Don’t Dismiss Accenture’s Blockchain Redaction Solution — You May Need It One Day
Not “Immutability,” But Well-Governed, Tamper-Evident Systems Should Be Your Goal
by Martha Bennett
April 19, 2017

Why Read This Report
When Accenture first published its proposal for an “editable blockchain,” the press release was met with a certain degree of skepticism. After all, isn’t immutability supposed to be an essential feature of a blockchain? For CIOs assessing the technology in an enterprise context, however, the concept makes sense and doesn’t run counter to desired characteristics of a permissioned blockchain environment. Read this report to find out how it works and why it shouldn’t be dismissed without due consideration.

Key Takeaways
Immutability Is A Double-Edged Sword
Regardless of whether complete immutability is even possible, there are times when it’s not desirable and when existing standard mechanisms for dealing with mistakes or bugs in a blockchain aren’t acceptable.

The Ability To Alter A Blockchain Record Isn’t As Outrageous As It May Sound
In any permissioned blockchain network, the integrity of records is only as strong as the governance mechanisms applied to their entry and subsequent correction in the case of errors, fraud, or bugs. Seen in this context, the governance principles afforded by Accenture’s proposed blockchain redaction capability are unlikely to be any more vulnerable than existing alternatives, as well as technically less risky.
Don’t Dismiss Accenture’s Blockchain Redaction Solution — You May Need It One Day

Not “Immutability,” But Well-Governed, Tamper-Evident Systems Should Be Your Goal

by Martha Bennett
with Pascal Matzke, Jost Hoppermann, Michael Glenn, and Ian McPherson
April 19, 2017

All Enterprise Blockchains Need Error Correction Mechanisms

“Immutability of records”: When asked why they’ve chosen to investigate blockchain technology or embark on PoCs, pilot projects, or early deployments, this is typically the first benefit cited by those working on them. And in a permissionless system such as the public Bitcoin or Ethereum networks, in which participants don’t know each other and anybody can join, it is indeed a necessity to make it essentially impossible for anyone to make changes once data has been added to a block and the block accepted onto the chain. However, in an environment of permissioned blockchains — which the majority of enterprise blockchains will be — immutability can be as much of a curse as it is a blessing.

The reason for this is exception management, a topic that Forrester finds is often treated as an afterthought in blockchain projects. There are four main categories of exceptions a blockchain project team must consider before making technology choices. These are:

› **Straightforward input errors.** Trader “fat finger syndrome,” a wrong number, or name — it’s easy to make a mistake when entering data, and erroneous records will sooner or later end up on a blockchain.

› **Fraudulent transactions.** We all know the old joke about the criminal who, when asked why he robbed banks, answered “that’s where the money is.” As history has shown, and we continue to discover at all too regular intervals, the criminal fraternity is capable of great ingenuity when it comes to devising fraudulent schemes, and it would be careless to assume that fraudulent transactions won’t sooner or later appear in a blockchain-based system.

› **Bugs in the code.** This applies both to the chaincode itself and any so-called “smart contracts” that may be running on it. Given the nascent stage of the technology and the complexity of many of the planned architectures, it’s clear that there’ll be bugs to iron out for some time to come.
Errors or outdated terms in smart contracts. Aside from the potential need to remedy bugs in the software, there’s an additional issue when it comes to so-called smart contracts: By definition, a smart contract is only as good as the person or teams who devise the business rules and the developer(s) who translate those rules into software. Without a well-governed mechanism to deal with wrong, incomplete, or outdated logic in smart contracts, these constructs are likely to cause problems that exceed the benefits they’re intended to bring. There will also be times when smart contracts need to be updated to reflect new or changed regulatory requirements.

Complete Immutability Is An Illusion — And Not Always Desirable

Despite what some may claim, true 100% immutability is technically not possible. Yes, it can be made prohibitively expensive and time-consuming to try and subvert a blockchain, and there may also be economic reasons and other motivations for not doing so; Bitcoin has so far proven that this can indeed be the case. However, in any permissioned blockchain environment, “immutability” is really the result of the governance principles agreed upon by the participants and how these are applied, enforced, and audited; commercial interests obviously come into it as well. But from a purely technical perspective, it would be comparatively easy for participants to collude and subvert a chain.

Leaving the immutability argument to one side, there are circumstances when the inability to make changes to a record (including a smart contract) on a chain becomes a potentially major issue. For example:

- **Situations will arise when a record must be deleted.** For instance, if the wrong transaction consists of a highly confidential record being accidentally posted to the wrong owner; or if a false credit score or other damaging information is recorded by mistake; or if a person chooses to exercise their right to be forgotten. In all those cases, a mechanism is required to allow removal of a record from the chain, as posting a countermanding record clearly wouldn’t have the desired effect.

- **Sometimes, a record or smart contract must be updated.** Even when there’s no need for complete deletion, there are occasions when a correction is required. For example, a bug in a smart contract may result in the creation of erroneous or even illegal transactions; or a smart contract is no longer fit for purpose because a key rule wasn’t encoded or there’s been a change in regulation. The standard methods of making corrections may not be suitable in many of these circumstances. Recomputing the entire chain from the point of error onward is a cumbersome process and can get messy; in a complex and large system, it could even give rise to further errors and compromise the chain.

Accenture’s Blockchain Redaction Capability Offers A Solution Without Violating Integrity

It’s against the backdrop of situations like the ones described in the previous section that Accenture developed its concept of a “redactable” or “editable” blockchain. It provides a technically neat way of making corrections at the point in the chain where they need to be made, without touching any of the surrounding blocks; support for strong governance mechanisms ensures that data integrity is preserved. This is how it works (see Figure 1):
The secret sauce is a so-called “Chameleon Hash.” This goes alongside the standard hash between blocks. Unlike the typical hashes linking blocks, the Chameleon Hash can be opened with a private key; not that it’s likely anybody would wish to deploy the system with just a single key to unlock the hash. Keys would typically be fragmented (“sharded”) and given to various administrators, so they would need to come together and agree on any edits, which would minimize the risk of unauthorized changes. Accenture has also created an API that integrates major blockchain platforms with the industry standard hardware security modules (HSMs) to protect and store the sharded Chameleon hash keys.

Breaking the links between blocks means they can be altered without side effects. Once a block has been isolated from the preceding and following block, it can be edited or replaced as needed. When the new block is finished, valid hashes are created that link it to its predecessor and successor. Because the Chameleon hash values for the old and new block remain the same, the new block will hash to the chain without any need to recompute and validate the entire chain.

Various mechanisms ensure that the system isn't abused. First of all, it’s designed to leave a trace: breaking the standard hash that links two blocks always leaves a scar — in the case of multiple changes to a block, there’ll be multiple scars. Depending on the requirements of a particular use case, it’s also possible to keep old blocks for future inspection. Unlocking the Chameleon Hash is of course a matter of governance: It’s up to each blockchain network’s participants to decide under what circumstances an edit or deletion is appropriate and permissible; how many keys are needed to unlock the hash; who should hold those keys; how the unlocking of the hash and altering of the block are supervised; whether old blocks should be retained; and so on.
**FIGURE 1** High-Level Overview Of How Accenture’s Blockchain Redaction System Works

A standard hash link and a Chameleon Hash link are formed between each block.

After a block is edited, the original hash is broken, leaving a permanent “scar” to indicate that a change was made and a new standard link is created in its place.

This process can be repeated any number of times. Versions are visible through each broken hash.

Based on original graphic Copyright 2016 Accenture All rights reserved.

**Recommendations**

**Never Lose Sight Of How You’re Going To Meet Governance Needs**

Don’t allow yourself to get caught up in the debate about so-called immutability of records. Instead, focus on making sure that your organization’s blockchain initiative doesn’t end up putting the (technology) horse in front of the (business and regulatory requirements) cart; this applies whether you’re driving the project or are participating in somebody else’s.\(^4\) Instead, focus on the governance requirements of your use case(s). In the context of preserving the integrity of records, make sure that your project leaders consider carefully:

› **What mechanisms are in place to deal with exceptions.** Whether it’s errors, fraudulent transactions, bugs in the code, or errors on smart contracts, they all require strong governance mechanisms to ensure they can be dealt with in a way that doesn’t end up being a potential vulnerability in the system. When seeking assurances on these points, make sure that you get insight into the end-to-end system, not just a partial view of the blockchain element in isolation.
Whether you may encounter the need to amend or delete records or code on the chain. If you do, you must ascertain what mechanism is going to be used for achieving this, and whether these methods are acceptable, both from a compliance and a technical perspective. In all cases, you’ll need to be able to prove how you’re making sure that changes are legitimate and any unauthorized changes are obvious.

Endnotes

1 Strictly speaking, “smart contract” is a misnomer; these constructs are neither smart nor are the contracts in a legal sense unless a separate, ‘real-world’ contract is in place to govern the applicability of the terms captured in the software.

2 Legislation which may require you to remove data from a blockchain or distributed ledger system includes, but isn’t limited to the US Fair Credit Reporting Act, the US Gramm-Leach Billey Act, US SEC “Regulation S-P,” and the “Right to be Forgotten” under the EU’s data privacy rules.
Don't Dismiss Accenture’s Blockchain Redaction Solution — You May Need It One Day
Not “Immutability,” But Well-Governed, Tamper-Evident Systems Should Be Your Goal


We work with business and technology leaders to develop customer-obsessed strategies that drive growth.

PRODUCTS AND SERVICES
› Core research and tools
› Data and analytics
› Peer collaboration
› Analyst engagement
› Consulting
› Events

Forrester’s research and insights are tailored to your role and critical business initiatives.

ROLES WE SERVE

Marketing & Strategy Professionals
CMO
B2B Marketing
B2C Marketing
Customer Experience
Customer Insights
eBusiness & Channel Strategy

Technology Management Professionals
› CIO
Application Development & Delivery
Enterprise Architecture
Infrastructure & Operations
Security & Risk
Sourcing & Vendor Management

Technology Industry Professionals
Analyst Relations

CLIENT SUPPORT
For information on hard-copy or electronic reprints, please contact Client Support at +1 866-367-7378, +1 617-613-5730, or clientsupport@forrester.com. We offer quantity discounts and special pricing for academic and nonprofit institutions.