Blockchain Development
Week: 9
Title: Smart Contracts

Dr Ian Mitchell

Middlesex University,
Dept. of Computer Science,
London

September 26, 2019

Lecture Objectives

- Blockchain
- permissionless
- permission
- Smart contracts

Smart Contracts

Key points

Definition 1 [1]

business rules and a state

Smart Contracts

Definition 1 [1]

business rules and a state

Key points

- State change

Smart Contracts

Definition 1 [1]

business rules and a state

Definition 2 [7]

A smart contract is a secure and unstoppable computer program representing a n agreement that is automatically executable and enforceable

Key points

- State change
- Enforceable

Smart Contracts

Definition 1 [1]

business rules and a state

Definition 2 [7]

A smart contract is a secure and unstoppable computer program representing a n agreement that is automatically executable and enforceable

Key points

- State change
- Enforceable
- Unstoppable
- Automatic

Smart Contracts

Definition 1 [1]

business rules and a state

Definition 2 [7]

A smart contract is a secure and unstoppable computer program representing a n agreement that is automatically executable and enforceable

Definition 3

Business Logic & State: Smart contracts are executed when pre-conditions are met and can allow automatic ledger updates.

Key points

- State change
- Enforceable
- Unstoppable
- Automatic
- Pre-conditions

Smart Contracts

Definition 1 [1]

business rules and a state

Definition 2 [7]

A smart contract is a secure and unstoppable computer program representing a n agreement that is automatically executable and enforceable

Definition 3

Business Logic & State: Smart contracts are executed when pre-conditions are met and can allow automatic ledger updates.

Key points

- State change
- Enforceable
- Unstoppable
- Automatic
- Pre-conditions
- Updates
- Disintermediation
- Code is Law (Semantically sound)
- Fault tolerant
- Secure
- Intervention?

Smart Contracts

Principles

- Code is not understood
- Code understood by humans & machine
- Combination of SC and NL
- Mark-up languages
  - Legal Knowledge Interchange Format
  - XML Schema
- SC's are inherently deterministic
- Any Node must get same result
- Issues:
  - floating points can be calculated differently in OS or Browsers
  - JS math functions can be calculated differently
- If the result is different then discordance
- No consensus and BC fails
- Non-deterministic functions are not permitted
- Programs need to be reliable and stable

Smart

Definition *adj.* [6]

1. astute, as in business; clever or bright
2. quick, witty, and often impertinent in speech
3. well-kept, neat

⋮

Smart

Misnomer

- Deterministic
- Does not think
- it is not astute
- Smart?

Smart

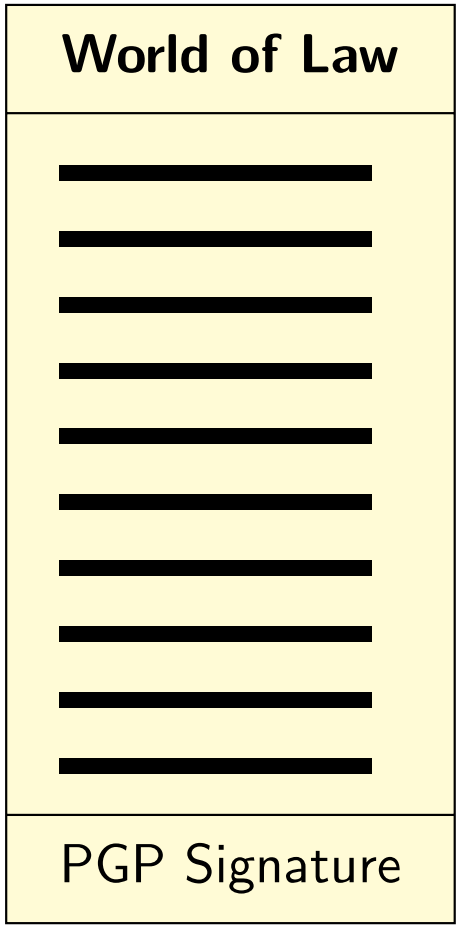
Misnomer

- Deterministic
- Does not think
- it is not astute
- Smart?
  - Disintermediation
  - Arbitration
  - Automation
  - Third parties
  - Unstoppable
  - Enforceable
  - Secure
  - Compliant
  - Legal?

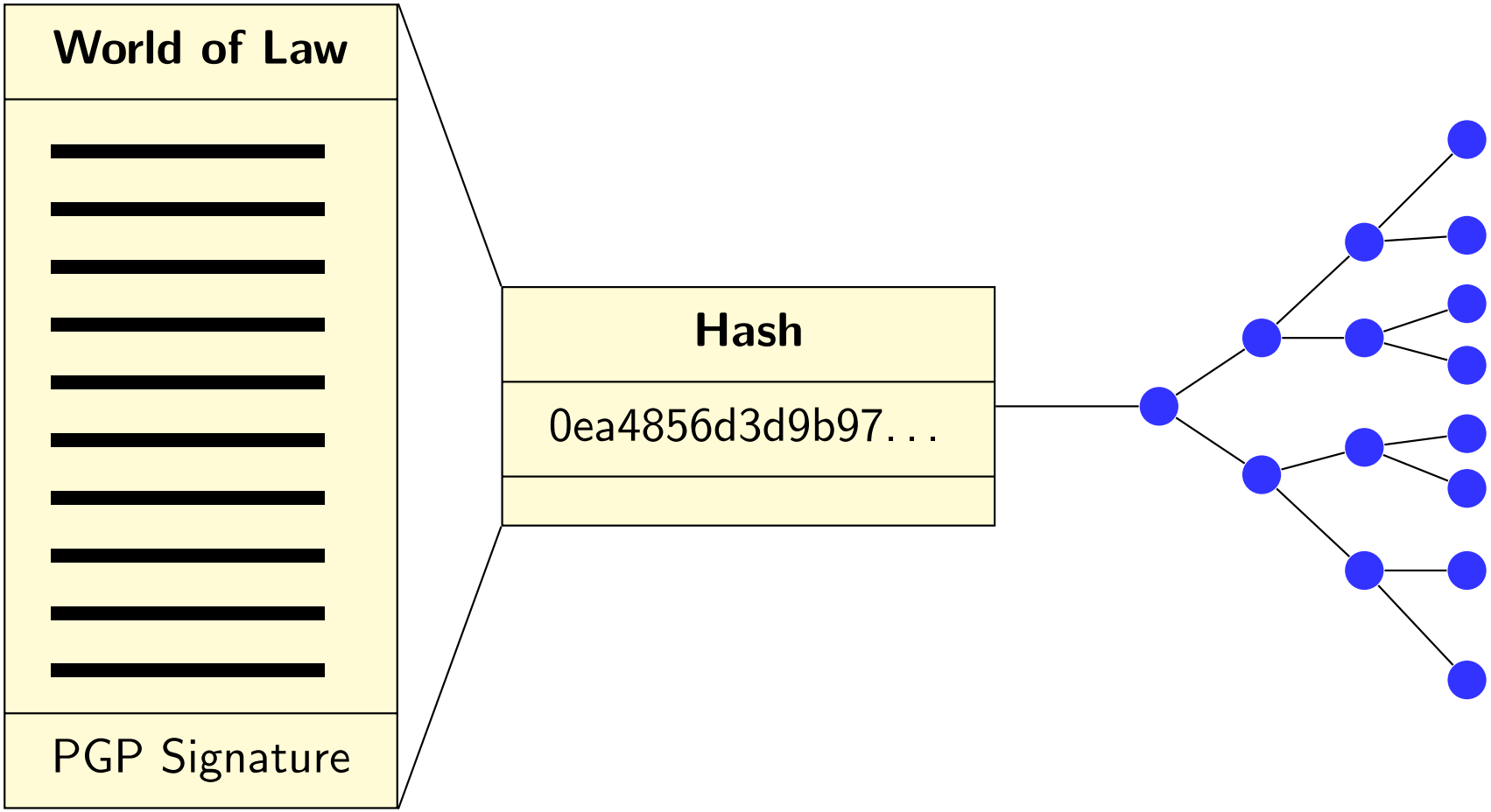
Ricardian Contract [5]

- Understood by Computer and Law
- Address challenge of issuance of value over internet
  - A contract offered by issuer to holder
  - A valuable right held by holders and managed by the issuer
  - Concise and readable
  - Signed
  - Keys
  - Unique
  - Secure

Bowtie Model[1]



Bowtie Model (adpated from [1, 5])



Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?

Smart Contracts

Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

Smart Contracts

Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

Smart Contracts

- Denotational Semantics

Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

Smart Contracts

- Denotational Semantics
  - real-world
  - meaning of the full contract
- Operational Semantics

Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

Smart Contracts

- Denotational Semantics
  - real-world
  - meaning of the full contract
- Operational Semantics
  - Execution
  - Correctness
  - Safety

Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

Smart Contracts

- Denotational Semantics
  - real-world
  - meaning of the full contract
- Operational Semantics
  - Execution
  - Correctness
  - Safety
- Smart Legal Contract

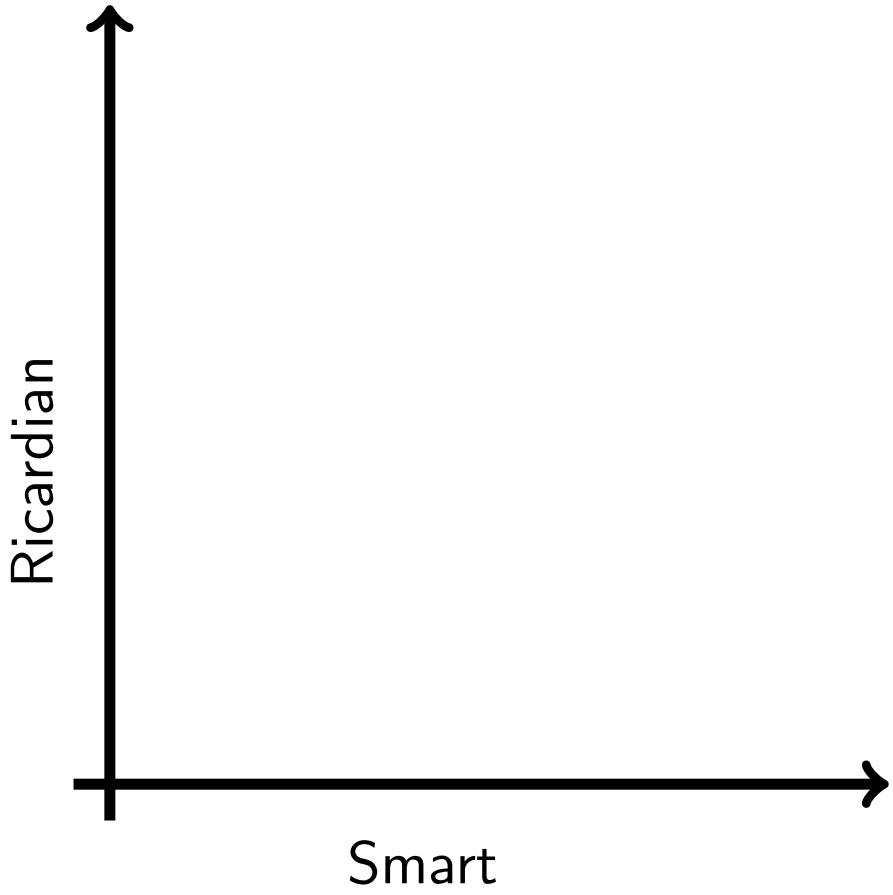
Smart Contracts?

- Law
- Accountancy
- Genesis
- Each TX includes Hash
- Is Ricardian Smart?
- Ricardian are different
  - Enforceable
  - Unstoppable
  - Automatic
  - pre-conditions
- Semantic richness
- Contract is attached

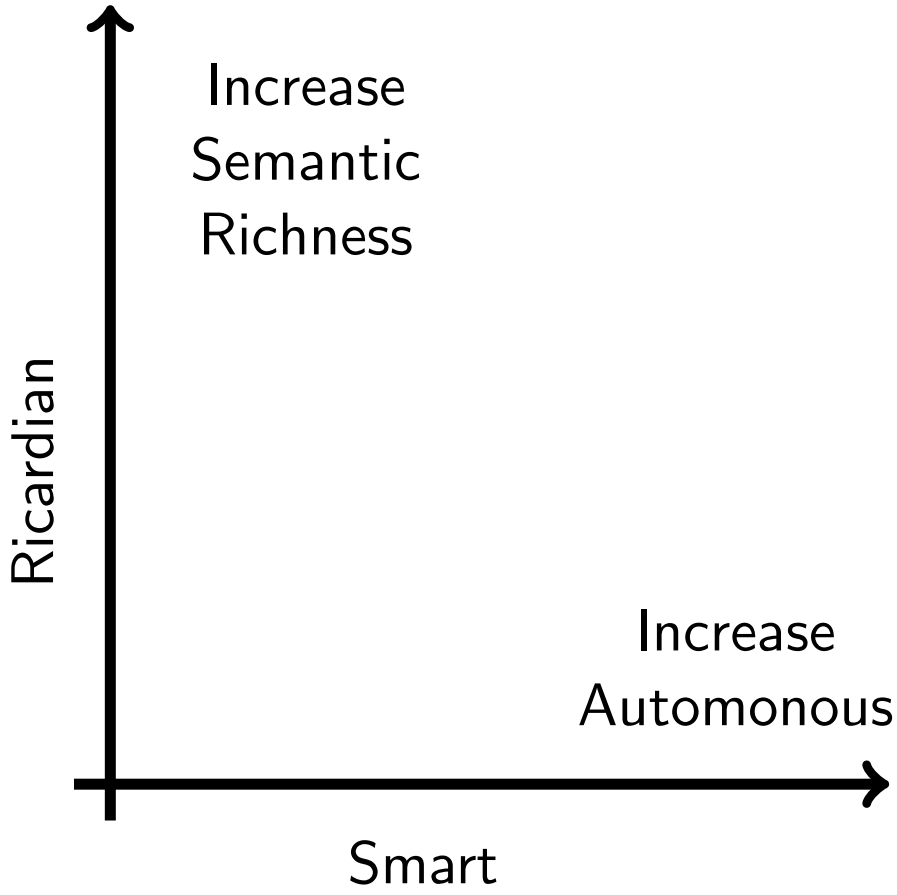
Smart Contracts

- Denotational Semantics
  - real-world
  - meaning of the full contract
- Operational Semantics
  - Execution
  - Correctness
  - Safety
- Smart Legal Contract
- Difference
  - ₿, operational
  - Ricardian, denotional

Smart Legal Contracts (adapted from [4])



Smart Legal Contracts (adapted from [4])



Smart Contract Templates [3, 2]

- Prose
- Parameter
- Code
- Common Language for Augmented Contract Knowledge (CLACK)
- Domain Specific Language (DSL)
- General-purpose Programming Languages (GPL)

- DSL needed to write Smart Contracts
- Ethereum
  - Solidity
  - Vyper
- Non-programmers
- graphical DSL
- convert semantics to code
- Flow, process
- deployed to BC
- Tibco StreamBase (Java-based)

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

Examples

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

Oracle Definition

An Oracle is an interface that delivers data from an external source to smart contracts [1]

Examples

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

Oracle Definition

An Oracle is an interface that delivers data from an external source to smart contracts [1]

Examples

- Weather reports
- Stock exchange
- IoT

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

Oracle Definition

An Oracle is an interface that delivers data from an external source to smart contracts [1]

Examples

- Weather reports
- Stock exchange
- IoT
- **Changes**
- Authentic
- Trusted
- SC registers with Oracle
- Integrity
- Back to Centralisation?

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

Oracle Definition

An Oracle is an interface that delivers data from an external source to smart contracts [1]

Examples

- Weather reports
- Stock exchange
- IoT
- **Changes**
- Authentic
- Trusted
- SC registers with Oracle
- Integrity
- Back to Centralisation?
- Decentralise Oracles
  - source data derived from multiple sources
  - consensus applied
  - authenticity

Oracles

- SC Limitations
- Access to external data
- Provide data to SC

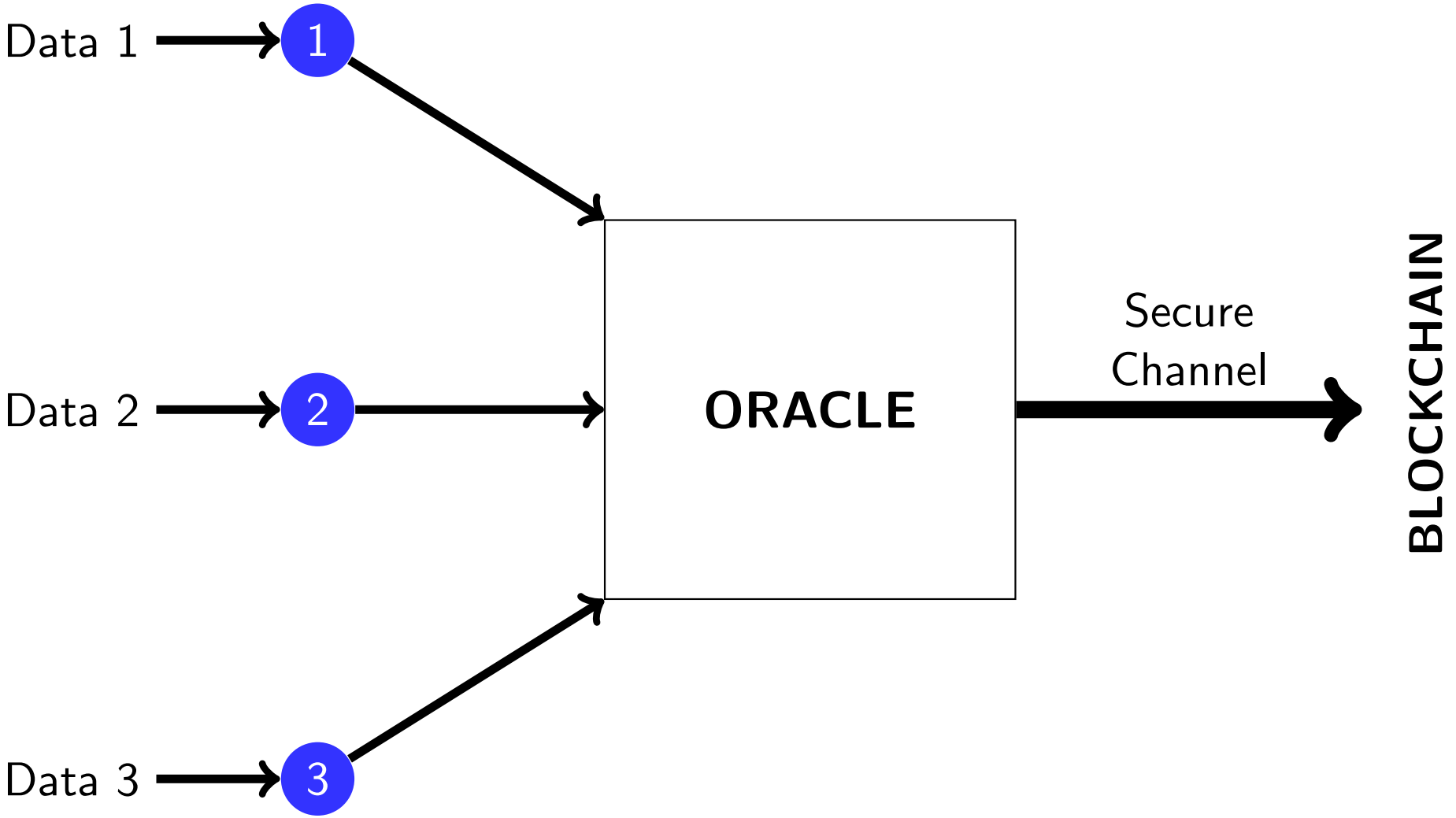
Oracle Definition

An Oracle is an interface that delivers data from an external source to smart contracts [1]

Examples

- Weather reports
- Stock exchange
- IoT
- **Changes**
- Authentic
- Trusted
- SC registers with Oracle
- Integrity
- Back to Centralisation?
- Decentralise Oracles
  - source data derived from multiple sources
  - consensus applied
  - authenticity

Oracles (adapted from [1])

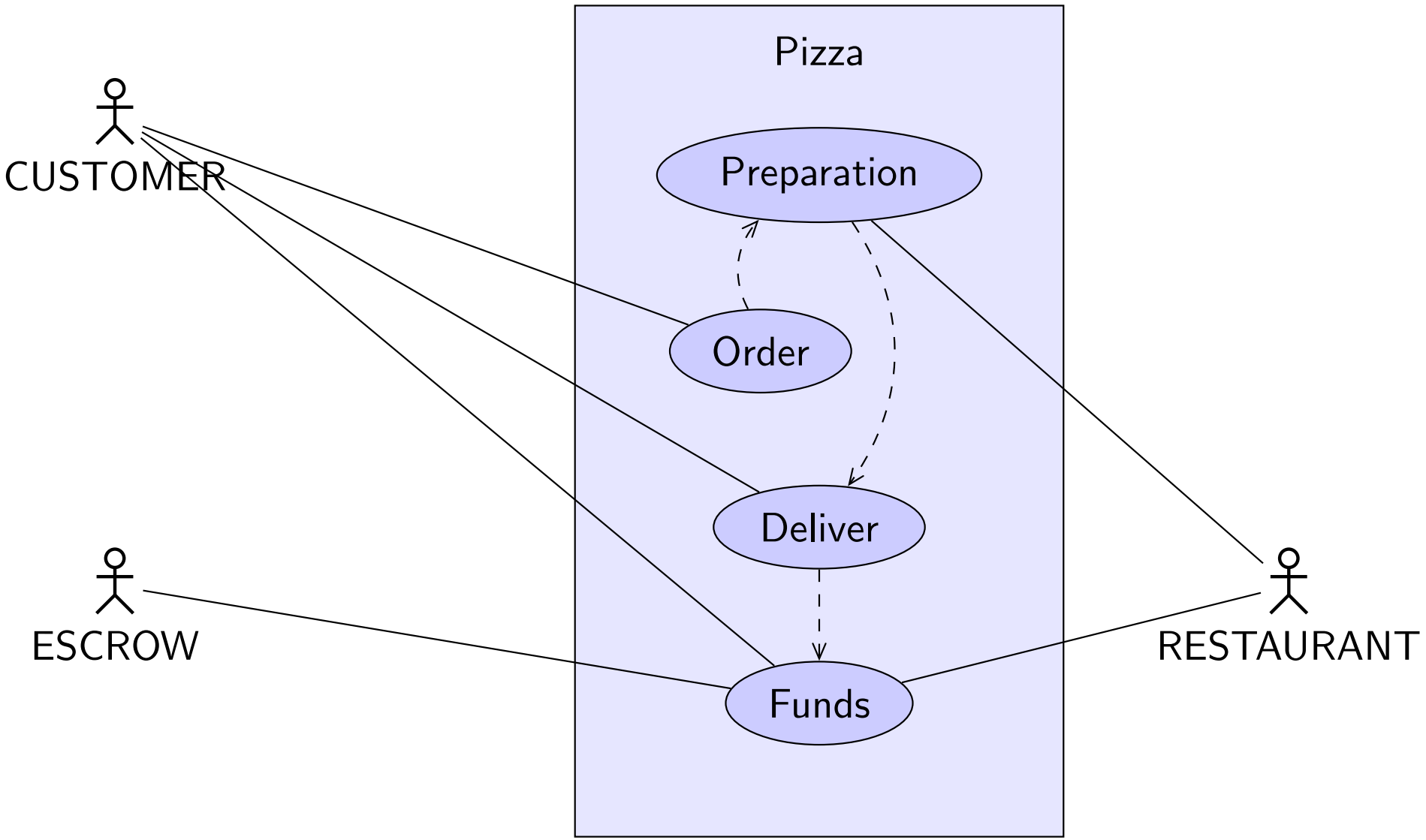


Decentralised Autonomous Organisations (DAO)

- Smart Contract does not require BC
- Smart Contract can run on BC
- Why?
- Ethereum
- Hyperledger Fabric
- Broader applications

- Trust?
- Code is Law?
- Fully Autonomous
- Interaction
- Testing?
- Validation & Verification of code

Use Case

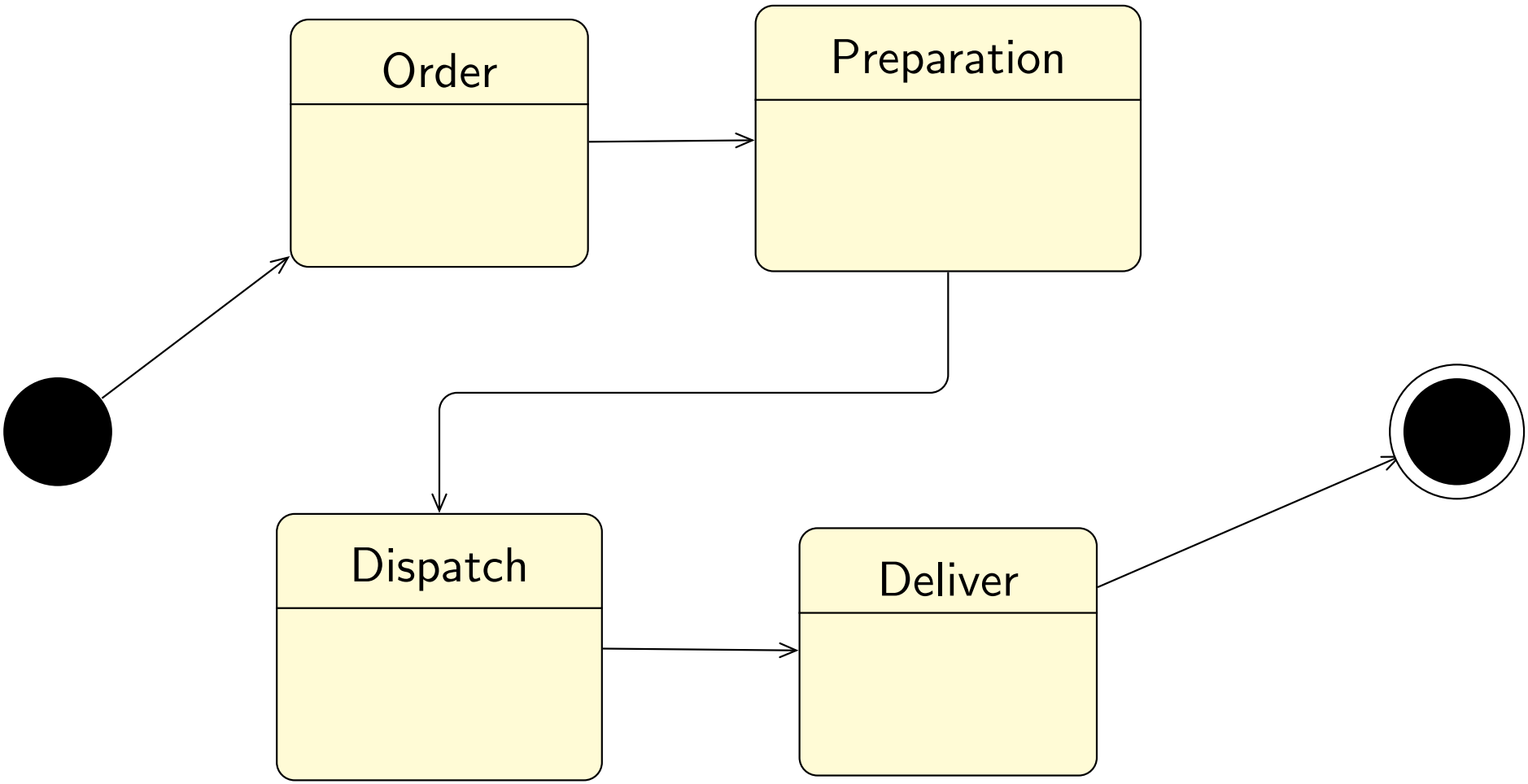

Pizza
CUSTOMER
Preparation
Order
Deliver
ESCROW
Funds
RESTAURANT

Use Case - Notes

- order-delivery latency
- customer-restaurant agreement
  - Order concurs
  - Time between order and delivery
- Money is transferred to intermediary
- Full funds are released to restaurant
  - Order concurs
  - deliver within agree time
  - Time is decided on order, can vary 30min., 60min., etc)
- Full funds are placed in escrow by customer
  - on pre-conditions reached full funds released
  - on pre-conditions breached partial funds are released to restaurant and customer
- Smart Contract

Pizza

State Transition Diagram



State/Event

Order/placeOrder

- verify order
- agreement:
  - amount
  - max preparation time
  - max delivery time
  - remuneration for failure to prepare within the max. preparation time
  - remuneration for failure to delivery within the max. delivery time
- transfer funds to escrow
- create block with this information
- block's state changes

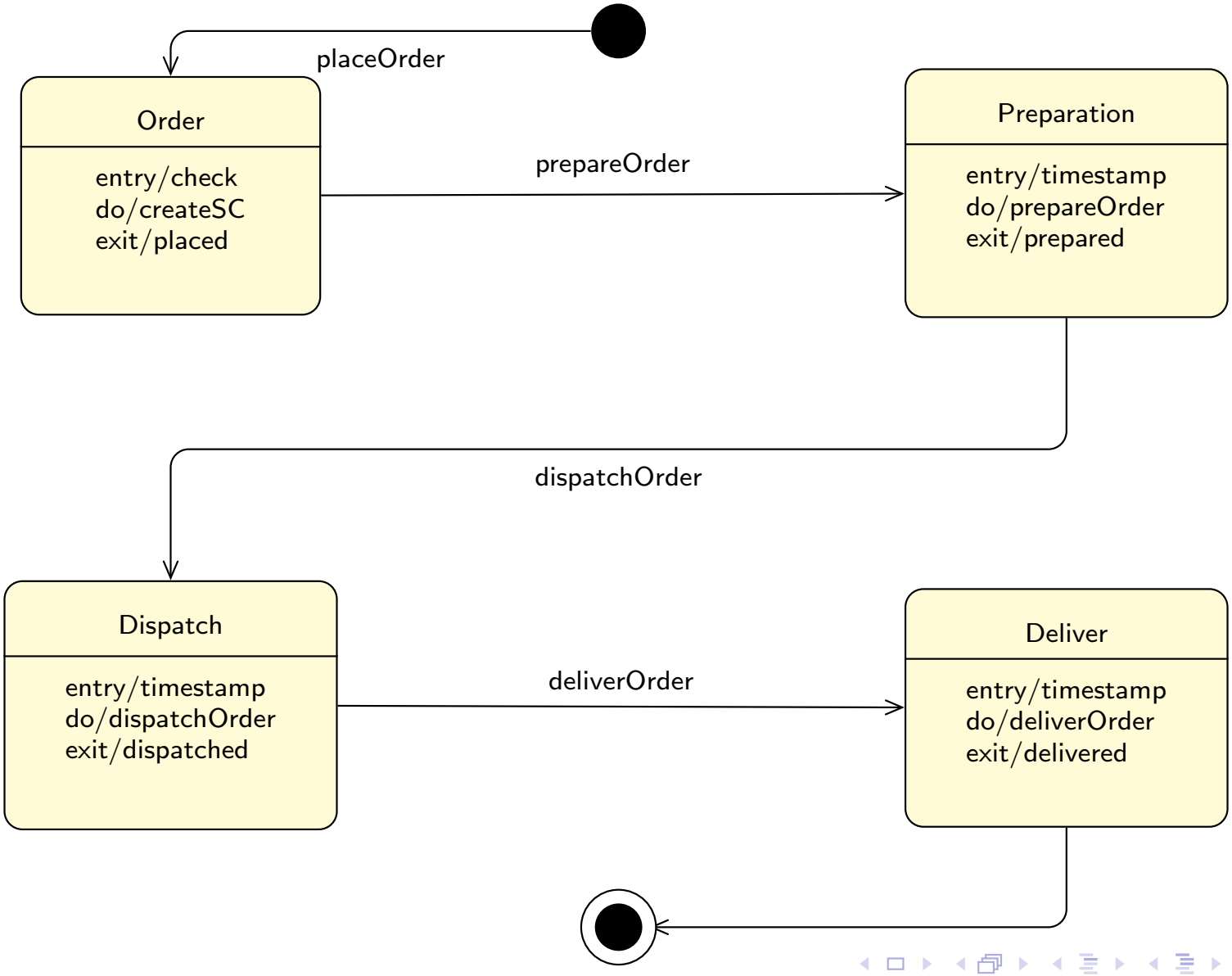
State/Event

Preparation

- ingredients
- cook
- package
- update status
- dispatch
  - prepared within agreed time
  - claim proportion of funds
  - prepared exceeding agreed time
  - remuneration for failure to prepare, funds transferred
- Events
- blockchain is updated

Pizza

State Transition Diagram



Summary

- Smart Contracts
- Ricardian Contracts
- Oracles
- DAO
- Verification and Validation of Smart Contracts

References I

[1] Imran Bashir. *Mastering Blockchain*. 2nd ed. Packt, 2018. ISBN: 9781788839044.

[2] Christopher D Clack, Vikram A Bakshi, and Lee Braine. "Smart Contract Templates: essential requirements and design options". In: *arXiv preprint arXiv:1612.04496* (2016).

[3] Christopher D Clack, Vikram A Bakshi, and Lee Braine. "Smart contract templates: foundations, design landscape and research directions". In: *arXiv preprint arXiv:1608.00771* (2016).

[4] I Grigg. *On the intersection of Ricardian and Smart Contracts*. https://iang.org/papers/intersection_ricardian_smart.html. [Accessed: Aug 2019].

References II

[5] I Grigg. "The ricardian contract". In: *Proceedings. First IEEE International Workshop on Electronic Contracting, 2004.* IEEE. 2004, pp. 25–31.

[6] P. Hanks, ed. *Collins Dictionary of English Language*. 2nd ed. Collins, 1986. ISBN: 0004331346.

[7] Nick Szabo. "Formalizing and securing relationships on public networks". In: *First Monday* 2.9 (1997).