talks about “trustless”[1].

Trust is a human value.
It's fickle and dependent on a multitude of factors (context, time, goals, people) of which technical solutions are only a part.

1. It’s not about zero trust, it’s about choosing who you trust.

FIN

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos

Backup slides

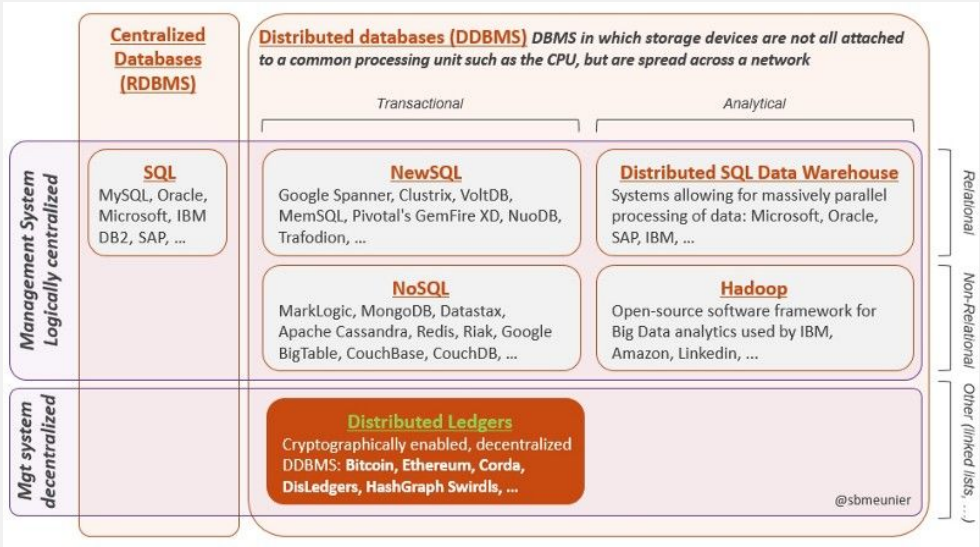
SOME HELPFUL CONCEPTS

“ISN’T IT JUST A DATABASE?”

One can follow the following conceptual path:

1. Centralised database
2. Replicated database
3. Distributed database
   (Up to now, all under the control of one entity.)

4. Distributed ledger / blockchains
   (Under the control of multiple entities)



https://medium.com/@sbmeunier/blockchain-technology-a-very-special-kind-of-distributed-database-e63d00781118

SOME HELPFUL CONCEPTS

HORIZONTAL & VERTICAL

Horizontal component

- Data is distributed horizontally: every node in the network has a full* copy of the data

Vertical component

- Data is ordered chronologically, in a series of blocks, each connected (and cryptographically linked) to the previous

* Some argue that only sending *some* of the data to *some* of the nodes is a superior model