

Zenlink Whitepaper v0.7

序章

一般に、Defi(分散型金融)には広い意味があり、大きく4つのタイプに分けられる:

1. KyberやUniswapのようなDEX。
2. CompoundやLendのようなローン系。
3. デリバティブや予測市場は、GNOやSNXに代表される。
4. オラクルなどの種類、代表はBAND、RENなど。

なかでも目を引くのがDEX。過去3年間(2017~2019年)の浮き沈みを経て、2020年に本格的な流行がやってきます。2017年、年間取引量が500万ドルに満たない分散型取引所は1つ(IDEX)しか存在しませんでした。

2018年、DEXの取引量は爆発的な成長を遂げ、取引量は27億ドルに達した。2019年、DEXの取引量は若干縮小しましたが、それでも25億ドルを上回りました。

2020年、DEXは急速な発展を遂げ、第1四半期の取引量(23億ドル)は2019年の通期取引量にほぼ匹敵しています。第2四半期の総取引高は過去最高の37億ドルに上昇しました。DEXは下半期もこの発展傾向を維持し、急速に発展することが期待されます。

注目のDefiやDEXに比べ、正式にリリースされた第3世代ブロックチェーン・プロジェクト、Polkadotもある。既存のブロックチェーンネットワークと比較して、Polkadotネットワークは、異種シャーディング、スケーラビリティ、アップグレード可能、透明なガバナンス、クロスチェーンコンポーザビリティなど、いくつかの明確な優位性を持っています。

異種のシャーディングネットワーク: Polkadotは基本的にシャーディングブロックチェーンですが、各シャードはパラチェーンであり、複数のチェーンをネットワークで接続し、基盤となるRelaychainが提供するセキュリティを共有しながら、トランザクションを並行して処理することを可能にすることを意味します。

拡張性: Polkadotは、複数の専用チェーンをシャード化したネットワークに橋渡しすることで、複数のトランザクションを並行して処理することを可能にします。ブロックチェーンネットワークのパフォーマンスボトルネックを解決します。将来的には、ネストされたリレーチェーンを通じて、ネットワーク内のパラチェーン(シャード)の数をさらに拡張することができます。

ノーフォークアップグレード: Polkadotは、ブロックチェーンをノーフォークでアップグレードすることを可能にします。これらの

アップグレードは、Polkadotの透明なオンチェーンガバナンスシステムによって実現されます。

オンチェーンガバナンス: すべてのDOTホルダーは、既存の提案に対して提案または投票することができます。

また、Polkadotのガバナンスシステムにおいて、受動的なステークホルダーを代表するボードメンバーの選出に協力することができます。

クロスチェーンの複合性: Polkadotのクロスチェーン互換性とメッセージパッシングにより、シャードは通信、価値交換、機能共有を行うことができ、既存のブロックチェーンネットワークや暗号化された資産と相互に作用することができます。

今後、DEXエコシステムのさらなる成長とパブリックブロックチェーン技術の急速な発展を予測し、パブリックブロックチェーン技術の急速な発展を踏まえ、Polkadotの高流動性、アップグレード可能、クロスチェーンDEXネットワーク「Zenlink」を提案します。

全体について

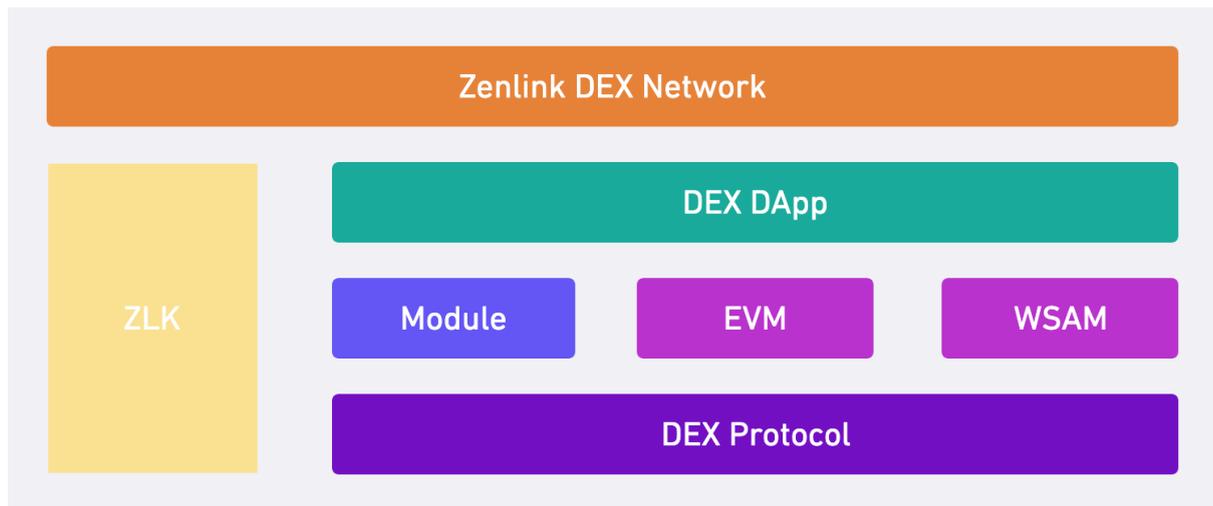
Zenlinkは、新世代のクロスチェーンDEXネットワークを構築することを約束します。Zenlink DEXモジュールを統合することで、Zenlinkはパラチェーンが迅速にDEX機能を所有し、他のパラチェーンと流動性を共有することを可能にします。

Polkadot上のすべてのDEX DAppsをリンクすることができます。ユーザーは簡単かつ迅速に交換を完了できるだけでなく、低スリッページの取引経験を楽しむことができます。ZenlinkのネイティブトークンZLKは、公正で透明なガバナンス機構と合理的な価値の取得方法を提供し、経済活動中のユーザーに長期的なネットワーク発展への参加を動機付けることができます。

ZenlinkはPolkadotをベースとしたクロスチェーンDEXネットワークです。一般に、Zenlink DEXネットワークは以下の部分から構成されます。

ZenlinkはPolkadotをベースとしたクロスチェーンDEXネットワークです。一般に、Zenlink DEXネットワークは以下の部分から構成されます。

1. Zenlink DEX プロトコル。トップレベルの統一された一般的なDEXプロトコルであり、以下の3つの実装を含みます。
 - a. Module: モジュール: Zenlink Protocol標準に準拠したSubstrate Runtime Module Library (SRML) 層上のモジュール。パラチェーンはこれと迅速に統合し、DEXの機能を得ることができ、さらに他のパラチェーン上のDEXと流動性を共有することができる。
 - b. EVMコントラクト: Ethereum EVMの運用と互換性を持たせるために採用された契約上の展開モードであり、Polkadotの初期段階において採用された移行スキームでもある。Zenlink DEX EVM版は、プロトコル層の全機能を実装し、完璧なテストプロセスを補完し、また、EVM契約のパラチェーンに初めて展開でき、プラットフォームの適用性を大幅に拡大する。
 - c. WASMコントラクト: Polkadot のオリジナルのコントラクト実装であり、将来的にPolkadotの主要なコントラクト展開モードでもあります。Zenlink DEX WASMバージョンは、WASMコントラクトプラットフォームに初めてデプロイする機能を持ち、プラットフォームの適用性が大きく向上しています。
2. Zenlink DEX DApp: Polkadot上のほとんどのDEXと接続することができ、ユーザーは低スリッページで複数のDEXとワンクリックで取引できる、シンプルでユーザーフレンドリーなDEX業界のアグリゲーターの入り口です。
3. Zenlinkトークン。Zenlink DEXプロトコルのネイティブトークンで、流動性メリットやガバナンスの分配などに使用されます。



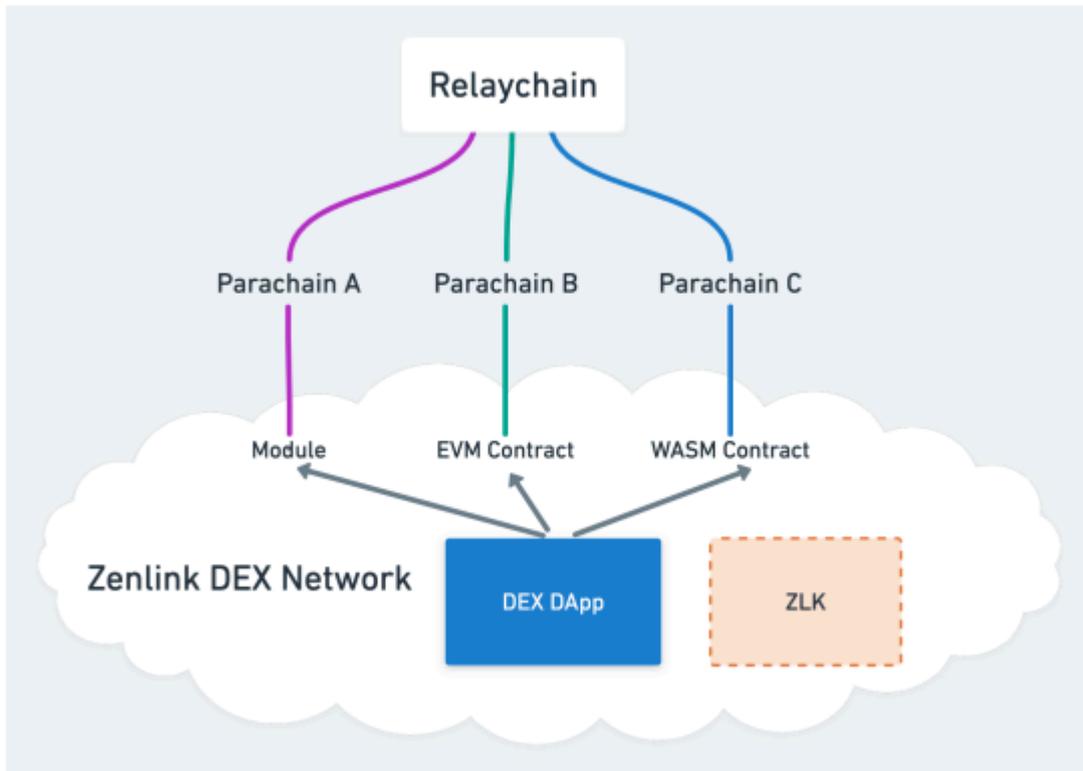
Zenlinkの計画概要

まず、Zenlink DEX Moduleは、Zenlink DEX Protocolをベースに実装される予定です。パラチェーンはこれと迅速に統合し、DEX機能を得ることができ、さらに他のパラチェーンのDEXと流動性を共有することができるようになります。他のパラチェーンにあるDEXと流動性を共有することもできます。続いて、モジュール、EVMコントラクト、WASMコントラクトを実装する予定です。

また、Zenlinkのエコシステムを完成させるために、フロントエンドのウェブサイトアプリケーションであるZenlink DEX Dappを構築したいと思います。Polkadot上のほとんどのDEXと接続するサンプルでスムーズなZenlink DEX DAppによって、ユーザーはワンクリックで複数のDEXと低スリッページでシームレスに取引できるようになります。

さらに、公平で透明なガバナンスの仕組みと合理的な価値獲得方法を提供し、エコシステムの利用者がネットワークの発展に長期的に参加する動機付けを行うため、Zenlinkは高性能なPolkadot上でネイティブトークンZLKを発行する予定です。

一般的に、初期段階では、取引プロトコルを定義し、異なる取引実装を実現することで、パラチェーン上に最初の取引所とアグリゲータアプリケーションを設定し、ガバナンスとインセンティブ機能の両方を考慮したネイティブトークンZLKを発行しました。以上の段階を経て、完全なZenlink DEXネットワークが形成されます。



中長期計画では、他のパラチェーンプロジェクトと連携し、より多くのパラチェーンにDEX実装を展開するとともに、Zenlink DEX DAppと他のPolkadotのDEXアプリケーションを接続し、より多くの取引ペアとより大きな流動性を提供する予定です。

Zenlink DEXプロトコル

Polkadotをベースとしたトップレベルの一般的な分散型取引プロトコルである。その特徴は

- ユニファイド・ユニバーサル・インターフェイス規格: モジュールはいつでも新しいものと交換できるので、システムのアップグレードやカスタマイズがいつでも可能という利点があります。また、アプリケーションシステムの拡張も容易で、より広いネットワーク環境に対応できます。言語に依存しない特性により、すべてのプログラマーがモジュールを書くことができ、この一連の標準規格を実現しさえすれば、Zenlink DEXネットワークにアクセスすることができるなど、完全に利用可能なものとなっています。
- クロス・チェーン・インターコネクション: プロトコル・インターフェース設計において、各パラチェーンの通信特性を集約し、相互運用可能な通信機構を形成する。Polkadot自体の相互運用性と連携し、パラチェーン間のメッセージ伝送をモジュール層で実現します。

アーキテクチャ

ZenLink DEX プロトコルでは、Assets、Dex.NET、DEX.NET など、いくつかのモジュールを定義しています。ここでは、ABC/DOT の取引ペアを例にとって、基本的な機能を説明します。

モジュール資産

Assets は ZenLink Dex プロトコルの基本モジュールです。ERC20のような外観をしています。ユーザーはこれを使って流動性やトークンを管理することができます。

Issue(発行)

ERC20トークンの新規発行

```
fn issue(origin, #[compact] total: T::TokenBalance, asset_info: AssetInfo)
```

パラメーター:

- 'total': 初期の総供給量
- 'asset_info': アセット情報には'name', 'symbol', 'decimals' が含まれます。

Transfer(転送)

トークンをオーナーから受け取り側に転送します。

```
fn transfer(origin, #[compact] id: T::AssetId, receiver: <T::Lookup as StaticLookup>::Source, #[compact] amount: T::TokenBalance)
```

パラメーター:

- `target`: アセットの受取手
- `amount`: 転送するアセットの量

Approve(承認)

トークンIDを持つ呼び出し元のトークンに対するspenderの許容amountを金額として設定します。操作が成功したかどうかを示す真偽値を返します。

```
fn approve(origin, #[compact] id: T::AssetId, spender: <T::Lookup as StaticLookup>::Source, #[compact] amount: T::TokenBalance)
```

パラメーター:

- `spender`: 支払い者のアカウント
- `amount`: 許容量

Transfer From(転送元)

トークンを送信者から受信者に許容機構を使用して移動します。操作に成功したかどうかを示す真偽値を返します。

```
fn transfer_from(origin, #[compact] id: T::AssetId, from: <T::Lookup as StaticLookup>::Source, target: <T::Lookup as StaticLookup>::Source, #[compact] amount: T::TokenBalance)
```

パラメーター:

- `id`: アセットID
- `from`: アセット送信者
- `target`: アセット受信者
- `amount`: 転送するアセット量

DEXモジュール

DEXはZenLink DEXプロトコルのコア・モジュールです。以下の機能を実装しています。

- トークン取引ペアの初期化
- トークンのスワップ
- 流動性の追加と抽出
- プロトコル全体で使用される流動性定数関数の定義

```
pub fn create_pair(  
    origin,  
    token_0: AssetId,  
    token_1: AssetId,  
)-> DispatchResult
```

パラメーター:

- origin: トレードアカウント
- token_0: アセットID
- token_1: アセットID

解説:

- Token_0とToken_1は、取引ペアを構成する2つのアセット(流動性プールとも呼ばれる)を表します。
- (token_0, token_1) と (token_1, token_0) 同じ取引ペアです

流動性追加

```
pub fn add_liquidity(  
    origin,  
    token_0: AssetId,  
    token_1: AssetId,  
    amount_0_desired : T::TokenBalance,  
    amount_1_desired : T::TokenBalance,  
    amount_0_min : T::TokenBalance,  
    amount_1_min : T::TokenBalance,  
    target_parachain: Paraid,  
    deadline: T::BlockNumber,  
)->DispatchResult
```

パラメーター:

- origin: トレードアカウント
- token_0: 取引ペアを構成するアセットID
- token_1: 取引ペアを構成するアセットID
- amount_0_desired: 流動性プールに預けたい token 0 の量
- amount_1_desired: 流動性プールに預けたい token 1 の量
- amount_0_min: 流動性プールに預ける token 0 の最低量
- amount_1_min: 流動性プールに預ける token 1 の最低量
- target_parachain: 流動性プールのパラチェーンID
- deadline: このトランザクションの締切ブロック

解説:

次のシナリオで考えてみましょう:

- アリスはABC(Parachain200のネイティブアセット)とXYZ(Parachain300のネイティブアセット)を持っています。
- アリスはABC/XYZ流動性プールに流動性を預けたいです。
- ABCはParachain300でABC'と表記されています。
- Parachain300に流動性プール(取引ペア ABC'/XYZ)があります。

例外事項:

- Parachain200で流動性追加取引を行う場合、Parachain300のアセット(XYZ)を十分に確保してください。そうでない場合、取引は失敗し、Parachain200のアセット(ABC)はABC'という形でParachain300に存在することになります。
- Parachain300で流動性追加取引を行う場合、Parachain300(ABC')にParachain200の資産を十分に確保する必要があります。ABC'がない場合、Parachain200に十分なABCがあったとしても、取引は失敗となります。

流動性の削除

```
pub fn remove_liquidity(  
    origin,  
    token_0: AssetId,  
    token_1: AssetId,  
    liquidity: T::TokenBalance,  
    amount_token_0_min : T::TokenBalance,  
    amount_token_1_min : T::TokenBalance,  
    to: <T::Lookup as StaticLookup>::Source,  
    deadline: T::BlockNumber,  
)->DispatchResult
```

パラメーター:

- origin: トレードアカウント Trading account.
- token_0: 取引ペアを構成するアセットID
- token_1: 取引ペアを構成するアセットID
- liquidity: 耐えられる流動性の高さ
- amount_token_0_min: 流動性削除後に得られると予想される token_0 の最低量。
- amount_token_1_min: 流動性削除後に得られると予想される token_1 の最低額。
- to: 受信するアカウント (Recipient account)
- deadline: このトランザクションの締切ブロック

token_0をtoken_1とスワップする

```
pub fn swap_exact_tokens_for_tokens(  
    origin,  
    amount_in: T::TokenBalance,  
    amount_out_min: T::TokenBalance,  
    path: Vec<AssetId>,  
    to: <T::Lookup as StaticLookup>::Source,  
    target_parachain: ParaId,  
    deadline: T::BlockNumber,  
) -> DispatchResult
```

パラメーター:

- origin: トレードアカウント
- amount_in: 送信したい正確な token 0 の量
- amount_out_min: あなたが期待する token 1 の最低量
- path: トランザクションパス
- to: 受け取りアドレス (Recipient address)
- target_parachain: 流動性プールのパラチェーンID
- deadline: このトランザクションの締切ブロック

解説:

- pathはアセットIDの配列として表現されます。
- 最初の要素は送信中のアセットを表し、最後の要素はターゲットアセットを表します。
- [A, B]: A/Bの流動性プールでAとBをスワップします。
- [B, A]: A/Bの流動性プールでBとAをスワップします。
- [A, B, C]: A/Bの流動性プールでAとBをスワップし、B/Cの流動性プールでBとCをスワップします。

さらに、

- pathの長さ制限はありません。
- APIサーバーからpathを取得します。現在プールは1つしかないため、パスの長さは2で固定されています。

token_0とtoken_1をスワップする

```
pub fn swap_exact_tokens_for_tokens(  
    origin,  
    amount_in: T::TokenBalance,  
    amount_out_min: T::TokenBalance,  
    path: Vec<AssetId>,  
    to: <T::Lookup as StaticLookup>::Source,  
    target_parachain: Paraid,  
    deadline: T::BlockNumber,  
)->DispatchResult
```

パラメーター:

- origin: トレードアカウント
- amount_out: 取得したい正確な token 1 の量
- amount_in_min: 送信する token 0 の最低量
- path: トランザクションパス
- to: 受け取りアドレス (Recipient address)
- target_parachain: 流動性プールのパラチェーンID
- deadline: このトランザクションの締切ブロック

解説:

- pathはアセットIDの配列として表現されます。
- 最初の要素は送信中のアセットを表し、最後の要素はターゲットアセットを表します。
- [A, B]: A/Bの流動性プールでAとBをスワップします。
- [B, A]: A/Bの流動性プールでBとAをスワップします。
- [A, B, C]: A/Bの流動性プールでAとBをスワップし、B/Cの流動性プールでBとCをスワップします。

さらに、

- pathの長さ制限はありません。
- APIサーバーからpathを取得します。現在プールは1つしかないため、パスの長さは2で固定されています。

パラチェーンにマッピングされたネイティブアセットを転送する

```
pub fn transfer(  
    origin,  
    asset_id: AssetId,  
    target: <T::Lookup as StaticLookup>::Source,  
    amount: T::TokenBalance  
)->DispatchResult
```

パラメーター:

- origin: トレードアカウント
- asset_id: アセットID
- target: 受け取りアドレス (Recipient address)
- amount: アセット量

解説:

- このインターフェースは他のパラチェーンによってマッピングされたアセットをチェーン内のアカウントにのみ転送することができます。
- 例えば、Parachain300のABC'をParachain200のアカウントではなく、Parachain300のアカウントに転送するなど。

マッピングされたネイティブアセットを他のパラチェーンに転送する

```
pub fn transfer_to_parachain(  
    origin,  
    asset_id: AssetId,  
    para_id: ParaId,  
    account: T::AccountId,  
    amount: T::TokenBalance  
) -> DispatchResult
```

パラメーター:

- origin: トレードアカウント
- asset_id: アセットID
- para_id: 対象のパラチェーンID
- account: 受け取りアドレス (Recipient address)
- amount: アセット量

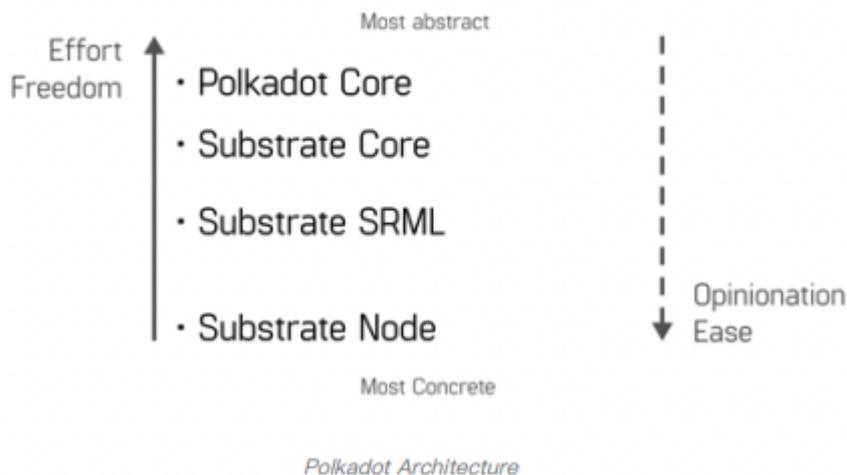
解説:

- このインターフェースは、マッピングされた/ネイティブなアセットを他のパラチェーンのアカウントに転送することができます。

Zenlink DEXモジュール

Polkadotは、層状で明確なアーキテクチャを巧みに採用し、開発者が独自のブロックチェーンを迅速に開発できるようにしています。

You choose!



Zenlinkプロトコル規格に準拠した一般的で安定したZenlink DEXモジュールをSRML (Substrate Runtime Module Library)レイヤに実装しました。

その特徴は:

- 挿して、繋いで、便利に使える: Polkadot上のパラチェーンは、Zenlink DEX Moduleを導入することでパラチェーン上にDEXを迅速に実装でき、より広いZenlink DEX Networkに接続することが可能です。パラチェーン上のトークンはZenlink DEX Networkに流動性を供給する一方で、他のパラチェーンやDOTとより自由な価値の流通を得ることができます。
- 柔軟かつ自由なアップグレードが可能: パラチェーンはZenlink DEX Moduleを交換するだけで、最新バージョンにアップグレードし、より強力な機能を体験することができます。

トレードの仕組み

トレードの仕組みは、一般的にOrderBookモデルとautomated market makerモデル(AMM)の2種類に分けられます。AMM自体は長い歴史を持っていますが、今回、ブロックチェーンのDefiアプリケーションによって、さらに身近な存在になりつつあります。今日のDefiの文脈では、AMMは「定型関数マーケットメーカー(CFMM)」であることがほとんどです。一般的な慣例に従って、当面はAMMという用語を使用しましょう。AMM(CFMM)は、Uniswap、Balancer、Curveなど、多くのDEXで採用されています。

その特徴は以下の通りです：

- トレーダーは、特定の取引相手とではなく、資産のプールと取引を行います。
- 流動性プールを維持し、安定した取引環境を提供するために、特定の数式が使用されます。
- マッチングシステムは必要なく、代わりに独自の数式に頼って決済を自動化します。
- 流動性プールを注入するユーザーをサポートします。

以上のことから、Zenlinkは当初、Polkadotエコシステムとコールドスタート流動性共有のための安定的でシンプルなトレードの仕組みを提供するためにConstant Function Market Maker(CFMM)モデルを使用することを検討しました。後期には、徐々に発展させながら、コンスタント・ミーン・マーケット・メーカー(CMMM)モデルへの転換を検討する予定です。Zenlinkは、リキッドマイニングのためにn次元の自動マーケットメーカーを提供する予定です。

ユーザーは最大n個のトークンを流動性プールに提供でき、各トークンの流動性プールにおける相対的なウェイトを設定でき、価格変動に応じてユーザーのポートフォリオを自動的にリバランスすることができます。

CFMM

まずはシンプルなCFMMモードでシステムを構築していきますが、最もシンプルなものは何もないません。

$$x * y = K$$

2つのトークンXとYを取引する分散型取引所を想定。トークンXとYの数をそれぞれxとyとする。そして、交換前と交換後を一定にする。

つまり、誰かが Δx トークンを売ると、その人は Δy トークンを手に入れることができます：

$$x * y = (x + \Delta x) * (y - \Delta y) = K$$

そして、 Δy は以下のようになり、

$$\Delta y = y - \frac{K}{x + \Delta x}$$

価格pは以下になります。

$$p = \frac{\Delta x}{\Delta y} = \frac{\Delta x(x+\Delta x)}{y(x+\Delta x)-K}$$

したがって、ユーザーは売りたいトークンXの量を提示するだけでよく、モジュール内で自動的にプログラムを計算することで、価格pを提示し、理想的に得ることができるのです。

流動性プール

金融市場の創造と発展には、流動性が不可欠です。AMM DEX は通常、独自の流動性プールを有しています。流動性プールは、基本的にZenlinkトレーディングモジュールまたはスマートコントラクトに設定されたトークンのプールです。Zenlinkトレーディングモジュール内の流動性プールは、以下のような特徴を持ちます：

- ユーザーは自由に流動性プールを作ることができます。つまり、Zenlink DEX Networkに取引ペアを自由に追加することができます。
- 各取引ペアの流動性プールは各パラチェーンの構築に依存し、それ自体は互いに独立したものです。
- 複数の流動性プールを含む取引はZenlink DEXアグリゲーターによってマッチングさせることができます。
- 流動性プールはスマートコントラクトによって人手を介さずに維持されます。

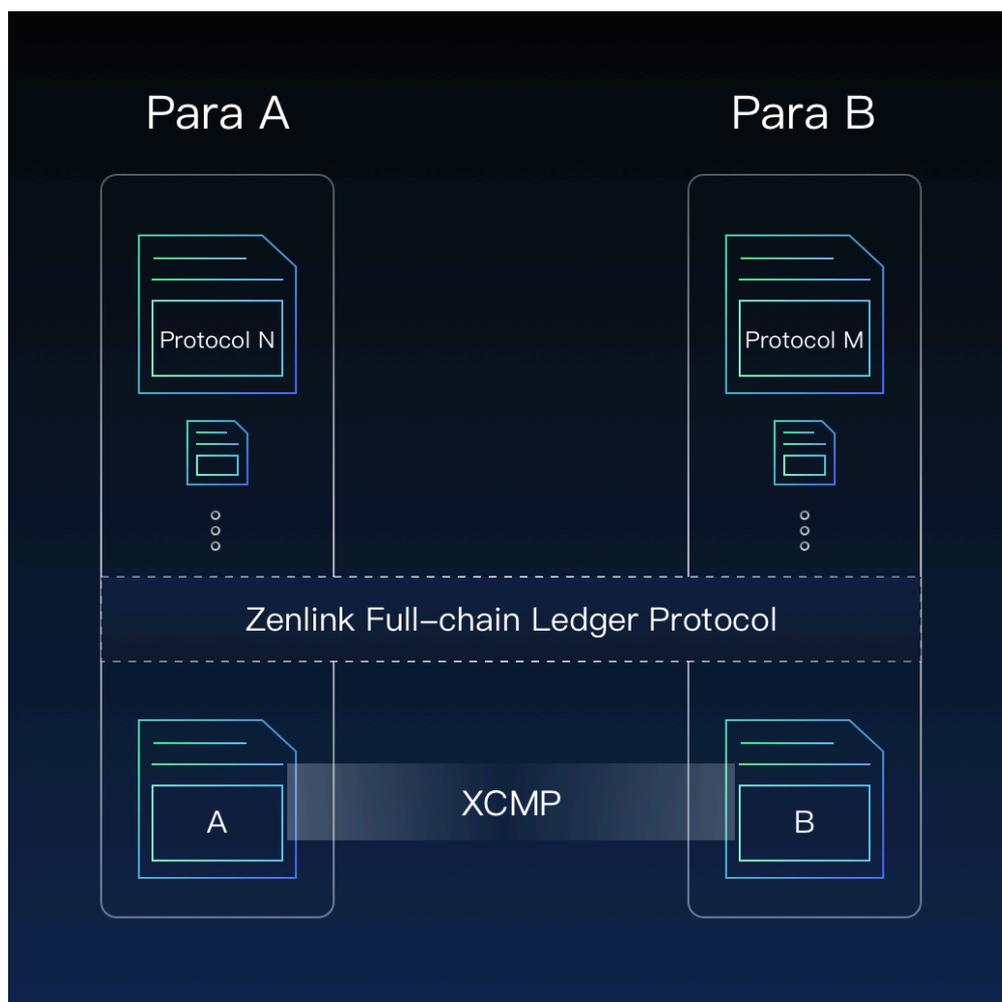
Zenlink DEXモジュールの技術ソリューション

Polkadot上のパラチェーンは、基本的に完全に独立した自由にプログラム可能なランタイムモジュールの集合体です。イーサリアムのスマートコントラクトと比較すると、パラチェーンはより完全に分離されており、パラチェーン間の実行計算を独立して並列化することができます。これにより、Polkadotの総合性能は数段向上しますが、パラチェーン間の相互作用はより複雑化することになります。

Gavin氏を中心としたチームでは、パラチェーン上の通信プロトコルXCMPの開発を精力的に進めています。XCMPの開発が完了しても、XCMPプロトコル上でアプリケーション関連の各種一般プロトコルを開発する必要があります。Zenlink DEXプロトコルは、パラチェーンが相互運用し、流動性プールを共有するためのプロトコルを実装するものです。

Polkadotのクロスチェーンプロトコルは、従来のTCP/IPプロトコルと比較することができます。XCMPプロトコルは、TCP/IPのデータリンク層と似ており、各チェーン(ルート)間の無差別知覚プロトコルの伝送機能を解決するものです。

Zenlink DEXプロトコルは、TCP/IPのアプリケーション層と似ています。各パラチェーンはZenlink DEXプロトコルを実装している限り、Polkadot上のパラチェーンの流動性プールをすべて共有することができます。



Zenlink DEXの機能は、以下の3つの方法でパラチェーンに取り込むことができます。

1. Substrateモジュールとしてパラチェーンに統合
2. パラチェーン上のWASMコントラクトモジュールに展開
3. パラチェーン上のEVMコントラクトモジュールに展開



Zenlinkは、Zenlink DEX Protocolをパラチェーンにワンクリックで統合するために、上記の3つの方法を具体的に実装しています。

Zenlink DEX DApp

長い時間をかけて、分散型取引所 (DEX) と集中型取引所 (CEX) が共存し、補完し合っていくことが見て取れます。そして、Polkadotのネットワークを見れば、今後、様々なタイプの分散型取引所が存在することは間違いないでしょう。Zenlink DEX DAppは、これらの異なる取引所への抽象的な統一アグリゲータのエントリーポイントです。

以下のような機能を有しています：

- 分散型取引所の様々なインターフェースプロトコルに対応。
- アグリゲーターは、既存の取引ペアの複数の取引所にわたる価格を自動的にマッチングし、スリッページポイントが最も低い取引経路を提供し、最終的に複数の取引所にわたる取引をマッチングします。
- まだサポートされていない取引ペアについては、アグリゲーターが複数の取引所間でパス検索を行い、取引を確定させます。
- エンドユーザー向けにシンプルで統一されたアプリケーション・インターフェースを提供する。ユーザーは、その背後にあるロジックを気にすることなく、アプリケーション内でワンクリックで交換することができます。

トークノミクス

Zenlinkの技術アーキテクチャとエコシステムの計画に基づいて、ネイティブトークンZLKの主なアプリケーションシナリオは次のとおりです。

流動性マイニング

ネットワークに流動性を提供するユーザーやプールに対しては、流動性提供者の資産額と預け入れ期間に応じて、対応するZLKトークンを非線形関数で放出することになります。金額が大きく、流動性の期間が長ければ長いほど、さらなる優遇措置があります。「coins per day」というコンセプトが導入される予定です。

取引ネットワークのオンチェーンガバナンス

ZLKトークンは、トークン上場、流動性アクセス、他のDEX枠へのアクセス、プロトコルのアップグレードなど、プロトコル全体が構築する取引ネットワークに深く関わっていくこととなります。

ネットワーク収益の獲得

Zenlink DEX ネットワークは、分散型のトレーディング・ネットワーク・エコシステムです。Zenlink DEXプロトコルをトップレベルプロトコルとして、各パラチェーンや他の取引所アプリケーションのZenlink DEXモジュールで低レベル取引ノードを構成し、すべての取引ノードはZenlink DEZ DAppでリンクされて、豊富な取引ペアと強い流動性を提供するように指示されている。Zenlink Token (ZLK) は秩序ある発展とコミュニティガバナンスの目標を達成するために使用されます。

PolkadotとDeFiの発展により、Zenlink DEXネットワークはさらにレンディングサービス、オラクル、金融派生商品などの種類を導入し、進化と成長という目標を達成する予定です。

ロードマップ

マイルストーン1: 2020 Q4

- Zenlink DEX プロトタイプを基板テストネットチェーン上に構築します。
- テスト用のZenlink DEX Dappをビルドします。

マイルストーン2: 2021 Q1

- Zenlink DEX Smart Route 機能を実装。
- Zenlink DEX DAppのパブリックβテスト1回目を開始。
- WASMコントラクトの実装

マイルストーン3: 2021 Q2

- EVMコントラクトを実装する。
- Zenlink DEX DAppパブリックベータテスト第2回を開始。
- Zenlink DEXプロトコルをKusama parachainの一部にデプロイ。
- KusamaでZenlink DEX DAppをローンチ。
- Zenlinkトークンの発行

長期計画

- 2021 Q3、より多くのパラチェーンと協力し、Zenlink DEXプロトコルを統合し、Zenlink DEXネットワークにより多くの取引ペアと流動性を提供します。
- 2021 Q3、PolkadotのいくつかのパラチェーンにZenlink DEXプロトコルを導入し、Polkadot上でZenlink DEX DAppをローンチする予定です。
- 2021 Q4、Zenlink DEX Aggregatorがより多くのPolkadot DEXアプリケーションにアクセスする。

要約

私たちが将来的に想定しているのは、Polkadotの上にどんどんプロジェクトが構築されていく、そんなシナリオです。様々なビジネスを展開するパラチェーンは、互いに交流し、コミュニケーションする必要があります。デジタル資産の巨大な流動性は、デジタル世界全体における重要なリンクとなります。Zenlinkは、これらの結合を支える重要なサポートとなることを約束し、ネットワーク全体の価値が自由に流れることを可能にします。

Version History

1. 2020.08.17, v0.1 created.
2. 2020.08.30, v0.2 updated.
3. 2020.09.13, v0.3 updated.
4. 2020.09.21, v0.4 updated.
5. 2020.11.03, v0.5 updated.
6. 2020.12.01, v0.6 updated.
7. 2021.06.15, v0.7 updated.

本文書はチーム発行の公文書を日本語へ翻訳したものとなりますが、チームが発行する公文書ではありませんのでご了承ください。いかなる場合も下記リンク先のチーム公式文書の内容を正として取り扱います。 <https://zenlink.pro/en/whitePaper/>

以上