Strafford

Presenting a live 90-minute webinar with interactive Q&A

Using Blockchain for Commercial Contracts: Permissioned vs. Permissionless, Control, Amendments, Security, Transparency

THURSDAY, JUNE 18, 2020

1pm Eastern | 12pm Central | 11am Mountain | 10am Pacific

Today's faculty features:

Melissa L. Markey, CISSP, Attorney, **Hall Render Killian Heath & Lyman**, Denver

Mark F. Radcliffe, Partner, **DLA Piper**, East Palo Alto, Calif.

The audio portion of the conference may be accessed via the telephone or by using your computer's speakers. Please refer to the instructions emailed to registrants for additional information. If you have any questions, please contact **Customer Service at 1-800-926-7926 ext. 1**.

Tips for Optimal Quality

FOR LIVE EVENT ONLY

Sound Quality

If you are listening via your computer speakers, please note that the quality of your sound will vary depending on the speed and quality of your internet connection.

If the sound quality is not satisfactory, you may listen via the phone: dial **1-877-447-0294** and enter your **Conference ID and PIN** when prompted. Otherwise, please **send us a chat** or e-mail **sound@straffordpub.com** immediately so we can address the problem.

If you dialed in and have any difficulties during the call, press *0 for assistance.

Viewing Quality

To maximize your screen, press the 'Full Screen' symbol located on the bottom right of the slides. To exit full screen, press the Esc button.

Continuing Education Credits

FOR LIVE EVENT ONLY

In order for us to process your continuing education credit, you must confirm your participation in this webinar by completing and submitting the Attendance Affirmation/Evaluation after the webinar.

A link to the Attendance Affirmation/Evaluation will be in the thank you email that you will receive immediately following the program.

For additional information about continuing education, call us at 1-800-926-7926 ext. 2.

Program Materials

FOR LIVE EVENT ONLY

If you have not printed the conference materials for this program, please complete the following steps:

- Click on the link to the PDF of the slides for today’s program, which is located to the right of the slides, just above the Q&A box.
- The PDF will open a separate tab/window. Print the slides by clicking on the printer icon.

1

Blockchain

Panel

- Melissa Markey
  - Hall, Render, Killian, Heath & Lyman, P.C., Denver
  - mmarkey@hallrender.com

- Mark Radcliffe
  - DLA Piper, Silicon Valley Office
  - Mark.radcliffe@dlapiper.com

Blockchain -- before

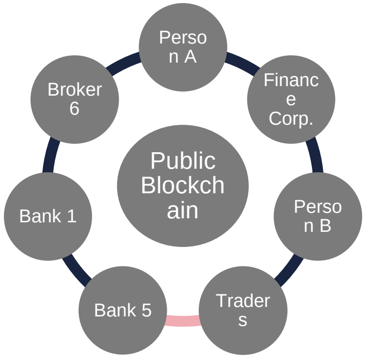
- In multi-party transactions, multiple sources of the same data
    - Financial records
    - Sales records
    - Contracts
    - Title ownership and transfer
    - Digital Assets
- Data has to be shared, reproduced, compared, and reconciled
- Third party representations and warranties of accuracy substitute for direct knowledge
- Process can be inefficient, expensive and vulnerable to fraud and mistake

Blockchain -- after

- In multi-party transactions, each relevant data element can be made available to all parties from a single source
  - Financial records
  - Sales records
  - Contracts
  - Title ownership and transfer
  - Digital Assets
- Data from blockchain entries viewable by all parties with access
- Third party representations and warranties of accuracy replaced by direct knowledge
- Process provides clear chain of provenance and is both more efficient and tamper evident

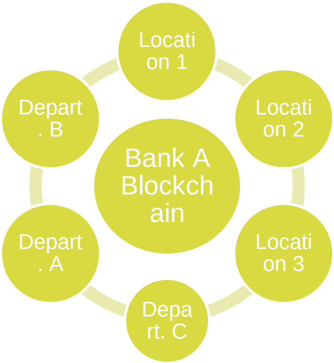
Multiple Blockchain models

Public



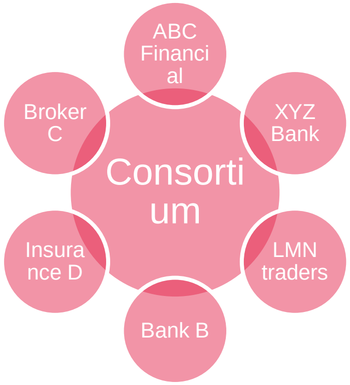
- CoMany, unknown participants
- Writes by all participants
- Reads by all participants
- nsensus by proof of work

Private



- Known participants from one org
- Writes permissions centralized
- Reads may be public or restricted
- Multiple algorithms for consensus

Consortium



- Known participants from multiple orgs
- Writes require consensus of n participants
- Reads may be public or restricted
- Multiple algorithms for consensus

Journey Map -- Considerations

- Questioning Traditional Systems
- Blockchain Benefits
- Is Blockchain the Answer?

2

Smart Contracts: Basics

Using and Enforcing Smart Contracts

- Characteristics of Blockchain
  - Decentralized
  - Trustless (usually)
  - Consensus Driven
  - Transparent
  - Immutable

Using and Enforcing Smart Contracts

- Characteristics of Smart Contracts
  - Transparent (within the chain)
  - Autonomous
  - Immutable
  - Self-Enforcing
  - Secure

Risks of Smart Contracts

- Transparency
  - Sometimes privacy is a good thing
  - What about effect on competition?
- Autonomous
  - Removes flexibility and ability to respond to unexpected events
- Immutable
  - If result is unexpected, can’t unwind; have to engage in another transaction to reverse
- Self-enforcing
  - Even if the parties realize there is a problem…
- Secure
  - Usually….

Considerations for Smart Contracts

- Most jurisdictions require an intent to agree and to be bound
  - What if the entire agreement is in source code?
- Impact of error in coding (see Smart Contract: Advanced)
  - Software bugs
  - Unanticipated result from accurate coding
    - COVID-19
- Enforceability of a “pure code” agreement
- Anonymity/pseudonymity of parties
- Jurisdictional issues / limitations

Jurisdictional Issues and Limitations

- State laws related to smart contracts
  - A number of states have adopted laws recognizing blockchain technology and addressing enforceability of smart contracts
  - Many of these laws amend the Uniform Electronic Transactions Act (“UETA”) to expressly recognize that blockchain transactions constitute electronic writings, and to provide for enforcement of smart contracts
  - Some states address other issues, such as taxation of cryptocurrencies
- Cross-jurisdictional issues and choice of law are critical

Jurisdictional Issues and Limitations

- In Europe, recognition of electronic signatures is based on the electronic Identification, Authentication and Trust Services Regulation (eIDAS)
  - To be legally binding, electronic signatures must use the services of a recognized Trust Service Provider (TSP); blockchain typically does not meet this standard
- Further, it is not clear whether blockchain-based timestamping will meet eIDAS standards
- This could raise questions regarding enforceability of certain smart contracts between US and EU entities unless Member States take action authorizing blockchain signatures

Jurisdictional Issues and Limitations

- However, jurisdictions are grappling with these issues, as is the blockchain industry
- The risk is that different jurisdictions are taking different positions; this could result in significant conflicts in characterization of cryptoassets and different interpretations of the meaning and enforceability of smart contracts
- Consistency across jurisdictions is important to ensure consumer protection, prevent anti-competitive activities, provide safeguards against criminal actions, and encourage reasonable investment in the technology.

3

Smart Contracts: Advanced

Amending Smart Contracts

- Hallmark characteristic is performance is automatic, self-enforcing
  - So some adjustments to the contract can be programmed into the agreement
    - For example, the smart contract between Bob and Alice provides that if Bob’s credit score is downgraded, pricing increases automatically. The smart contract has a process that permits automatic updating of credit score and adjustment of pricing that is transparently reflected on the blockchain

Amending Smart Contracts

- Amendment, rescission, and reformation are more complex
  - Traditionally, parties have the right to amend
  - Rescission and reformation are important tools in event of unilateral or mutual mistake, fraud, or unconscionability
  - Once a smart contract is created, it cannot be modified and any functions that are not included cannot be added

Amending Smart Contracts

- Contracts may need to be amended to deal with:
  - Versioning conflicts
  - Coding errors
  - Attacks within the contract
  - Changes to hard-coded addresses
  - Deprecated APIs
  - Off-chain events

Amending Smart Contracts

- Ethereum Solidity has “selfdestruct” functionality
  - This can be a solution for emergencies
  - Or, it can be an attack vector
- If a smart contract includes the selfdestruct functionality, it can be called by an authorized entity; this sends the Ether balance to a specified address and deletes the smart contract from the blockchain going forward
- For amendments or termination based on anticipated events, it may be possible to program functions that trigger on the occurrence or failure and create different states for the contract

Smart Contracts: A New Potential Source of Liability

- "When smart people hear the term ‘smart contracts’, their imaginations tend to run wild"
- Definition: *A self-executing contract written in computer programs that automatically execute the transaction if certain conditions under the programs are met*
- Can be used for many kinds of contracts: escrow, capital markets trading, real property and IP transfers, insurance claim processing, supply chain management and so on
- A classic analogy to “smart contract”: vending machine (if coin is inserted, then automatically provides soda)
  - Issue: many contracts are not so simple

Smart Contracts: Not so Smart

- Smart contracts are software: all software has bugs
  - McConnell, Code Complete: Industry Average: "about 15 - 50 errors per 1000 lines of delivered code." He further says this is usually representative of code that has some level of structured programming behind it
  - The National University of Singapore (NUS) uncovered several severe smart-contract bugs: out of the 19,366 Ethereum smart contracts they analyzed, 8,833 of them had bugs!
- Smart contract errors
  - TheDAO raised $165M in ether to fund Ethereum based projects (alternative to venture capital)
    - Hacker discovered flaw in the code and “diverted” $50M in ether
    - Community forked the Ethereum blockchain to recover the funds but significant minority view: **Code is Law** and did not fork
  - Parity Wallet: bug caused $30M in ether to be locked up in July 2017

Smart Contract Errors

- TheDAO raised $165M in ether to fund Ethereum based projects (alternative to venture capital) (2016)
  - Hacker discovered flaw in the code and “diverted” $50M in ether
  - Community forked the Ethereum blockchain to recover the funds but significant minority view: **Code is Law** and did not fork
- Lendf.me hack (2020)
  - $25M theft from Lendf.me but later returned
  - Incompatibility between ERC777 and DeFi smart contracts used by attacker to hijack a normal transaction and perform additional illicit operations (re-entrancy attack like TheDAO)

Hybrid Implementation Model is Likely

- Traditional contract law governing certain issues for smart contracts
  - Liability for coding errors.
  - Dispute resolution (arbitration probably preferred).
  - Indemnities for errors/breach of law.
  - Limitations of liability.
  - Disclaimer of consequential damages.
  - Ability to “unwind” illegal transactions.
  - Replacement of “oracles.”

4

Smart Contracts: Privacy & Security

Privacy and Security

- If data stored on the chain can be connected to individuals and is subject to privacy laws, consider:
  - Who is controller and who is processor?
  - What about written agreements between the nodes?
  - Who is responsible in the event of a breach somewhere along the chain?
  - What about decisions based on automated processing only?
- Use best practices to minimize risk to personal data
  - Encryption, data obfuscation, data aggregation
  - Have you selected the best possible based to process the data?

Privacy versus Anonymity

- Blockchains often provide less anonymity than rumored…
- If truly anonymous, regulatory compliance issues arise related to “Know Your Customer” and anti-money laundering rules
  - Tools have been developed that permit implementation of KYC and AML on blockchain
- Blockchain intermediaries can be subject to routine law enforcement demands

Cybersecurity

- Distributed network avoids the risk of a single point of failure – usually
- Consensus mechanism helps prevent hacks through a single node; to compromise the blockchain requires attack at multiple nodes
- Transparency discloses attempts to manipulate the blockchain more quickly

Cybersecurity

- Cybersecurity Risks/Hacks
  - Coding errors and protocol vulnerabilities
    - The DAO exploit - $55 million
    - How do you amend a smart contract once it is in flight?
  - Identity-based attacks: Spoofing and Sybil attacks
    - Real identities are spoofed, or new, false identities are created, to flood the network with new nodes and connect to honest participants
    - Sybil attacks are somewhat mitigated by requiring proof of work or other investment in the blockchain

5

Smart Contracts: Liability Issues

New Theory: Coders of Public Blockchain Protocols as Fiduciaries

- Professor Walch: IN CODE(RS) WE TRUST: SOFTWARE DEVELOPERS AS FIDUCIARIES IN PUBLIC BLOCKCHAINS
- Public blockchains are critical infrastructure
- Governance is based on decisions relating to software development and implementation of the consensus protocol
- Critical issues
  - Who is responsible for coding decisions in a decentralized ecosystem
  - Possible application of fiduciary duty (similar to officer and directors in corporations as well as lawyers and doctors)
- Focuses on developers of “blockchain clients” not developers of “smart contracts” which depend on the blockchain clients

Advantages according to Walch

- Encouraging developers to perform their duties with deliberation and care
- Reduce harm by developers to others by acting without care or exploiting their role
- Increase efficiency and economic activity due to reduction in investigation and due diligence needed
- Creation of an accountability standard to match seriousness of their responsibilities

Disadvantages according to Walch

- Potential inhibition of innovation
- Blockchains are platform technologies and legal intervention should be at the application layer
- Too extreme and too high a duty to place on the individuals
- Impossible to determine whether fiduciary duty standards are met because of diverse interests of “entrustors” (or beneficiaries)
- Deter programmers from participation in blockchain project
- Users should do their due diligence and this approach is paternalistic
- Unfair since developers do not expect this liability
- Developers are not compensated as fiduciaries

Tort Liability for Software Errors

- ***"Broadly speaking, a tort is a civil wrong, other than a breach of contract, for which the court will provide a remedy in the form of an action for damages."***
- Tort Theories (US)
  - Negligence
  - Strict liability
- Limits: Economic loss doctrine, limited to personal damages and property damages (no lost profits without other harm)

Tort Law in the US

- State law with some special federal laws
- American Law Institute has developed influential “Restatements” of the law, but the Restatement may be adopted by the relevant state courts
- Status of Restatements
  - Restatement (Second) of Torts (1963/1979)
  - Restatement (Third) of Torts (2000)
- Tort law in many states reflects principles found in the Restatement (Second) of Torts and over time, increasingly from the Restatement (Third) of Torts. However, state laws vary widely

Negligence Theory: Restatement (Second) of Torts

- § 282. Negligence Defined
- In the Restatement of this Subject, negligence is conduct which falls below the standard established by law for the protection of others against unreasonable risk of harm. It does not include conduct recklessly disregardful of an interest of others.
- § 285. How Standard of Conduct is Determined.
- The standard of conduct of a reasonable man may be established by a legislative enactment or administrative regulation which so provides, or adopted by the court from a legislative enactment or an administrative regulation which does not so provide, or established by judicial decision, or applied to the facts of the case by the trial judge or the jury, if there is no such enactment, regulation, or decision.

Negligence Theory: Restatement (Third) of Torts

- Negligence Definition
  - A person acts negligently if the person does not exercise reasonable care under all the circumstances. Primary factors to consider in ascertaining whether the person's conduct lacks reasonable care are the foreseeable likelihood that the person's conduct will result in harm, the foreseeable severity of any harm that may ensue, and the burden of precautions to eliminate or reduce the risk of harm.

Strict Liability in Tort: Restatement (Second) of Torts

- § 402A. Special Liability of a Seller of Product for Physical Harm to User or Consumer.
  - A product is defective when, at the time of sale or distribution, it contains a manufacturing defect, is defective in design, or is defective because of inadequate instructions or warnings. A Product:
    - contains a manufacturing defect when the product departs from its intended design even though all possible care was exercised in the preparation and marketing of the product;
    - is defective in design when the foreseeable risks of harm posed by the product could have been reduced or avoided by the adoption of a reasonable alternative design by the seller or other distributor, or a predecessor in the commercial chain of distribution, and the omission of the alternative design renders the product not reasonably safe;
    - is defective because of inadequate instructions or warnings when the foreseeable risks of harm posed by the product could have been reduced or avoided by the provision of reasonable instructions or warnings by the seller or other distributor, or a predecessor in the commercial chain of distribution, and the omission of the instructions or warnings renders the product not reasonably safe.

Product Liability: Restatement (Third) of Torts

- One engaged in the business of selling or otherwise distributing products who sells or distributes a defective product is subject to liability for harm to persons or property caused by the defect.
- A product is defective when, at the time of sale or distribution, it contains a manufacturing defect, is defective in design, or is defective because of inadequate instructions or warnings. A product:
  - (a) contains a manufacturing defect when the product departs from its intended design even though all possible care was exercised in the preparation and marketing of the product;
  - (b) is defective in design when the foreseeable risks of harm posed by the product could have been reduced or avoided by the adoption of a reasonable alternative design by the seller or other distributor, or a predecessor in the commercial chain of distribution, and the omission of the alternative design renders the product not reasonably safe;
  - (c) is defective because of inadequate instructions or warnings when the foreseeable risks of harm posed by the product could have been reduced or avoided by the provision of reasonable instructions or warnings by the seller or other distributor, or a predecessor in the commercial chain of distribution, and the omission of the instructions or warnings renders the product not reasonably safe.

Product Liability: Restatement (Third) of Torts (Components)

- One engaged in the business of selling or otherwise distributing product components who sells or distributes a component is subject to liability for harm to persons or property caused by a product into which the component is integrated if:
  - (a) the component is defective in itself, as defined in this Chapter, and the defect causes the harm; or
  - (b)
    - (1) the seller or distributor of the component substantially participates in the integration of the component into the design of the product; and
    - (2) the integration of the component causes the product to be defective, as defined in this Chapter; and
    - (3) the defect in the product causes the harm.

Challenges of Applying Tort Theory to SW Development

- Negligence
  - Lack of reasonable man
  - Proof of causation
  - Substantial factor
- Strict Liability in Tort
  - Limited to certain types of products
  - Policy decision by courts

Liability for Regulatory Violations in SW Development

- Commodity Futures Trading Commission ("CFTC")
  - Commissioner suggested that "smart contract" developers (not blockchain protocol developers) are liable for software which violates CFTC regulations
  - Is foreseeability part of the analysis
- Securities and Exchange Commission ("SEC")
  - EtherDelta decision

Liability for Regulatory Compliance (Prediction Markets): CFTC Commissioner Brian Quintenz

- That leaves us with the developers of *the smart contract code that underlies these event contracts, as well as the individual users* who then use that code to create and wager on their own event contracts. The developers of the code could claim that they merely created the protocol and therefore have no control over whether and how users choose to use it once it is part of the public domain. They would place the liability on the individual users, who are the actual creators and counterparties of the event contracts.
- In my view, this analysis misses the mark. Instead, I think the appropriate question is whether these code developers could reasonably foresee, at the time they created the code, that it would likely be used by U.S. persons in a manner violative of CFTC regulations. In this particular hypothetical, the code was specifically designed to enable the precise type of activity regulated by the CFTC, and no effort was made to preclude its availability to U.S. persons. Under these facts, I think a strong case could be made that the code developers aided and abetted violations of CFTC regulations. As such, the CFTC could prosecute those individuals for wrongdoing

Quintez Response to Coin Center Criticism

- After criticism by Coin Center, he expressed a willingness to review:
  - *I appreciate this thoughtful analysis of my recent speech’s content (and tone), and I look forward to working with* ***@coincenter*** *and others to clarify some misinterpretations and reach mutual understandings on these challenging & fascinating issues*

Liability for Regulatory Violations: EtherDelta

- Coburn founded and ran EtherDelta which was a “decentralized exchange” for trading EC20 tokens
- He wrote the code for EtherDelta
- He did not register as an ATS as required for exchanges trading securities as required by the Securities and Exchange Act of 1934
- SEC investigated and reached a settlement with the following finding:
  - *During the relevant period, Coburn founded EtherDelta, wrote and deployed the EtherDelta smart contract to the Ethereum Blockchain, and exercised complete and sole control over EtherDelta’s operations, including over the operations constituting the violations described above. Coburn should have known that his actions would contribute to EtherDelta’s violations and thus, under Exchange Act Section 21C(a), caused EtherDelta to violate Section 5 of the Exchange Act.*

6

Smart Contracts: Appendix

New Theory: Coders of Public Blockchain Protocols as Fiduciaries

- Professor Walch: IN CODE(RS) WE TRUST: SOFTWARE DEVELOPERS AS FIDUCIARIES IN PUBLIC BLOCKCHAINS
- Public blockchains are critical infrastructure
- Governance is based on decisions relating to software development and implementation of the consensus protocol
- Critical issues
  - Who is responsible for coding decisions in a decentralized ecosystem
  - Possible application of fiduciary duty (similar to officer and directors in corporations as well as lawyers and doctors)
- Focuses on developers of “blockchain clients” not developers of “smart contracts” which depend on the blockchain clients

Open Issues in Walch Proposal

- Which developers are responsible and are fiduciaries?
  - Walch is unclear using the following formulations (see footnote 244 in Haque et al article)
    - “core developers” (pg 6)
    - “prominent developers” (pg 7)
    - “software developers” (pg 7 discussing the BTC fork)
    - “key developers” (pg 7 discussing the BTC fork)
    - “dominant developers” (pg 8 discussing the ETH fork)
    - “small number of developers” (pg 8 discussing the ETH fork)
  - Walch also notes that those who shape the code/functions but do not write code
- Who are the “entrustors” to whom a fiduciary duty is owed?
- What is the nature and scope of the fiduciary duty?

Industry Response: Haque/Plummer/Rosario I
Blockchain Development and Fiduciary Duty

- Developers are not “agents” of network participants (blockchain technology is implemented through clients running a particular protocol: the Bitcoin network is implemented through 22 different “clients” other that Bitcoin Core)
  - No authority to bind other network participants
  - Influence (speaking for the community is not enough)
- No delegation of power or authority by network participants
- Effect of changes to software on network participants is very attenuated

Industry Response: Haque/Plummer/Rosario II
Blockchain Development and Fiduciary Duty

- Other alternatives make “fiduciary duty” application unnecessary (Tamara Franklin limits fiduciary duty to relationships where the “entrustor” cannot otherwise protect himself from abuse of power)
- Developers not incentivized to “act” improperly
- Other problems:
  - Impractical to interpret:
    - Which “developer” owes a fiduciary duty?
    - Who is the beneficiary of the fiduciary duty?
  - Risk of developers abandoning projects

References

- Walch **In Code(rs) We Trust: Software Developers as Fiduciaries in Public Blockchains** https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3203198
- Haque, Seira, Plummer & Rosario **Blockchain Development and Fiduciary Duty** https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3338270
- CDT **Strict Products Liability and the Internet of Things** https://cdt.org/blog/when-iot-kills-preparing-for-digital-products-liability/

Force Majeure in Smart Contracts

- 3 major issues:
  - Defining the force majeure event
  - Defining the impact of the force majeure event
  - Deciding whether force majeure is triggered automatically, by code, or requires human intervention

Force Majeure in Smart Contracts

- We have all learned that boilerplate force majeure may not be the best approach…
- Force majeure events are often challenging to define
- What about mission-critical contracts?
- What about contracts in which one or both parties are required by regulation to have disaster response/ COOP?
- What about situations where partial performance or best efforts recovery is desired?
- What about common law impossibility of performance?