- Block
  - Contains a link to the previous block
  - Contains a summary-like link to a number of transactions (transaction data)
  - Contains a blockheader which contains the metadata of the block
    - Merkle-Root
    - Previous block hash
    - Timestamp
- Chain
  - Term blockheader is introduced without further explanation
  - Term hash is introduced without further explanation
- Hash [This section sounds a bit confusing to me.]
  - A hash is a function turning any input into an output (hash) of fixed length
  - A hash-function has several characteristics
    - One-way function: It only works in one direction, input to output (hash). There is no way to generate an input from an output (hash)
    - Deterministic: The same input always results in the same output (hash)
    - Quick to compute
    - Infeasible to find two different inputs that lead to the same output (hash)
    - A minor change to the input changes the output (hash) unpredictably
- Public blockchain technology
  - A public blockchain does not necessarily have to be anonymous, but most invocations are pseudonymous.
- Enterprise blockchain technology
  - [The section is a bit difficult to deconstruct, I suggest to cut it into shorter sentences]
  - Blockchain use cases for enterprises and consortia require different features than public blockchains.
    - Identity: In many industries it is necessary to know the identity of the participants in the blockchain network. Laws and policies like Know-Your-Customer and Anti-Money-Laundering require for participants to be identified.
    - Transaction performance: High transaction throughput and low latency of transaction confirmation are important for most industries.
    - Privacy: Even though identities of network participants are shared, participating parties may require or prefer privacy about the contents of single transactions.
  - [I would put the second half of the section into a different section about consensus algorithms]

- Permissioned blockchain [To use this term as a synonym to blockchain for enterprises might be misleading]
  - In a permissioned blockchain, participants must receive permission to enter the blockchain.
- Permissionless blockchain
  - In a permissionless blockchain everyone can join the blockchain.
- Transaction proposal [Might be difficult to understand for non-technical end-users]
  - A transaction proposal is a proposed transaction send to the blockchain by an end user. At the point of proposing, it is not yet decided if the proposal gets accepted by the network community and becomes a valid transaction in the blockchain.
  - Illustration: Alice works at a car dealer and sells a car (Car1). On her work computer's application interface she creates a new transaction and proposes to transfer ownership of Car1 to the buyer.
- Transaction [I suggest using simpler terms]
  - A transaction is the changing of the state of an object in the blockchain.
  - Illustration: Alice works at a car dealer and sells a car (Car1). On her work computer's application interface she creates a new transaction and proposes to transfer ownership of Car1 to the buyer. The proposal is accepted by the network and becomes a transaction in the blockchain.
- State [I would differentiate between global/world state, and state of an asset/object]
  - The state of an object consists of current facts about that object in the blockchain. Transactions change the state of objects.
  - Illustration: Alice works at a car dealer and sells a car (Car1). On her work computer's application interface she creates a new transaction and proposes to transfer ownership of Car1 to the buyer. The proposal is accepted by the network and becomes a transaction in the blockchain. The state of Car1 is updated to include the new owner.
  - World/Global state
    - The world/global state is the current combined state of all objects in the blockchain. In Fabric this is referred to as world state, in Sawtooth as global state.
- Immutability [Suggest a tiny rewording for better understandability and addition of]
  - Immutability of a block means that once the block is committed to the blockchain, it can not be edited. Mistakes in a block can only be undone by reversing the faulty transaction in a later block.
  - Illustration: Alice works at a car dealer and sells a car (Car1). On her work computer's application interface she creates a new transaction and proposes to transfer ownership of Car1 to the buyer. The proposal is accepted by the network and becomes a transaction in the blockchain. The state of Car1 is updated to include the new owner.

    Later Alice notices, that she accidentally transferred ownership of the wrong car (Car2). She reverses that by proposing a new transaction in which she transfers ownership of the originally bought car (Car1) to the new owner, and transfers ownership of Car2 back to the car dealer.

- No single point of failure [Suggest to simplify explanation]
    - In blockchain there should be no single point of failure, as the data and infrastructure of the blockchain is spread and repeated across multiple organisations and hardware.
    - [The explanation for Fabric introduces several terms that are not explained (orderer/peer)]
- Peer
    - Typos: 'in a blockchain', 'network consist'
- Membership Service Provider (MSP) [I think it is important to include the MSP at least for Fabric, so that users/decision makers know how the rights in the network are distributed.]
    - The MSP is responsible for granting access to individuals or companies to the blockchain. It is also responsible for granting rights about which actions the individuals/companies are allowed to take and which channels they can join.