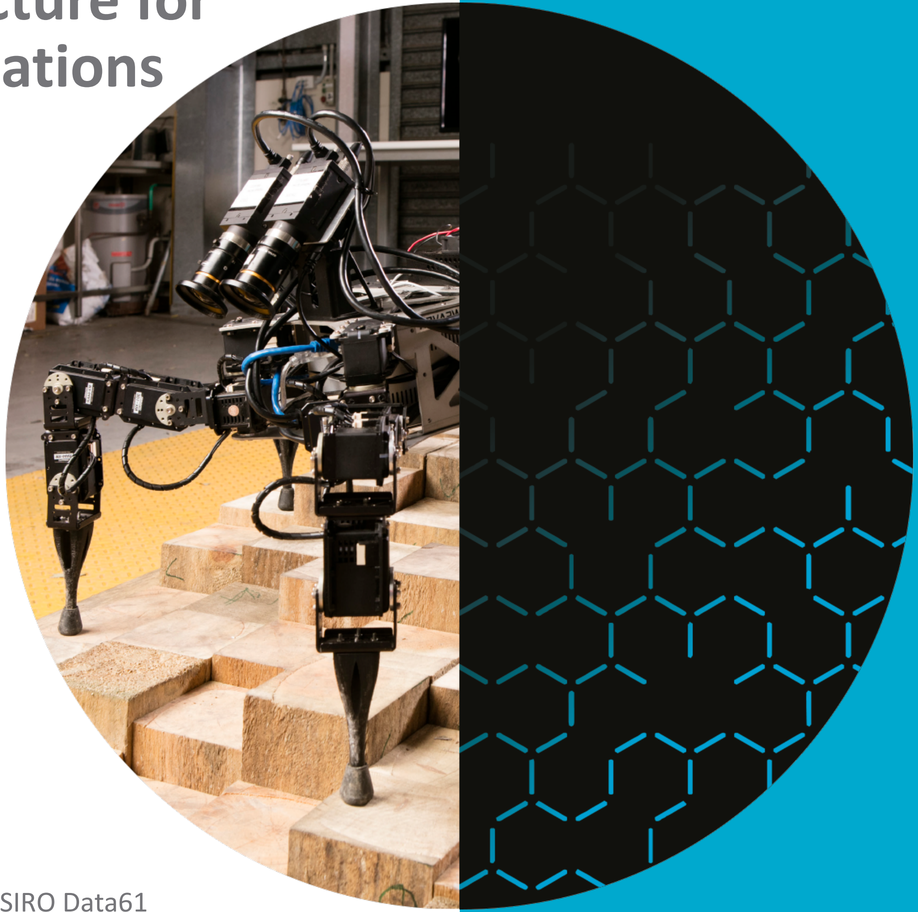
COMP6452
Software Architecture for Blockchain Applications



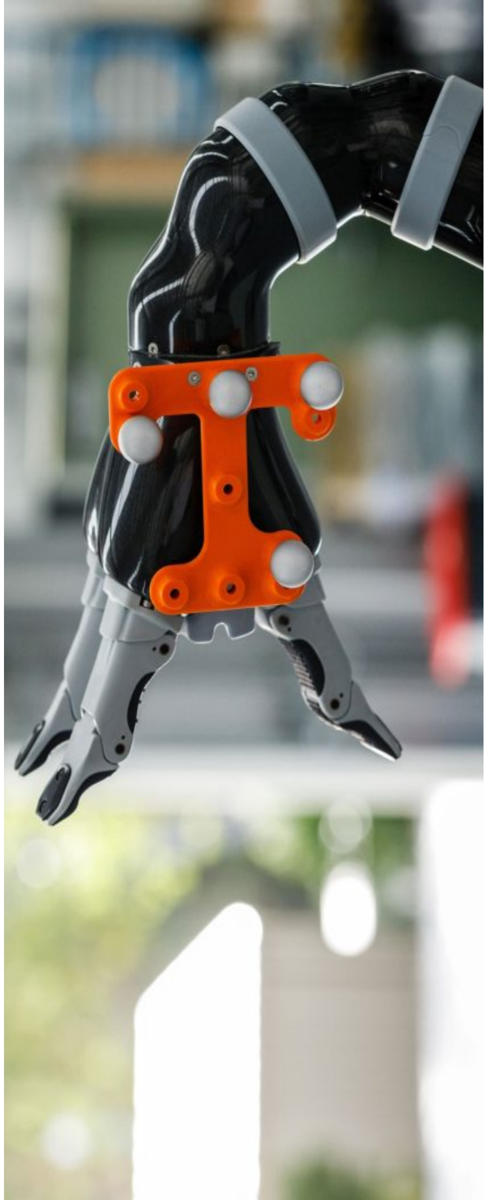
Data Management, Governance and Privacy

Helen Paik
| Senior Lecturer @ CSE, UNSW
| Visiting Researcher @ AAP team, CSIRO Data61
h.paik@unsw.edu.au

Australia’s National Science Agency

Outline

- Blockchain Data Management
- Blockchain Data Privacy and Data Quality
- Blockchain Governance



Background and motivations

Blockchain as a data store:

new challenges for the software developers

- No standard data model and query support
  - We don’t *yet* have a “standard” data model we can use to interact with blockchains
  - We view the data in blockchains as either key-value pairs, or documents (i.e., loosely structured collections of key-value pairs).
    - Usually, data items also reference off-chain data stores, which may or may not use traditional database.
  - Querying (i.e., reasoning about the data) or simple search/retrieving data in less structured and loosely connected data sources usually equates to ad-hoc, manually crafted programming
  - Considering the increasing demand for data analytics at scale on blockchain data, this lack of data model and query level abstraction is a challenge

Blockchain as a data store:

new challenges for the software developers

- Significant implications in different architectural choices
  - In terms of catering for the desired level of service from blockchain-based applications, there are “new” considerations as a software designer
    - Many contemporary implementations show low throughput and high latency.
    - There are fees charged to storing and manipulating data
    - The high trust level can be off-set by differing consensus mechanisms
    - Different configuration choices of public and private blockchains
- Understanding the choices available and its implication on the data is going to be a crucial skill
- This course, in many ways, try to address this particular challenge.

Blockchain as a data store:
new challenges for the software developers

- Issues with data privacy and quality
  - The data stored in blockchains are permanent and transparent to the “whole” network, leading to potential threats to privacy
    - Although encryption is possible, still open to brut-force decryption attack in the future (e.g., the suggestion of quantum computing rendering the current encryption system moot isn’t too far fetched)
    - Techniques analyzing patterns to link data to reveal an identity (i.e., privacy leakages) already exist
  - Open/public blockchain systems are vulnerable to garbage data being created in the system. The fact blockchains as a data store is permanent and immutable makes the data quality management issue even more important.
  - These issues also highlight the need for blockchain governance

This week's topics

- This week's topic isn't introducing new concepts. Rather, we try to understand the same concepts from a different viewpoint
- Blockchain data management
    - We will examine blockchains as a data store (cf. RDBMS) to gain different perspectives and levels of abstraction of the blockchain architecture
- Issues in Blockchain data
    - Data privacy
    - Data quality management
- Governance of Blockchains

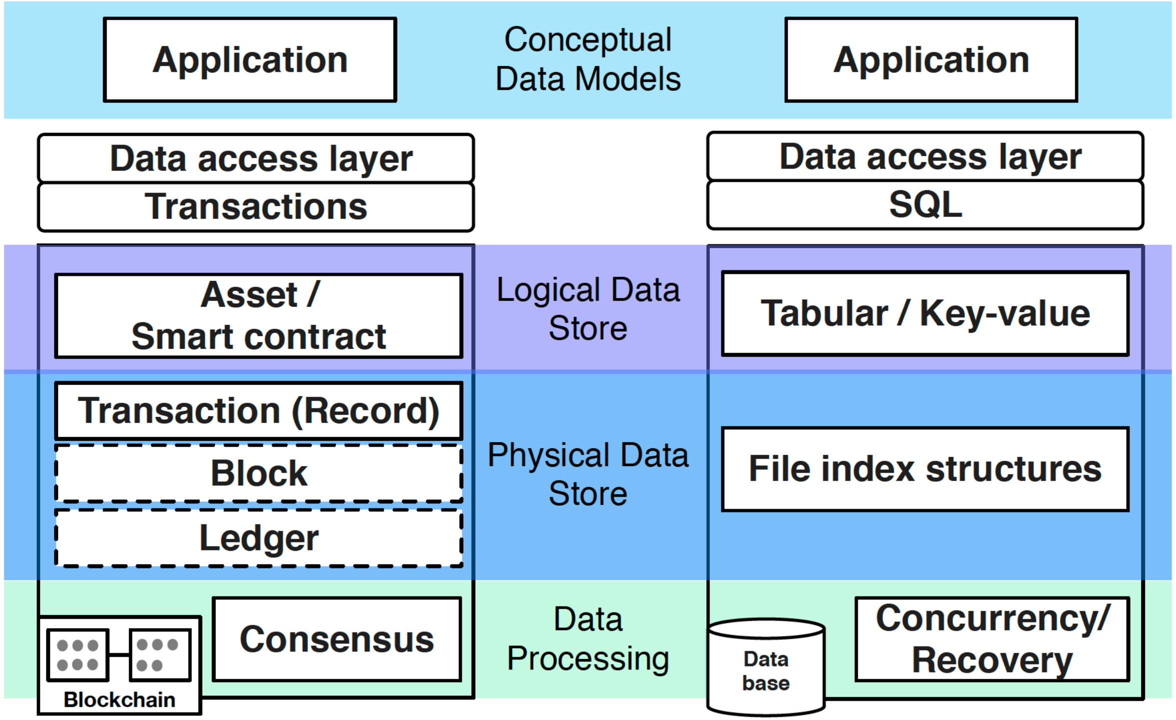
Blockchain Data Management

Examine blockchains as a data store

- To be able to relate how many software developers would see and use blockchains in their system, instead of looking at the blockchain as the first-class citizen, we are going to examine blockchains as one component of a system that allows the developers to store data

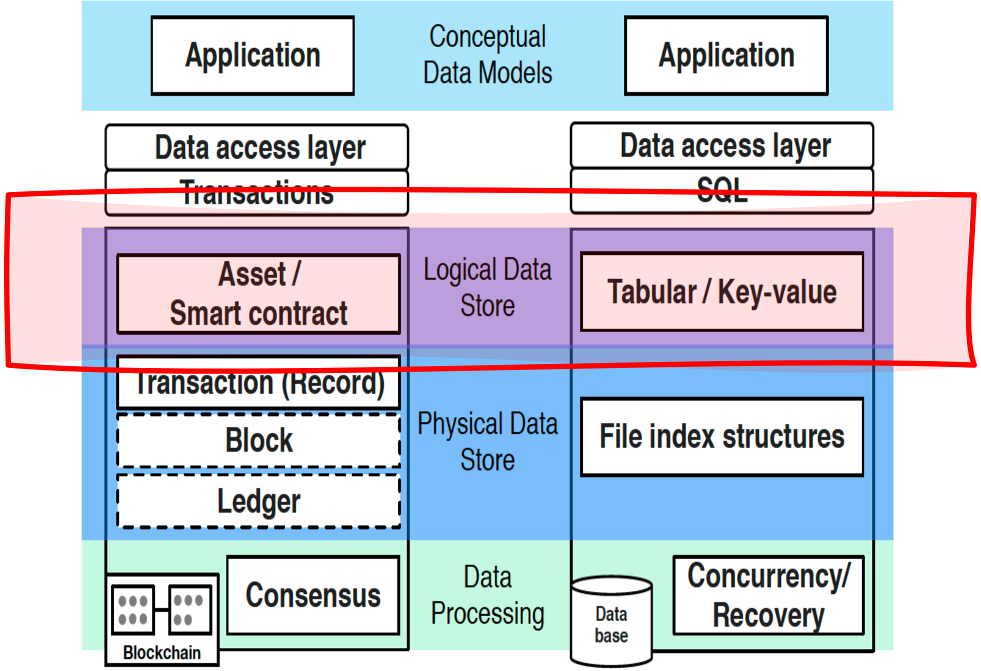
- We do this by looking the blockchain architecture through the eyes of a conventional data base system (in particular the most common database systems like relational database management system)

Blockchain architecture as a data store



Logical Data Layer

- In RDBMS, this will represent "relational tables" – the logical representation of the data that developers interact with
- what is "visible" to the developer as means to model and store data with blockchains?
  - Two blockchain constructs: Assets, Smart contracts



Primary data model for developers

- **Assets**:
  - cryptocurrency or digitalised traditional assets … the main use is to track pieces of information beyond ownership (e.g., attributes of a physical object)
  - Two types: UTXO (e.g., Bitcoin, R3 Corda) and Account-based model (e.g., Ethereum, Hyperledger)
    - – UTXO model enables parallel transactions and better privacy as they are stateless, but they can be fragmented which increases programming complexity
    - – Account-based model represents all assets per account which provides an integrated view of an account and reduces programming complexity. But since balances of all accounts are traced the global state, this model limits concurrent transactions and privacy.
- **Smart contracts:**
  - facilitate storing and manipulating data. Similar concepts exists in database (e.g., stored procedures), but smart contracts ensure that data within the smart contract can only be manipulated by calling the approved functions.
  - considered as “data with rules”.

Logical Data Layer – some observations

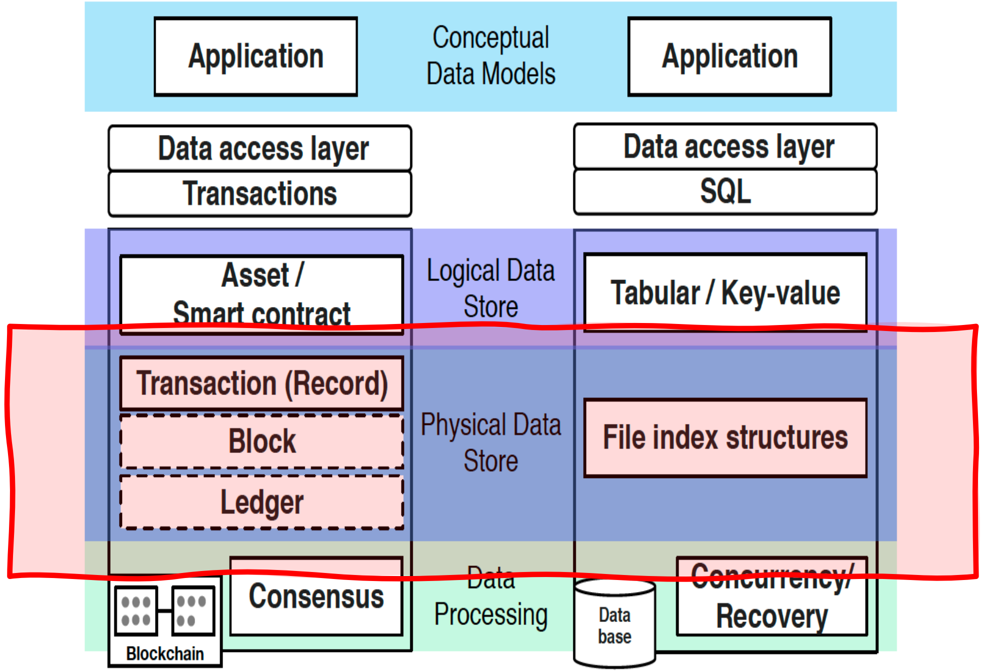
An account (aka., address) is a unique reference (i.e., key) to an asset or a smart contract. Depending on the blockchain platform, the asset or smart contract can be represented by a simple data structure, object, a JSON/XML

Thus, we can state that logical data layer of blockchains contains a schema-less data model based on key-value or document store.

Although it is possible to store more complex and schematic data, it is important that smart contracts are not over-engineered such that their cost-efficiency and security are lost.

Physical Data Layer

- In RDBMS, this represents "storage and index structures" -- effectively support querying and retrieving data
- what is "equivalent" in blockchains? How does a developer see the data is stored in blockchains?
  - Two blockchain constructs: blocks (with transactions entries) and ledger



Physical Data Layer: Data Storage Structure

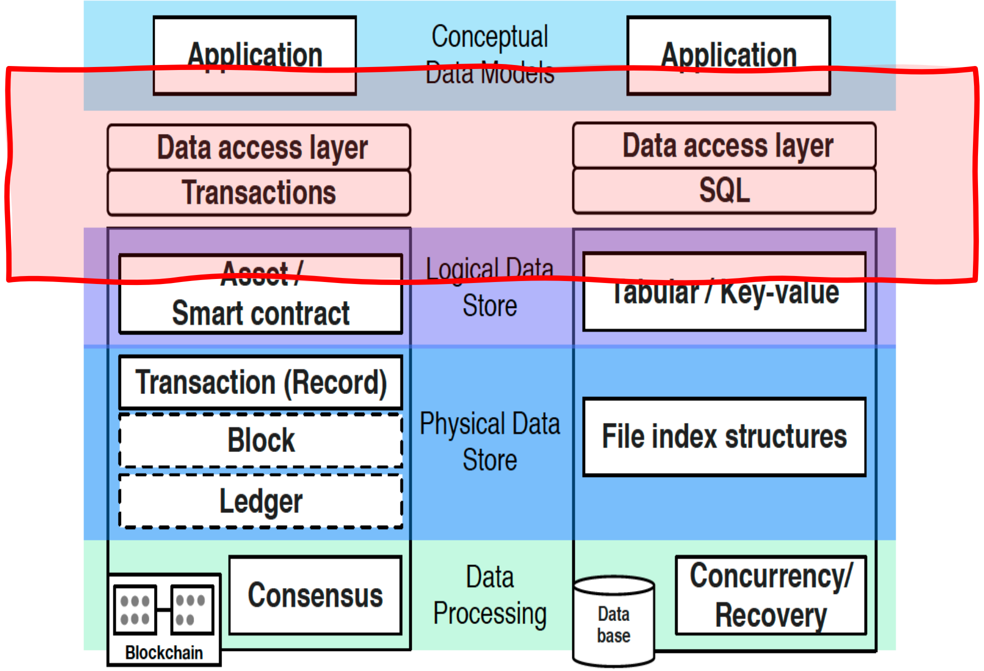
- **Transaction entries (in a Block)**
  - Holds the “data operation” results on the two data models (i.e., assets or contracts)
  - typical results stored are new accounts, new smart contracts or, updated assets
- **Blocks**
  - The exact content of a block depends on the platforms
    - Merkle tree Bitcoin, Trie Ethereum
    - In account-based models, the global states are tracked separately (e.g., CouchDB in HyperLedger)
- **Ledger**
  - a single global list chained blocks
    - (DAG, alternative ”chain” …)

Physical Data Layer: some observations

- Similarly to RDBMS, the physical storage of data in blockchains is decided by the platform implementation.
- More important distinction would be that the physical layer of blockchains is optimised for storage of transactions, rather than querying/searching or indexing transactions (e.g., for analysis)
- The high level of replications in blockschains increases availability and consistency, but unlike traditional distributed database systems, replications do not improve throughputs or latency.
  - the consensus process still controls the block creation … (i.e., bottleneck)
  - Recent techniques like sharding is proposed to improve the performance

Data Access Layer

- In RDBMS, this layer will represent APIs and SQL to insert/read/query data. The API to CRUD (Create, Read, Update, and Delete) operations are well established.
- What is the equivalent
  - in terms of a query language like SQL
  - In terms of CRUD operations?



Data Access Layer: CRUD centric view

- CREATE() (≈ INSERT a tuple in RDBMS) – create a transaction record. Again, here this either creates an asset, or smart contract
- UPDATE() (≈ UPDATE a tuple in RDBMS) – no updating … but one may argue that a transaction can change (transfer) the ownership of a title or debit cryptocurrency from one account and credit to another.
- DELETE() (≈ DELETE a tuple in RDBMS) – no delete ... But a transaction could be used to set an asset to a null value or change a state to an unusable state.

Data Access Layer: CRUD centric view

- READ() (≈ SELECT-ing tuple in RDBMS (??) – Compared to databases, reading blockchain data is not straightforward.

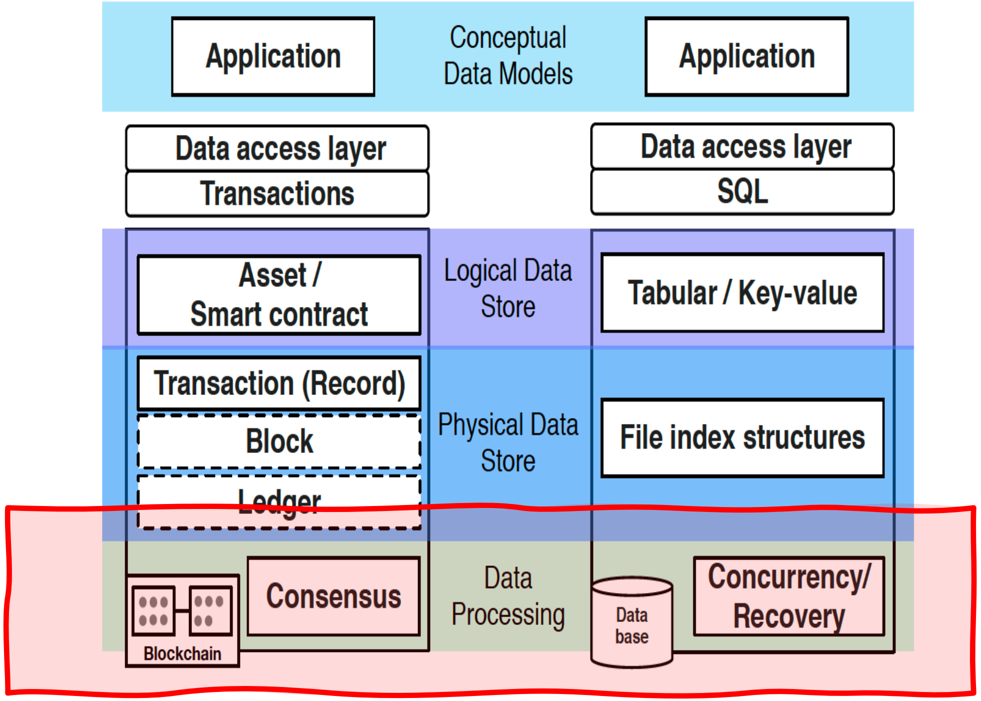
  - A blockchain transactions do not directly return results or indicate whether the transaction is executed.
  - A typical way to read data (your assets/smart contracts) is through unique identifiers (IDs) and a blockchain explorer/client
  - The tool (blockchain client) sequentially goes through the ledger, starting from the most recent block, looking for the given ID (be it asset/account, transaction, or smart contract). An explicit read is required to check if a transaction has been accepted/rejected or confirmed.

Data Access Layer: some observations

- On-going effort in blockchains to support more efficient data access
- Some of the examples/efforts in this direction:
  - Etherscan - copy the blockchain data to a centralised indexing server for faster access. Hyperledger Fabric maintains a purpose-built index
  - Ethereum Query Language (EQL) is an SQL-like query language which aims to provide a general-purpose query/answer implementation for blockchain data. It allows queries to quickly extract information scattered through several records in the blockchain using collections of blocks, types of objects (e.g., transactions and accounts) and a binary search tree
  - R3 Corda's ledger data are maintained in a relational database to enable both read and write queries using SQL. BigchainDB is an alternative design where a NoSQL query language is used to both read and write blockchain data.

Data Processing Layer

- in RDBMS, the primary goal is to provide ACID properties for all data operations
- What is the equivalent concept in blockchains?
  - How does the consensus protocol affect the discussion?
  - Can blockchains also provide the ACID (Atomicity, Consistency, Isolation, and Durability)



Data Processing Layer

- **Atomicity** (A in ACID)
  - Definition: a data operation (i.e., database transaction) commits or fails entirety (i.e., all-or-nothing)
- In blockchains:
  - Yes. It can comply
  - Each node independently executes a transaction. Only the successful transactions are included in a block. If it fails midway (e.g., due to insufficient assets or smart contract exceeding the set fee), all asset changes are rolled back, and the transaction is rejected.

Data Processing Layer

- **Consistency (C in ACID)**
  - Definition: Committed transactions are visible to all future transactions (i.e., "consistent state" database-wide)
- In blockchains:
  - No … but depends on consensus protocol
  - Consistency could be temporally violated in Nakamoto-consensus-based blockchains. Once the longest chain emerges, only the transaction in that chain will be confirmed. A confirmed transaction is valid and visible to all future transactions.

Data Processing Layer

- **Isolation (I in ACID)**
  - Definition: Uncommitted transactions are isolated from each other. In DB, this is achieved by sequentially ordering possibly conflicting operations
- In blockchains:
  - Yes. It does comply
  - Miners execute transactions sequentially. Also, a minor can see only the assets of blocks generated by the parent block and its predecessors (i.e., blocks are added sequentially).
  - Transactions in the temporary chains are isolated from each other (not affecting transactions in other chains).

Data Processing Layer

- **Durability (D in ACID)**
  - Definition: Once a transaction is committed, it is permanent (i.e., the ‘true’ state system-wide even after a system failure).
- In blockchains:
  - No … depends on the concensus protocol
  - In Nakamoto-based protocol, a transaction once included in a block can be later rejected. So the definition of “transactions” can carry more subtleties in discussing conventional sense of database transactions (i.e., the transactions are “likely/probabilistically” committed).
  - Confirmed transactions included in the main chain are permanent even after a system failure

Data Processing Layer: some observations

- The ACID properties in a database system are not strictly guaranteed in blockchains as the data processing layer in blockchains aims to guarantee a different set of properties (transparency, immutability and so on)
- Applications that utilise multiple blockchains start to emerge.
  - Transactions across multiple blockchains can be managed using a centralised manager (e.g., a central exchange that ensures the cryptocurrency exchanged from one system to another are confirmed by both, or rollback)
  - Truly decentralised approach might be more desirable. There are a few inter-blockchain protocols being proposed (e.g., utilising DAG)

Blockchain Data Administration

- In RDBMS, database administration assures that the database is maintained so that the desired level of performance and security is achieved.

- In blockchains, it is still a new domain and the best practices are not yet identified. However, notwithstanding a unique set of configuration and management challenges of a blockchain, the administration tasks could well be introduced into the job role of a database administrator

Blockchain Data Administration - Deployment

- The task would depend on the chosen blockchain platform
  - For public blockchains – administrator needs to install a blockchain client to provide connectivity to the blockchain and a blockchain explorer. May need to provision sufficient bandwidth, storage, memory and CPU to sync up with the public blockchain
  - In private/consortium blockchains – administrator needs to either install on-premise or in the cloud (more likely as there are many so-called “Blockchain-as-a-Service” infrastructure service offerings). In all cases, the administrator needs to set “critical” configuration parameters such as interblock time, or to generate the first block.

Blockchain Data Administration - Maintenance

- The task particularly relevant if the blockchains are installed on-premise or in a cloud
  - The administrator needs to monitor the performance (e.g., latency and throughput) and resource utilisation. Using this, adjustments in mempool size, block size or hashrate may be needed.
  - In general, as the nodes are fully replicated, explicit backup is not necessary (unless the chain is pruned to remove old blocks)
  - The administrator needs to keep up with software updates as “soft forks” to fix security vulnerabilities or to utilise new features in the blockchain platform
  - Managing hard forks may be necessary if significant functional changes or data corrections are required

Blockchain Data Admin – some observations

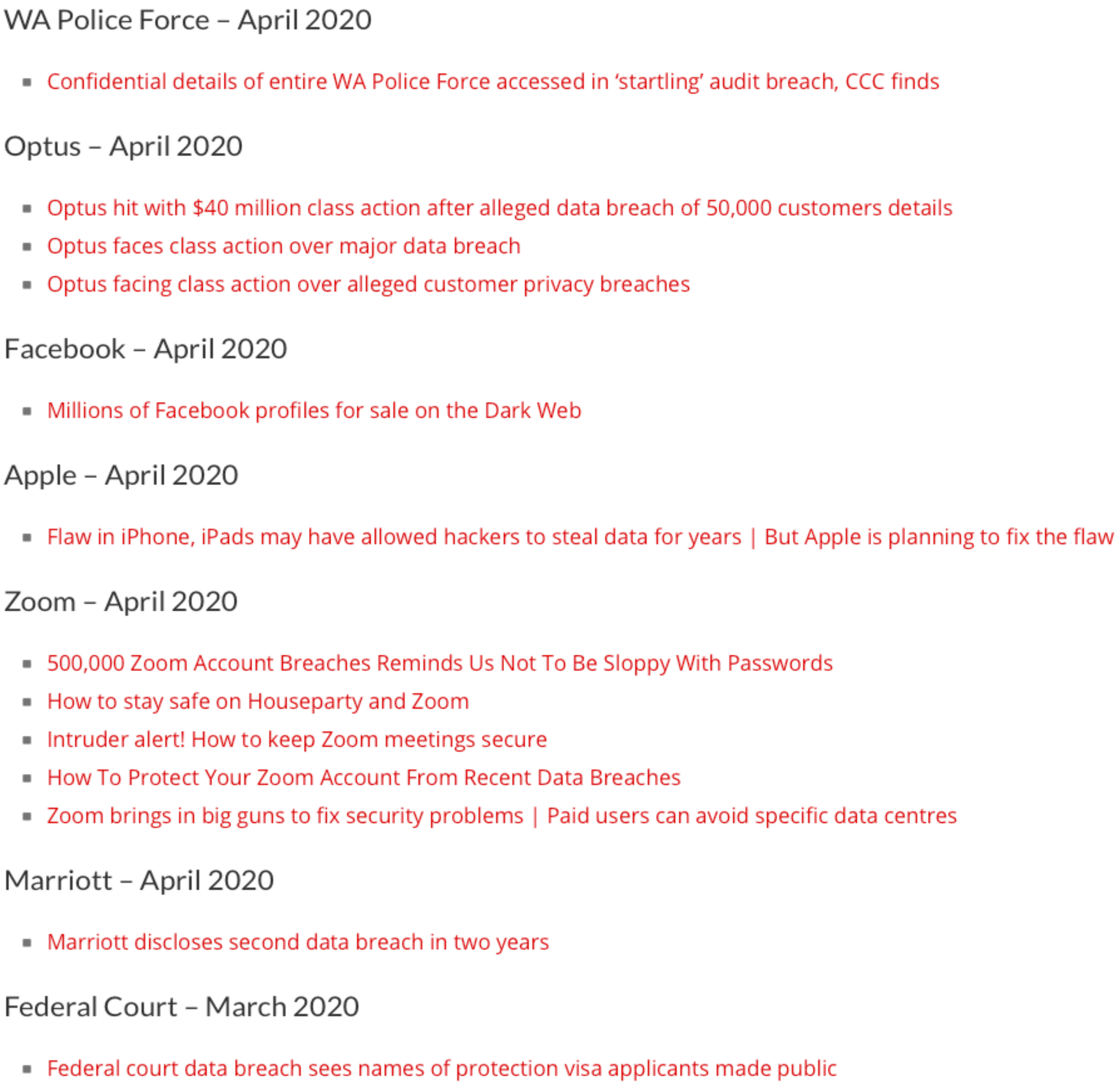
- Blockchain platforms are evolving fast. Implication for data administrators:
  - the blockchain will have to be migrated to another platform (either upgraded version of the same, or different platforms)
  - An analysis of different migration patterns have shown that migrating between blockchains is achievable [see ref. 3 for details]
  - A more important note in this migration issue (≈ interoperability issue)
    - the data design (model) should consider this issue at the start … proactive data designs such as the use of simplified data structures, use of smart contract registries are recommended

Concerns with Data in Blockchains

- Data Privacy
- Data Quality

Data Privacy

- Data breach, misuse is a constant and common problem
  - https://www.webberinsurance.com.au/data-breaches-list



WA Police Force – April 2020

- Confidential details of entire WA Police Force accessed in 'startling' audit breach, CCC finds

Optus – April 2020

- Optus hit with $40 million class action after alleged data breach of 50,000 customers details
- Optus faces class action over major data breach
- Optus facing class action over alleged customer privacy breaches

Facebook – April 2020

- Millions of Facebook profiles for sale on the Dark Web

Apple – April 2020

- Flaw in iPhone, iPads may have allowed hackers to steal data for years | But Apple is planning to fix the flaw

Zoom – April 2020

- 500,000 Zoom Account Breaches Reminds Us Not To Be Sloppy With Passwords
- How to stay safe on Houseparty and Zoom
- Intruder alert! How to keep Zoom meetings secure
- How To Protect Your Zoom Account From Recent Data Breaches
- Zoom brings in big guns to fix security problems | Paid users can avoid specific data centres

Marriott – April 2020

- Marriott discloses second data breach in two years

Federal Court – March 2020

- Federal court data breach sees names of protection visa applicants made public

Data Privacy

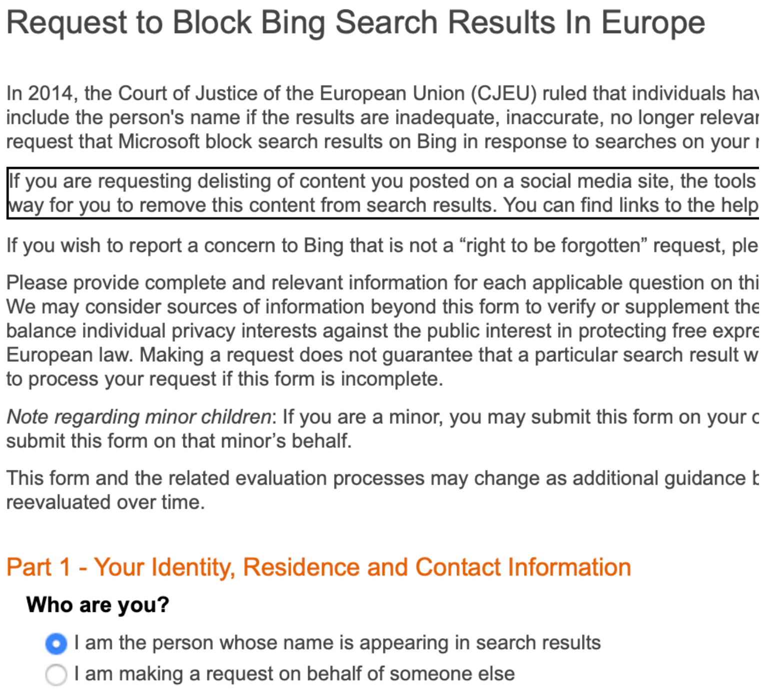
- Is preserving privacy all about stopping data breach?
  - A data breach = ***an unauthorised access*** to confidential and private, or other sensitive information
  - Is privacy = security?
  - Main privacy (data privacy) concerns:
    - What is being collected, how and why the data is shared or used …
- Many privacy concerns are being regulated
  - e.g., GDPR in EU, Australian Consumer Data Right (OAIC), …
  - Blockchain applications need to comply with regulations, but are there challenges or limitations ?

What’s required by privacy regulations?

- **Access** (and timeliness of the access) – the right to access their personal data and such access should be given without undue delay.
- **Rectification** – right to correct inaccurate data concerning him or her
- **Restriction of usage** – right to consent to use
- **Portability of the personal data** – right to receive the personal data in a machine-readable format for portability with other service
- **Right to be forgotten** – right to remove personal data concerning him/her without undue delay.

Right to be forgotten

- E.g., Bing Search



Request to Block Bing Search Results In Europe

In 2014, the Court of Justice of the European Union (CJEU) ruled that individuals hav include the person's name if the results are inadequate, inaccurate, no longer relevar request that Microsoft block search results on Bing in response to searches on your r

If you are requesting delisting of content you posted on a social media site, the tools way for you to remove this content from search results. You can find links to the help

If you wish to report a concern to Bing that is not a “right to be forgotten” request, ple

Please provide complete and relevant information for each applicable question on thi We may consider sources of information beyond this form to verify or supplement the balance individual privacy interests against the public interest in protecting free expre European law. Making a request does not guarantee that a particular search result w to process your request if this form is incomplete.

Note regarding minor children: If you are a minor, you may submit this form on your c submit this form on that minor’s behalf.

This form and the related evaluation processes may change as additional guidance t reevaluated over time.

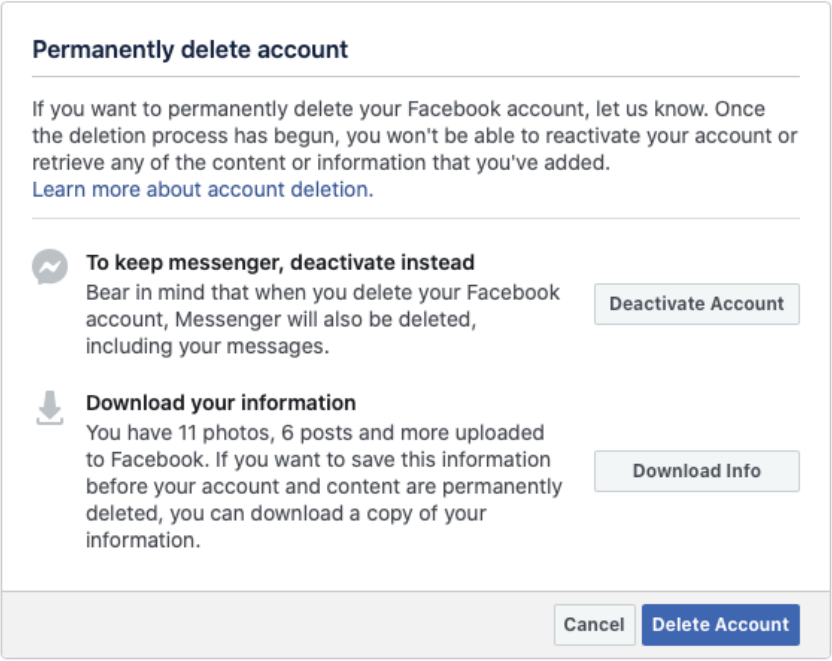
Part 1 - Your Identity, Residence and Contact Information

Who are you?

● I am the person whose name is appearing in search results

○ I am making a request on behalf of someone else

- E.g., Facebook



Permanently delete account

If you want to permanently delete your Facebook account, let us know. Once the deletion process has begun, you won't be able to reactivate your account or retrieve any of the content or information that you've added.
Learn more about account deletion.

To keep messenger, deactivate instead
Bear in mind that when you delete your Facebook account, Messenger will also be deleted, including your messages.

Download your information
You have 11 photos, 6 posts and more uploaded to Facebook. If you want to save this information before your account and content are permanently deleted, you can download a copy of your information.

Data Privacy in Blockchains

- Access (and timeliness of it)
  - … depends on the permissions
    - public blockchain, no “privileged” participant and access is equal to all
    - permissioned blockchain, possible to grant appropriate access rights
  - How to present the records to the user?
- Rectification
  - depending on the ownership of the transaction record, another transaction can be issued to rectify the error

Data Privacy in Blockchains

- Restriction of usage:
  - a user can consent to a pre-encoded policies in smart contracts
  - Smart contracts can make both the consent and usage transparent and auditable
- Portability of data:
  - the data format in blockchains is machine-readable
  - but moving the data between blockchain platforms is not straightforward (e.g., moving all the history of a user account).

Data Privacy in Blockchains

- Right to be forgotten
  - blockchains are immutable by design;
  - removing data to comply with this requirement is not feasible
  - a need for discussion on how to deal with the limitation
  - A possible approach:
    - any sensitive data can be stored outside the blockchain networks
    - Maintain a ”link” between a blockchain transaction record and the off-chain data

Data Quality in Blockchains

- The value of data depends on its quality – directly impacts the quality of decision making

- The blockchain data properties do not automatically increase the quality …
  - i.e., the same "garbage-in, garbage-out" principle applies to Blockchain data

- How do you determine the quality of data in Blockchains, are there challenges or limitations ?

How do you assess data quality?

- **Consistency** – data is free from contradiction
- **Traceability** – is there audit trail of access/changes to data
- **Availability** – is the data readily accessible
- **Compliance** – is the data compliant with standards or regulations in force
- **Confidentiality** – is the data only accessible to authorised users
- **Credibility** – is the data regarded as true and believable by users

Data Quality in Blockchains

- blockchains natively provide for some quality guarantees regarding the data stored on them:
  - E.g., no tampering with data

- The blockchains could do well for managing:
- (+) consistency, traceability, and availability
  - e.g, Replicated nodes, the use of cryptographic signatures, immutable ledger …

Data Quality in Blockchains

- Blockchains may have problems with:
- (-) compliance
  - how/when should the blockchains check for compliance of data?
    - Check by the protocol? … quite limiting, only for a small/private BC
    - Smart contracts? … flexible, but smart contracts have limitations
- (-) credibility
  - how ”credible” is your external data provider? (e.g., oracles).
- (-) confidentiality
  - may depend on whether the data is on public or a network with restricted access (and how the restriction is setup) …

Improving Data Quality in Blockchains

- Managing data quality is an on-going issue
- Compliance checking
  - protocol/smart contracts not flexible enough
  - may need a hybrid of on-chain/off-chain data architecture
- Credibility
  - data sources are not subject to the underlying security mechanisms of the blockchain technology.
  - Possibly … source data from multiple oracles, or utilse reputation management system

Improving Data Quality in Blockchains

- Compliance
  - If the quality assessment Smart contracts

- Credibility
- Blockchain Oracle Configuration - Trustworthy data sources are integral parts of the blockchain networks, but it is not subject to the underlying security mechanisms of the blockchain technology.
  - Several potential solutions have been discussed such as sourcing data from multiple oracles to mitigate the risks derived from relying on a single data source and avoid a single-point-of-failure

Governance of Blockchains

Governance

- Governance refers to the structure of decision making processes that every participant agrees to follow. It includes all processes that are involved in creating, updating, and abandoning formal and informal rules of a system.
- Applying the concept of governance may seem a contradiction or irrelevant … after all, the technology aims to break the conventional governance norms (e.g., no central authority)
- In a decentralised platform where there is no "centralised ownership or accountability" is in place, who should make a decision when a problem arises, how to respond to new challenges/regulations?

Why blockchain governance ?

- Some real world examples show the need for establishing a proper governance over blockchains
  - The DAO hack (in April 2016)
    - Choice (a) do nothing (hacker keeps the funds), (b) intervene with a hard fork, modifying the Ethereum protocol to rewind/restore the state (wasn’t blockchain immutable?)
  - cryptocurrency and finance regulations
  - government-enforceable privacy regulations

Governance Model: Governance by the infrastructure

- The governance rules are encoded and embedded in a blockchain system
  - referred to as "on-chain governance" because the governance rules have been encoded directly into the blockchain itself
  - The rules are considered immutable and self-executable since the normal operation of the blockchain network will guarantee their execution in a secure and decentralized manner
  - In this model, one can specify procedures to amend themselves. Tezos (https://tezos.com), for example, promises to build a self-amending blockchain and give participants the ability to change the protocol rules, including rules to change the rules

Governance Model: Governance by the infrastructure

- (+) on-chain governance is predictable and fair in its execution. Because changing the process or the result of on-chain governance is extremely difficult, the entire system is fully transparent and auditable
- (-) on-chain governance may handle new and unexpected situations inadequately. A very narrow and precise view of decision making.

Governnace Model: Governance of the Infrastrucure

- The rules operated by stakeholders or community outside of the platform
  - A governance model that is akin to open source software communities
  - The rules include the procedure and structure of decision committees on any changes to the protocol, including the decision to fork
  - The rules are not automatically executed at the technical level … a third-party authority might therefore be required for enforcement or oversight.
  - E.g., BIPs (*Bitcoin improvement proposals), EIPs (Ethereum improvement proposals)*
    - – *Both are still informal in both their procedures and structure*

Blockchain governance examples

- Dock Network - a decentralized data exchange protocol, with a focus on sharing professional information such as work experience, connections, and reviews
  - https://github.com/docknetwork/voting/blob/master/DGPs/DGP-1.md .
- Stellar.org (Stella Development Foundation) –
  - https://www-stg.stellar.org/about/governance/
  - https://www.stellar.org/foundation/mandate#direct-development
  - https://www.stellar.org/community-fund/info

Bitcoin and Ethereum Forks

- https://www.visualcapitalist.com/major-bitcoin-forks-subway-map/
- https://www.forks.net/list/Ethereum/

References

- Analysis of Data Management in Blockchain-based Systems: From Architecture to Governance, Paik, Xu, Bandara, Lee and Lo, IEEE Access, 2019
- Governance of blockchain systems: Governance of and by Distributed Infrastructure, Filippi, Mcmullen, Blockchain Research Institute and COALA. 2018
- Patterns for Blockchain Migration, Bandara, Xu and Weber, https://dblp.uni-trier.de/db/journals/corr/corr1906.html#abs-1906-00239

Thank You

Helen Paik

| Senior Lecturer @ CSE, UNSW

| Visiting Researcher @ AAP team, CSIRO Data61

|h.paik@unsw.edu.au

Australia’s National Science Agency