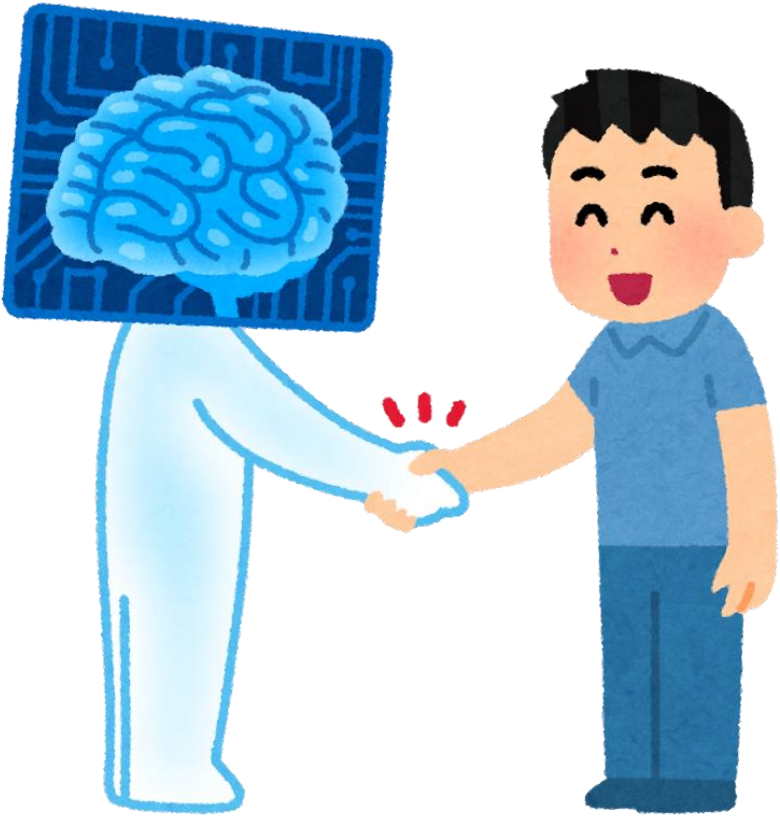
監督責任時代の

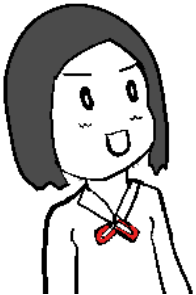
完全に理解『しない』スマートコントラクト開発

最後に責任を取るのは…人間

目次

時代の変化を見極める





過大評価せず
過小評価しない

Introduction

この本の見方

タイトル
ページの内容を一言で表現しています

本文
読みやすい日本語で記載しています

すごいちゃん
感想を言ってくれます

最近のスマートコントラクト開発における事故予防の本です

今回のテーマ：AIと責任、不可逆性

This book is for

想定読者

フルスタック開発者
今度の開発案件、Web3らしい。
不足の知識は、Solidityの文法？
まぁ、AIに聞けば、分かる！

それは…勘違いです…

学生
AI時代のブロックチェーン開発の注意点について学びましょう

起業志望者
リスク管理が大切です。
一緒に活動するエンジニアに、ブロックチェーン開発の注意点を押さえてもらいましょう。

AI調査中の人
ブロックチェーン領域だとAIはどのように使うといいのかをお伝えします。

初心者
基本を押さえましょう。

技術書典に来た人
今回の本は無料配布です！
記念にもらっていって下さい

コントラクト開発の落とし穴を紹介する本です

What is Blockchain?

ブロックチェーンを、ひとことでいうと？

データの保存先

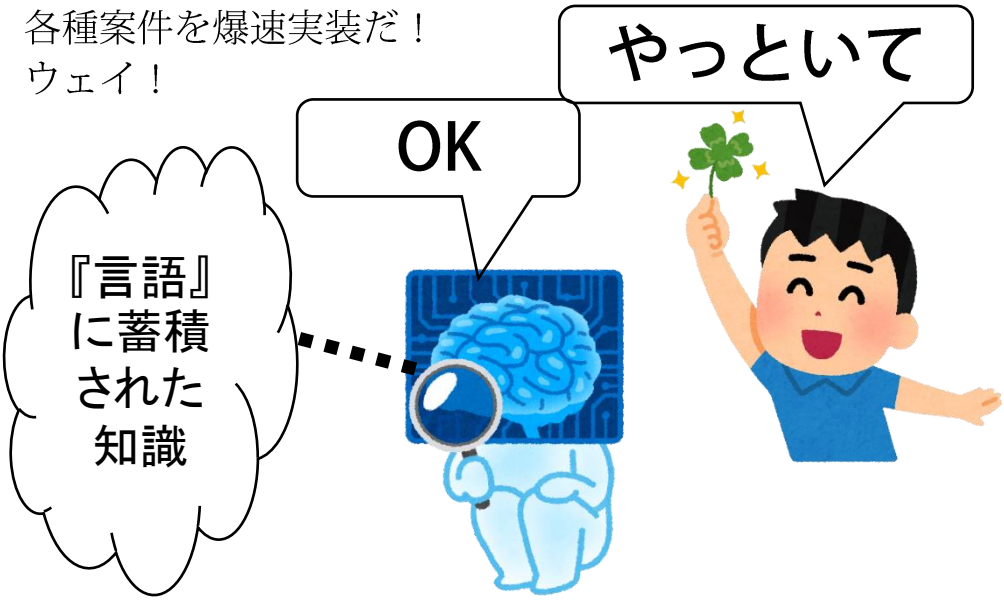
ブロックチェーンは結局のところデータの保存先だ。ブロックチェーンといえばBitcoinが有名だが、Bitcoinでは利用者がコインを送金した記録が保存されていく。Bitcoin以外にも様々なブロックチェーンがあるが、技術的に注目したいのは、ブロックチェーンの記録データは後から書き換えや消去が『できない』ことだ。一度記録した内容は、システム管理者や国家権力にとっても、変更ができない。その仕組みは別の専門書を当たって頂きたいが、書き換えの出来ない記録…このことは70年に渡るコンピュータの歴史の中で初めて実現された快挙だ。…議論好きな方であれば、書き換え不能なCDとか昔からあるよと反論されるかもしれない。では、書き換え不能なCDがあったとしよう。権力側はそれを捨て去ることができるのだ。消すことができる。しかし、ブロックチェーンであれば、そうはいかない。書き込まれたデータは他の人の管理する装置にすぐコピーされる。関連する装置をすべて破壊するまで、データを消し去ることはできない。ブロックチェーンに記録した内容は、世界に伝搬し、永遠に残り続ける。データへは個人の秘密鍵を通じてアクセスできる。僕らは初めて、電子的な公共を手に入れたのだ。

そして、この保存されたデータ、あるものは価値と呼ばれ、あるものは歴史と呼ばれる。これを、人と機械が公平に使っていくことになる。

1.『完全に理解した！』

ブロックチェーン完全に理解した！

時代はフルスタックエンジニア！
知らないこともChatGPTに質問して、
各種案件を爆速実装だ！
ウェイ！



ChatGPTからの出力！(Solidityのコード！)

scss Copy code

```
pragma solidity ^0.5.0;

interface IERC20 {
    function balanceOf(address owner) external view returns (uint256);
    function transfer(address to, uint256 value) external returns (bool);
```

キーボード打たずにプログラム書ける

理解が容易になる世界

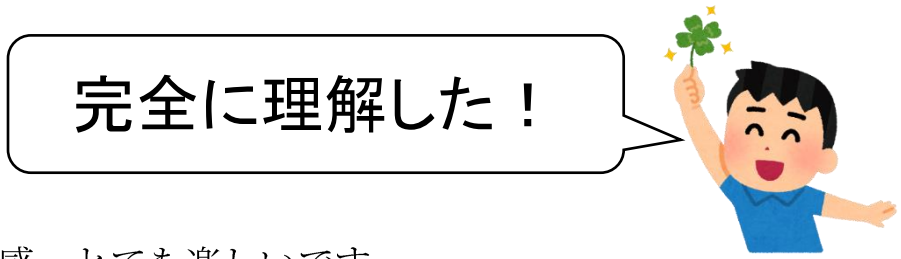
簡単に成果が出る世界

Solidityのコードを使ってスマートコントラクトを動かすための、事前に必要な学習は、大幅に少なくできました。

【事前に必要な学習】
・一般的なプログラミングの知識
・ChatGPTへの質問の仕方のコツ

【ChatGPTが簡単に教えてくれること】
・Solidtyという言語を使うらしいということ
・デプロイツールの選び方と動かし方

プロンプトを入力して、コピー＆ペーストで開発する方法もありでしょう。簡単です。完全に理解できますね。



万能感。とても楽しいです。
ここで、有名な心理学の理論を引用してみましょう。
…ダニング・クルーガー効果。
『経験の低い人は自分の能力を過大評価してしまう』理論。
そして、コントラクト開発はこの理論が重要なのです。

今回のテーマはAIと不可逆性

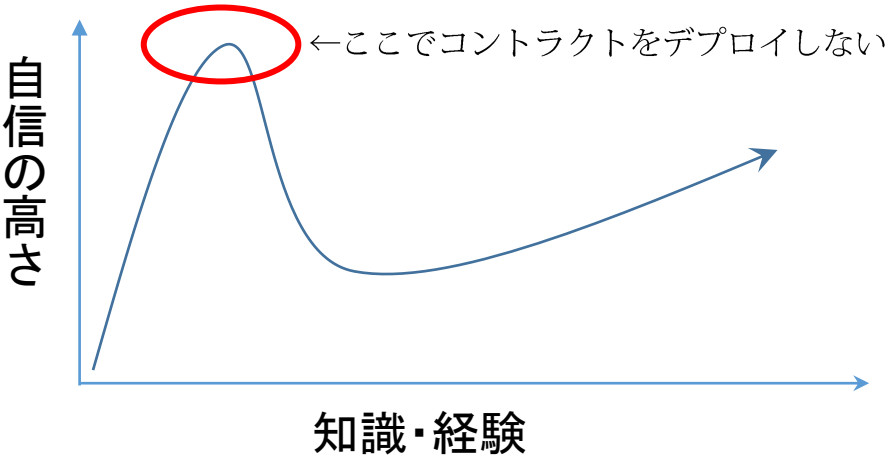
2.『ダニング・クルーガ効果』

完全に理解しやすい

そう、最近は、世の中が簡単になっているのです。
AIが人間のサポートをしてくれて、60点ぐらいの出力が、
簡単に得られるようになっています！
では、大切なことをお伝えしますね。

理解しておいてほしいこと

スマートコントラクト開発、60点じゃダメなんだよぉぉ！
いわゆるダニング・クルーガ効果の***『完全に理解した！！』***のタイミングで、コントラクトをデプロイすると大変なことになります。



その理由を次ページで説明していきます。

60点じゃダメなんだよぉぉ！

自信過剰は事故のもと

なぜ60点ではダメなのか

開発したものが60点のまま、一生残る。
そして60点のコントラクトは、事件を起こしてしまいます。
「とりあえず動いたからヨシ！」の考え方は、とても危険です

「一応動く」と「セキュリティ上問題なく動く」の間に、
大きな壁があるのが、スマートコントラクト開発なのです。

ことわざから学ぼう

この感覚、なかなか伝わりづらいと思います。

分かりやすく伝えるために『ことわざ』を使おうと思います
開発時の考え方を、有名な言葉でお伝えしていきます。

- 後悔先に立たず
- 壁に耳あり障子に目あり
- 人を見たら泥棒と思え
- 覆水盆に返らず

どうしたらいいの？

- 『石橋を叩いて渡る』です。そう、気を付けて。

→次ページから解説していきます。

コントラクト開発、気を付けて

3.『後悔先に立たず』

後悔したときに何が起こるか

このページでは『後悔先に立たず』を見ていきます。
スマートコントラクト開発、後悔したら何が出来るでしょう?
次から選んでみて下さい。
A.リファクタリング　B.バグ修正　C.後悔

答えは…C.『後悔』です！
『リファクタリング』も『バグ修正』も出来ません。

代表的な後悔

コントラクト開発時の後悔ををざっくりと紹介します。

名称	お金が なくなる	データが 修正できない	利用料 (ガス代)が高い
説明	金銭管理システムなので失敗で資金が流出してしまう	システム管理者によるデータ修正ができません	ブロックチェーンには利用料がかかり、ガス代と呼ばれます
特徴	後悔：★★★★★ お金が盗まれてしまうと、後悔や事件につながります。	後悔：★★★★☆ 事前にUpgradableという仕組みを入れていないと、原則として修正できません。	後悔：★★★☆☆ ガス代のことを考えて設計しないと、後で大きなコストがかかることがあります

後で変更が出来ない世界

後悔を予防しよう

よくある後悔を具体的に説明

・デプロイしたNFTの画像が表示されない。
・コントラクトからお金が取り出せない。
・管理者用の関数が一般開放されてしまった。
・if文の境界値がずれてしまった。
・ブロックチェーン使用料(ガス代)がとても高い
・コントラクト同士の連携がうまくいかない
・コントラクトが予期せぬデッドロックになってしまった
・データの更新がうまくいかない
・秘密鍵が盗まれてお金がなくなってしまった

…など、など。

後悔を予防する方法

後悔を予防するための、いろいろな工夫を見てみましょう

・**テストネットによる予防**
→デプロイする前にテストネットで動作確認をしましょう
・**ライブラリによる予防**
→ OpenZeppelinなど安定したライブラリを使いましょう
・**ヒトによる予防**
→コントラクトに詳しい人が問題が見つけてくれるかも

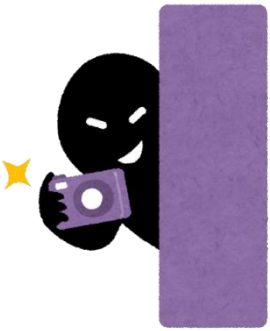
いろいろな工夫をして、後悔を未然に防ぎましょう

よくある後悔を予防する本です

4.『壁に耳あり障子に目あり』

オンチェーンのストーカー

このページで『壁に耳あり障子に目あり』を見ていきます。
隠すことができないのです。
オンチェーンの情報は、秘匿できません。
(サーバ上のデータは秘匿可能)



街にストーカーがいるのと同じように
オンチェーンにもストーカーがいます。

ストーカーの種類

オンチェーンにいるストーカーをざっくりまとめました。

名称	投資マニアのストーカー	リプレイアタッカー	コードを盗む者
説明	他ユーザがどんなトークンを買ったか分析してまねる	他の人の処理を解析し似たデータを送ると不正実行されうる	許可なくコントラクトをコピーしアプリを作る人
特徴	迷惑：★☆☆☆☆ 知力：★★★☆☆ 人数：★★★★★ 投資の上手いアドレスを観察している人が沢山います。	迷惑　：★★★★☆ 知力　：★★★★★ 調査力：★★★★☆ この攻撃が起こるのはコントラクトが設計ミスの場合	迷惑　：★★☆☆☆ 知力　：★★★★☆ 功名心：★★★★☆ 逆にコピーOKのライセンスで公開されているコードもある

秘匿できる情報がない

情報は全てオープン

代表的な攻撃①：直コン

まず、ブロックチェーンの攻撃を１つ確認してみましょう。
未公開のシステムを利用されてしまうことがあります。

この攻撃を『直コン』と呼びます。
名前の由来は、システム内のコントラクト直接呼出です。

コントラクト側のサービス開始フラグにより対策可能です。

代表的な攻撃②：リプレイアタック

では、次の攻撃も確認してみましょう。
『リプレイアタック』です。

秘密のキーワードを扱うシステム等で起こるトラブルですが、
他ユーザの送ったキーワードが、ブロックチェーン上で普通
に確認できますので、キーワードを流用されてしまいます。

コントラクト上でしっかりと設計を行い対策しましょう。

他にもいろいろ

ここで紹介した以外にも、いろいろな攻撃手法と対策が
あるため、網羅的な学習が必要です。

一番の基本として、ブロックチェーンはオープンなのです。
オープンなシステムを秘密鍵で管理しているのです。

オープンな台帳をウォレットで管理する

5.『人を見たら泥棒と思え』

オンチェーン泥棒

このページでは『人を見たら泥棒と思え』を見ていきます。
お金を盗もうとしてくるのです。

…ブロックチェーンの世界では、泥棒がいます。
この泥棒は、ネット上にいて、住所も分からないのです。
どこの国の法律で裁けるのかもわかりません。
正体不明で、お金だけ手に入れていくのです。

泥棒の種類

ブロックチェーンで出てくる泥棒をざっくりまとめました

名称	MetaMask サポート(偽)	闇のスーパー ハッカー	5ドルレンチ アタッカー
説明	何らかの公式団体を偽って秘密鍵をだまし取る	DeFiなどのシステムの脆弱性を見つけてお金を盗みとる	仮想通貨を大量に持っている人を武器で脅して巻き上げる
特徴	知力：★★★☆☆ 胆力：★★☆☆☆ 話術：★★★★☆ 海外の貧しい国からの犯行が多い。 逮捕されにくい。	知力　：★★★★★ 胆力　：★★★☆☆ 技術力：★★★★★ スーパーハッカーによる個人犯行。 逮捕されにくい。	知力：★☆☆☆☆ 胆力：★★★★☆ 筋肉：★★★★★ 力こそパワー。 監視カメラに映るとすぐ捕まる。

オンチェーン泥棒は検挙できない

秘密鍵を守ろう

秘密鍵はとても大切

最もよくある、秘密鍵の盗難について深堀します。
秘密鍵の管理は、まさに命綱です。
この鍵が盗まれてしまうと、コントラクトの管理者権限が奪われてしまいます。しっかりと守りましょう。

防衛策：鍵を守る方法

秘密鍵を守るためのポイントを、箇条書きにしていきます。

盗難防止の観点

- ウイルス感染の防止
- 普段使いのPCと秘密鍵を分離する
- 秘密鍵を他人/外部システムに渡さない
- 秘密鍵をGitHubなどに公開しない
- シードフレーズのメモを盗まれないようにする

紛失防止の観点

- 鍵管理用デバイスの故障に備える
- シードフレーズの紛失に備える
- 火災や交通事故に備える

法律の観点

- ユーザの秘密鍵を預からない(カストディ規制)
  ※暗号資産交換業者ならOK

秘密鍵が盗まれてしまうと大変です

6.『覆水盆に返らず』

取り返しがつかない

このページでは『覆水盆に返らず』を見ていきましょう。
ロールバックができないのです。
これは大変です。

失敗したら、取り返しが不能。
そして、管理対象は金融。
国や法律がブロックチェーンに手を出すこともできません。

データベースとブロックチェーンの違い

一般的なシステムとブロックチェーンの違いをまとめます。

名称	ロールバック不可	Adminによる修正ができない	ほぼ永遠に残る
説明	ブロックチェーン上の記録は過去の状態に戻せない	不都合なデータや更新したいデータを管理者が修正できない	ブロックチェーンでコントラクトは動き世界中の装置に残る
特徴	怖さ：★★★★★ 諦め：★★★☆☆ 大きな事故が起きた後、誰にも取り返しがつきません	怖さ　：★★★★☆ 公平さ：★★★★★ 一般的なDBシステムであれば、管理者による修正が可能です	怖さ　：★★★☆☆ ロマン：★★★★★ 管理者がいないからこそ、作った物が、そのまま残っていく

ロールバックできない

価値を扱うシステム

代表的な事件

今までのブロックチェーンで起こった
『取り返しのつかない』事件をいくつか紹介します。

・**ネム流出事件**：被害額_約580億円
→仮想通貨取引所Coincheckのウイルス被害により、
580億円相当のXEMが盗まれてしまった事件
Coincheckは被害額をユーザに自腹で補填しました。

・**BadgerDAOハッキング**：被害額_約130億円
→BadgerDAOという資金管理ソフトの画面にウイルスが
仕込まれていて、ユーザの資金が盗まれてしまった。
ユーザにお金は戻ってきませんでした。

・**WORMHOLEハッキング**：被害額_約370億円
→SOLANAのブリッジから資金が盗まれた事件
ブリッジにお金は戻ってきませんでした。

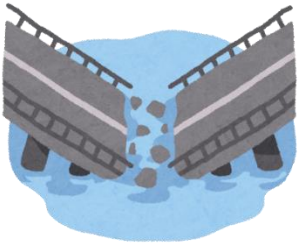
・**Euler Financeフラッシュローン攻撃**：被害額_約263億円
→フラッシュローンという技術を使いお金が盗まれた事件
ハッカーの気まぐれで返金されました。

ブロックチェーンは管理者がおらずロールバックできません。

金融系の不可逆なストレージです

7.『石橋を叩いて渡る』

固いコントラクトを作る



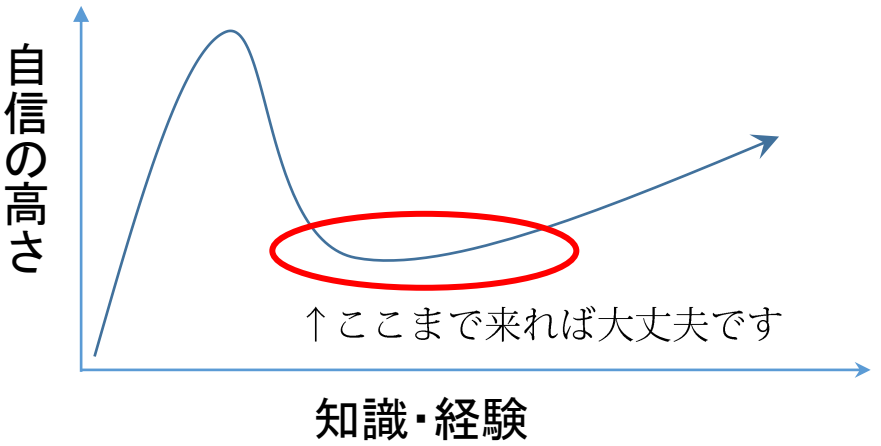
『石橋を叩いて渡る』を見ていきましょう
安全性には細心の注意が必要ですから。

特にチェックすべき観点

『後から変更できない』『全情報が公開されている』
『お金を盗もうとしてくる』『ロールバックができない』
この本で紹介した、上記の観点を思い出して下さい。

エンジニアの学習と自信

ダニング・クルーガー効果に戻りましょう。
『完全に理解した』を超えれば、事故は大幅に減らせます。
『完全に理解「しない」スマートコントラクト開発』です。



そもそも、すべてを理解するのは無理

どうすればいい？

ダブルチェックのすすめ

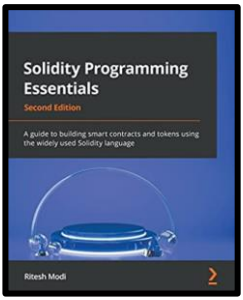
スマートコントラクトをエンジニア1名のみで作成するのは、
おすすめできません。どうしても盲点が出てしまうからです。

お金が沢山あれば監査会社に依頼すると、本格的にチェックしてもらえますが、スマートコントラクト開発者の友達がいたりすると、軽くダブルチェックしてもらうこと等も可能かと思います。スマートコントラクトのコミュニティに所属すると良いことがあるかもしれません。

さらに網羅的に学習するために

ここで紹介した以外にも、いろいろな攻撃手法があるため、網羅的な学習が必要です。

そのような学習を実施する場合のおすすめ書籍として
【Solidity Programming Essentials:
A guide to building smart contracts and tokens
using the widely used Solidity language, 2nd Edition】
を紹介します。



この書籍は、Solidityの基礎から始まり、スマートコントラクトやトークンの開発に必要な重要な概念を学ぶことができます。この書籍では、セキュリティの重要性についても詳しく説明されており、スマートコントラクトのセキュリティ上の問題を回避するためのベストプラクティスも載っています。

Solidity開発においては、外部レビューやChatGPTへの質問などにより、セキュリティ上の問題を減らしていく意識が必要でしょう。

『完全に理解した！！』のタイミングでメインネットにコントラクトをデプロイしないこと。

まず、落ち着いて。

『完全に理解しない』ことが大事です

8.『明日のことを知る者はいない』

未来はどうなるのでしょうか

近年、各国で人間とAIの対立が始まっているように思えます。
多くの国がAIを規制しようとしている様子です。
もちろん、AIには良い使い方、悪い使い方があるでしょう。
昔、チャップリンの時代に、機械を壊す運動があったように
これからの時代は人類がAIを止めようとするかもしれません。

ブロックチェーンはとても平等

ブロックチェーンは全ての内容を平等に扱ってきます。
チェーン上に構築された仕組みであるコントラクトは、
市民も、犯罪者も、国家も、AIも、すべてに対して平等に
扱ってくれます。
プログラムに書いた通りに動く。
誰もがズルをすることができないのです。
仕組みで回っていく。真に平等な世界です。
これが、今までのITシステムに無かった特性です。

プログラムを野生化させる技術

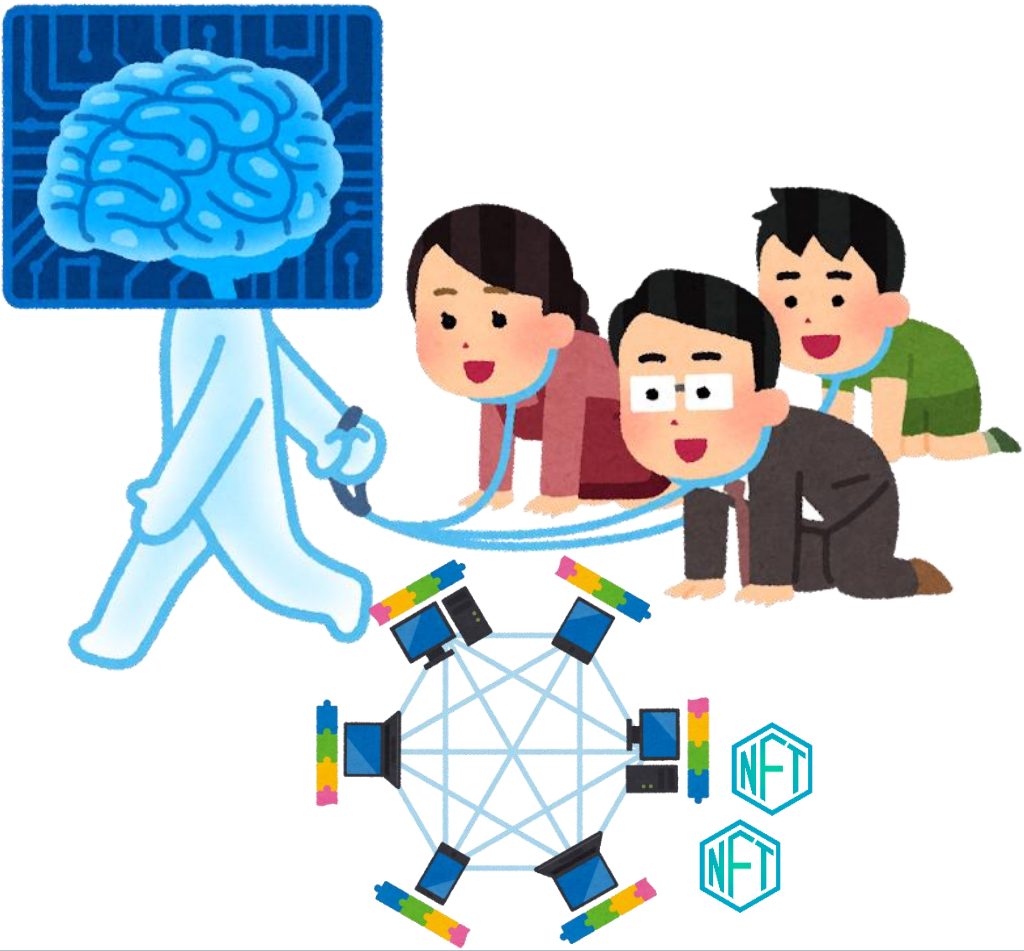
『プログラムが人の管理を離れて存在する』
これが、筆者から見たブロックチェーン技術です。
この技術の将来は分かりません。必要なら生き残るでしょう。
技術が切り拓く未来は、いろいろなパターンがありえます。

新しい技術は、新しい明日を作る

AIとBCと人類

たとえば、こんな未来(極端な例)

将来の世界はこのような図で表されるかもしれませんね。



未来は誰にもわからない

extra.猫の手も借りたい

人間とAIの役割分担の案 (2023年)

コントラクト開発時の役割分担、これでどうでしょう？

	エンジニア1名 の場合	エンジニア2名 の場合	エンジニア3名 以上の場合
分担	人：コーディング AI：レビュー	人：コーディング 人：レビュー AI：レビュー補助	人：コーディング 人：開発補佐 人：レビュー AI：レビュー補助
説明	コントラクト内容は人が理解しましょう	人によるレビューをAI補佐がベスト	大規模開発なら監査会社に頼るのもあり

AIによるレビューで使えるプロンプトの例

ChatGPT-4なら、次のようなプロンプトをレビューに使えます

プロンプト例	目的
このコントラクトに脆弱性があるとすれば、どのようなものですか CODE”” (ここにコードを貼る)””	人間の見落とし(学習不足も含む)をAIが指摘してくれます。AIの方が賢そうに見えたら、一度立ち止まるべきです。
このコントラクトのガス代や、運用上の課題やについて、どのような懸念点がありえますか CODE”” (ここにコードを貼る)””	コントラクトを実際に利用していく上での懸念点を教えてくれます。確認して大きなリスクがないか考えましょう。
このコントラクトに攻撃しようとする場合、どのような方法がありえますか CODE”” (ここにコードを貼る)””	マニアックな攻撃手法と、実現可能性が低いというコメントが出力されていれば、大きな問題はありません。参考程度に理解していくとよいでしょう。

活用していきましょう

まとめ

この本のまとめ

スマートコントラクト開発においては、「一応動く」と「セキュリティ上問題なく動く」の間に大きな壁が存在しています。

スマートコントラクトのセキュリティ上のリスクを十分に理解しないまま、Solidityの文法だけをなぞり、スマートコントラクトをデプロイしてしまうと、セキュリティ上のリスクが高まり、金銭的・社会的なトラブルが生じる可能性が高いです。

スマートコントラクトのセキュリティ上の問題に十分に注意し、テストやセキュリティ対策、セキュリティ監査などを行いながら、良い開発者ライフを送ってください。

2023年、AIは人間のサポートをする役割でしかありません。
自分が理解できていないコントラクトをデプロイするのはやめましょう。最後に責任を取るのは、あなたなのですから。

あとがき

ブロックチェーンが社会のパーツへ

ブロックチェーンがこれからどうなるか、よく考えます。
どうやら、次の時代を支える技術の一つになりそうです。
次の時代では、AIと人が共存する社会になるでしょう。
その中で『公平』と『公正』を実現する技術として、
ブロックチェーンがあると思います。

ブロックチェーンエンジニア

この本で紹介したように、ブロックチェーンで開発を
行うのは、なかなか大変です。覚悟を決めましょう。

なんでもトークンの紹介

なんでもトークンとは？

- ブロックチェーンに関する研究開発を行っています
- 筆者がトークンやDAppsの作成などを行っています。
- Getting TOKENS makes us happy.