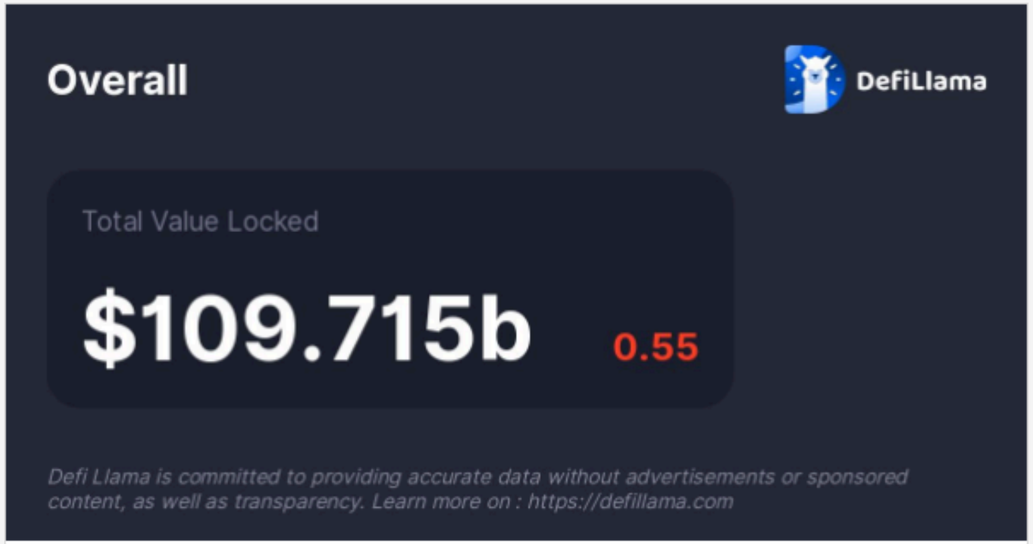
Introduction to DeFi

11 / 8

More on Oracle

More on DeFi

Let’s explore the DeFi space



DefiLlama

DefiLlama is a DeFi TVL aggregator. It is committed to providing accurate data without ads or sponsored content, as well as transparency.

DefiLlama

DeFiLlama

Dune — Crypto Analytics Powered by Community.

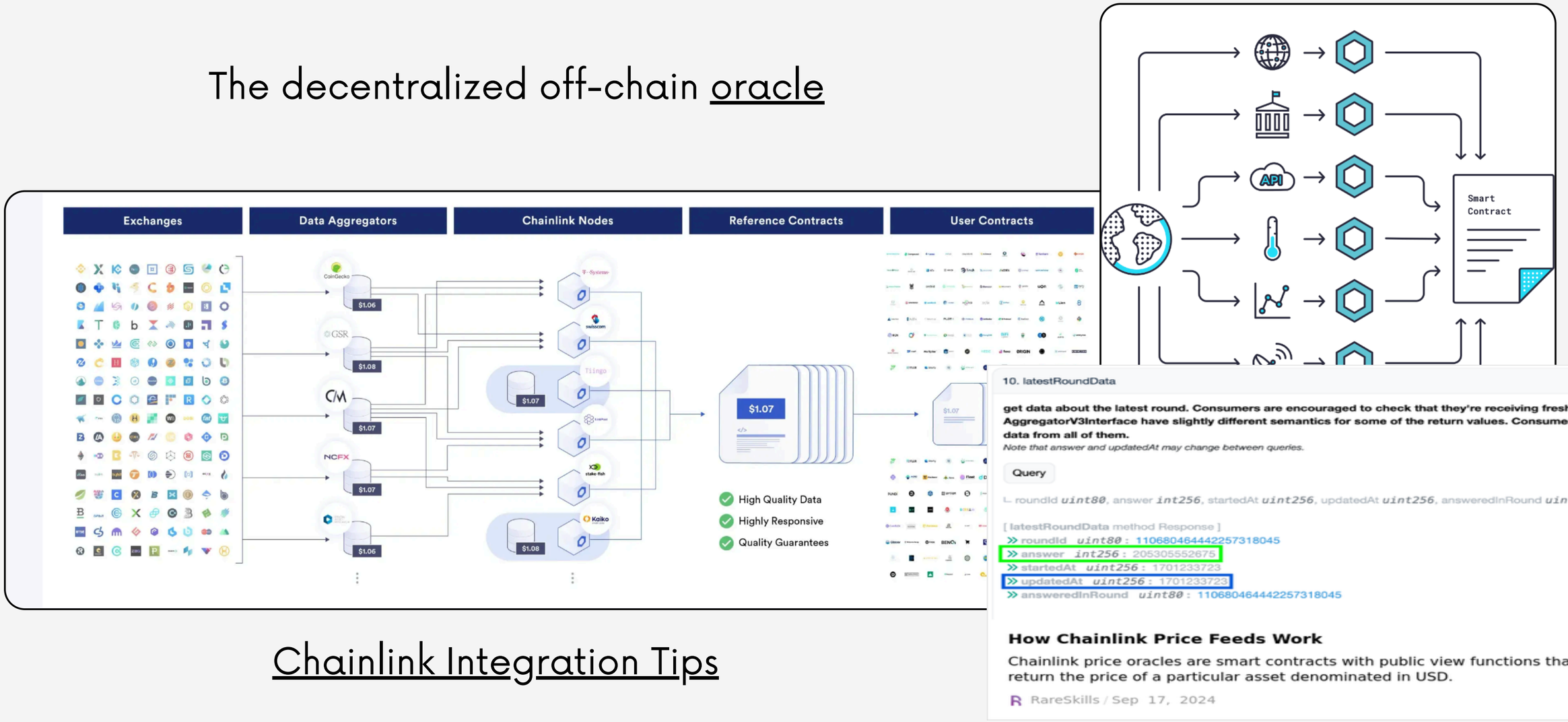
Connect, query and explore blockchain data with the world’s top analysts. Explore 100,000+ community dashboards and projects. Follow metrics for DeFi and NFTs.

dune.com

Dune

Chainlink

The decentralized off-chain oracle



Chainlink Integration Tips

Pyth

Another oracle provider

Price Feed Oracles for DeFi Apps

Explore Pyth's real-time oracle data for blockchain applications. Pyth delivers accurate, reliable market data to make smart contracts smarter and more secure.

Price Feed Oracles for DeFi Apps

What's the Difference Between Pyth and Legacy Oracles | Pyth Network

Looking for a better alternative to your smart contract oracle provider? Learn why you should use Pyth's low-latency pull oracles for your DeFi needs.

pyth.network

Oracle Manipulation

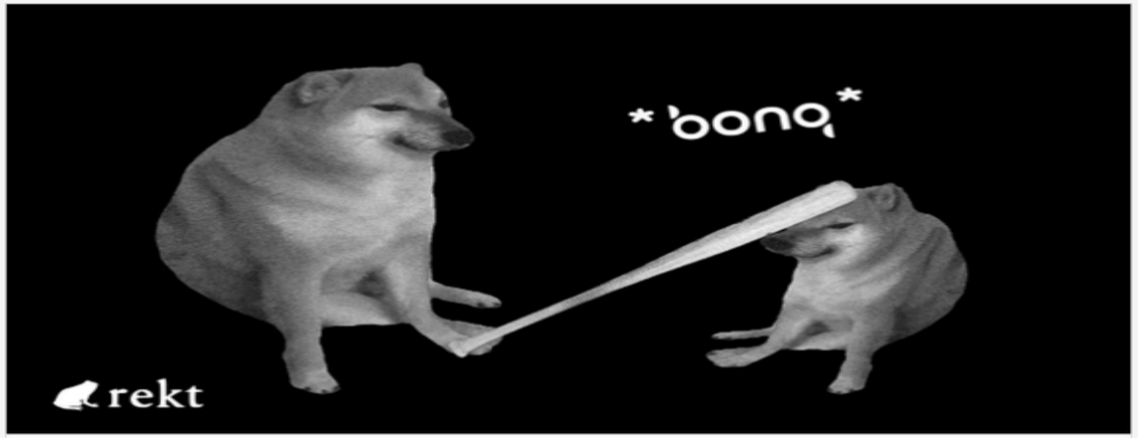
Some case study

Understanding the UwU Lend Exploit

Learn how UwU Lend was exploited, which resulted in a loss of assets worth $23 million.

Neptune Mutual / Nov 23, 2024

UwU Lend



BonqDAO

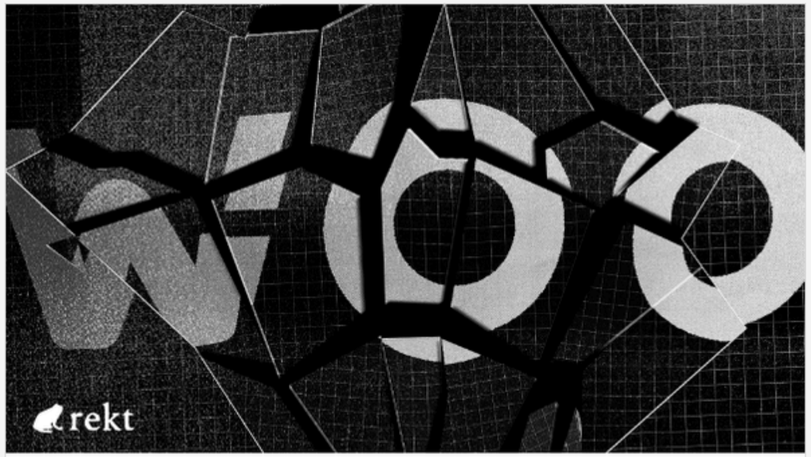
DeFi / Crypto - BonqDAO got bonked for $120M, but the anonymous attacker got away with less than $2M. The hacker was able to manually update the price feed of collateral by staking just $175 worth of TRB...

rekt

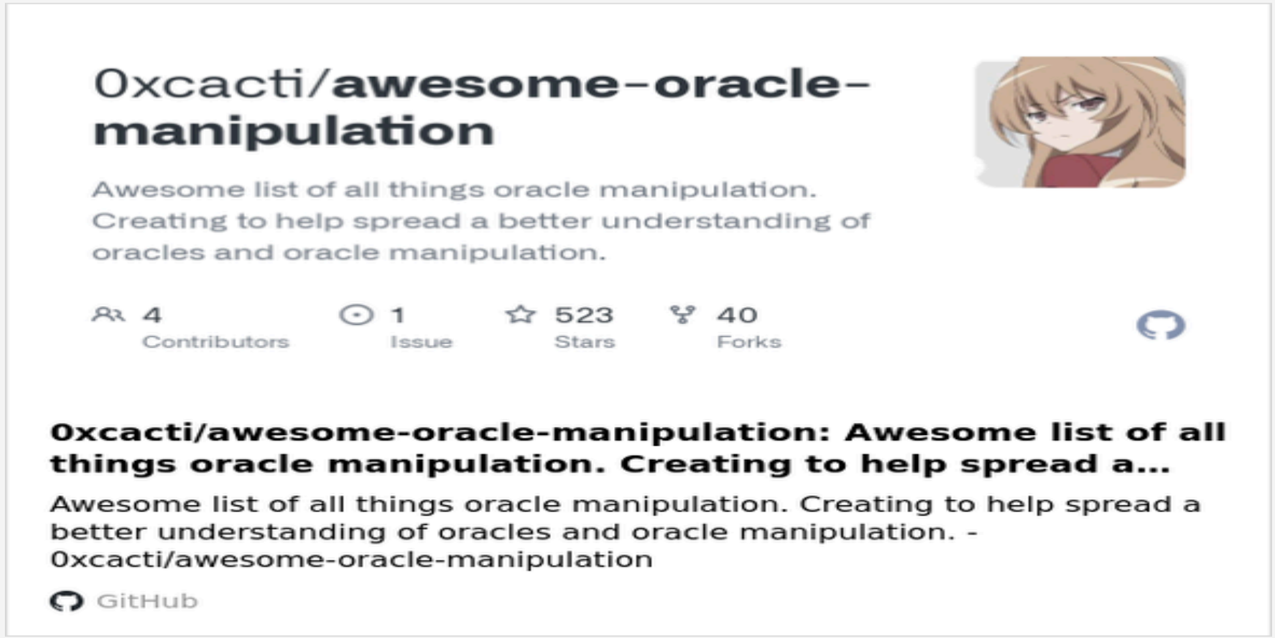
BonqDAO

Oracle Manipulation

Some case study



WooFi



Oracle Manipulation

Oracle Manipulation

- TWAP Oracle Attacks: Easier Done than Said?
- Safeguarding DeFi Smart Contracts against Oracle Deviations

Flash Loan

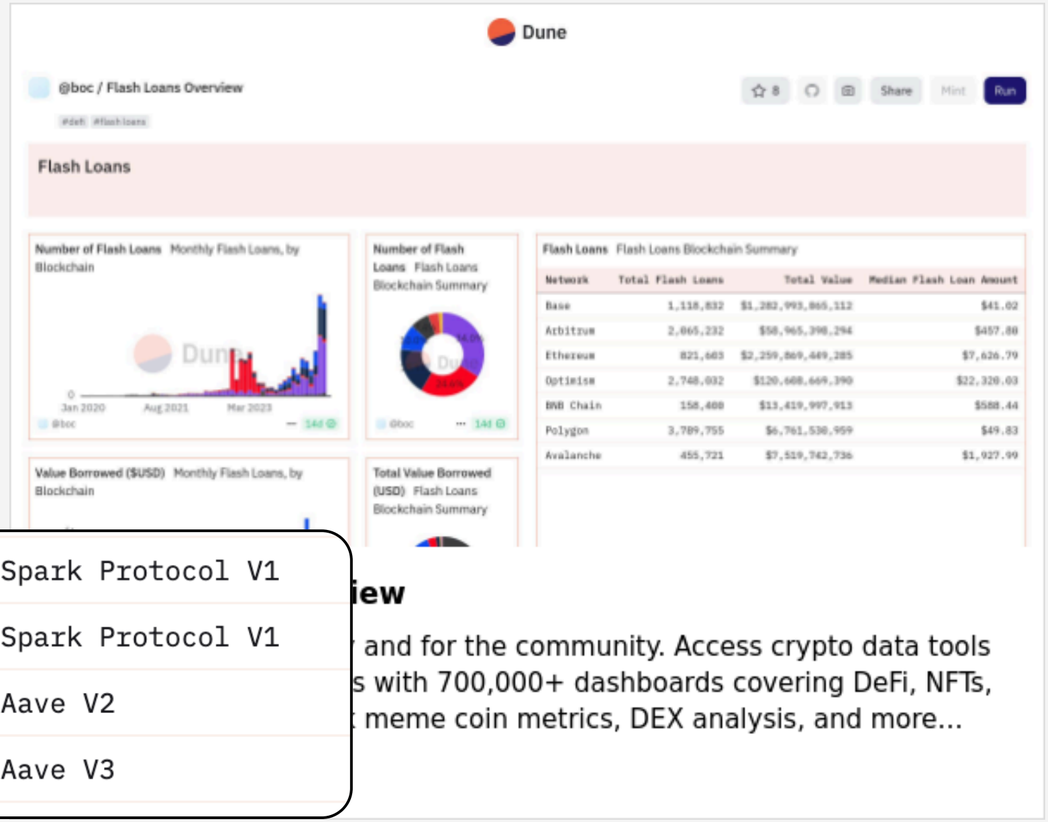
Definition

- Uncollateralized loans that allow users to borrow assets without any collateral, the borrowed amount should be repaid within the same transaction block.
- if the loan is not repaid within atomic transaction, the transaction is reverted, effectively making it as if the loan never occurred

Flash Loan

Limitation

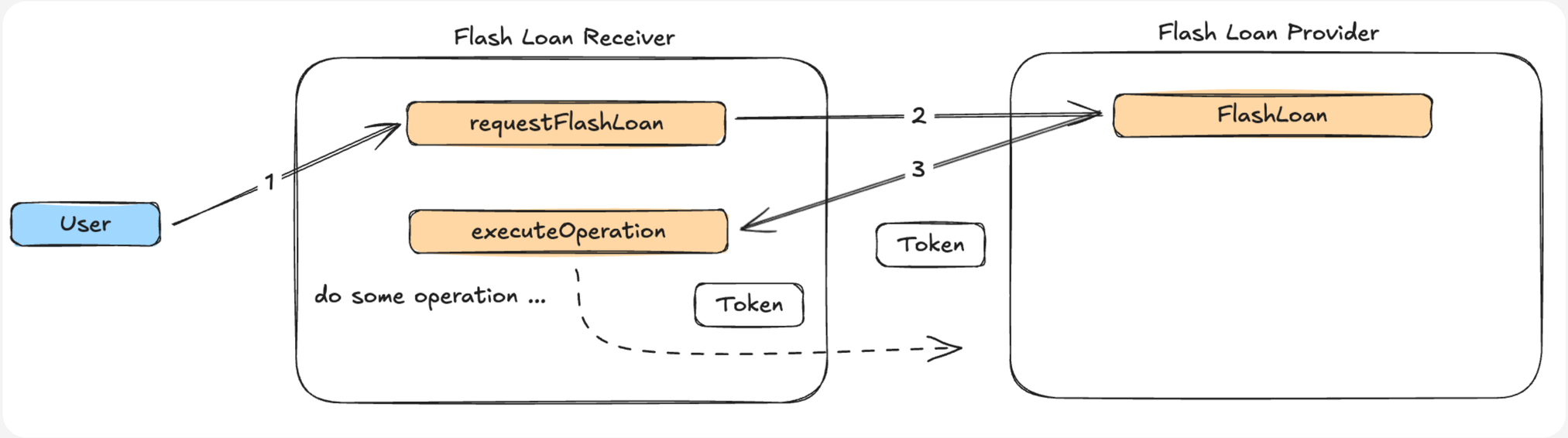
- The flash loan should be repaid within the same transaction
- Only contract account can request a flash loan, EOA cannot



2024-07-18 13:53	$612,466,577	$130	wstETH	Spark Protocol V1
2024-07-15 20:13	$595,478,052	$173	wstETH	Spark Protocol V1
2022-06-16 08:47	$587,766,780	$112	WBTC	Aave V2
2024-06-10 12:06	$585,205,956	$10,037	WETH	Aave V3

Flash Loan

Pattern



ERC-3156: Flash Loans

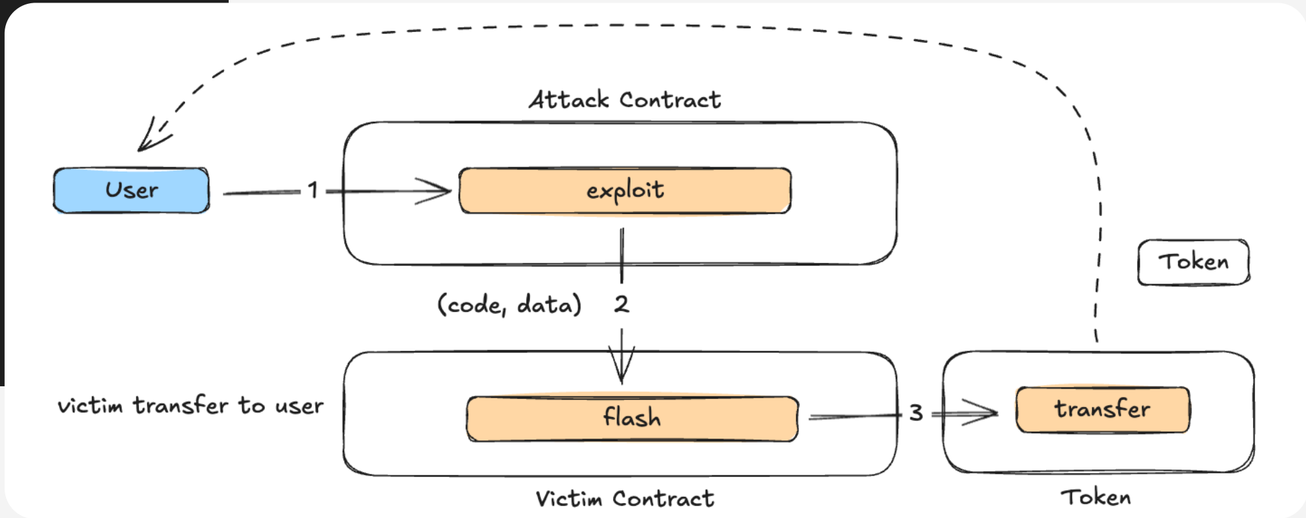
This ERC provides standard interfaces and processes for single-asset flash loans.

Ethereum Improvement Proposals

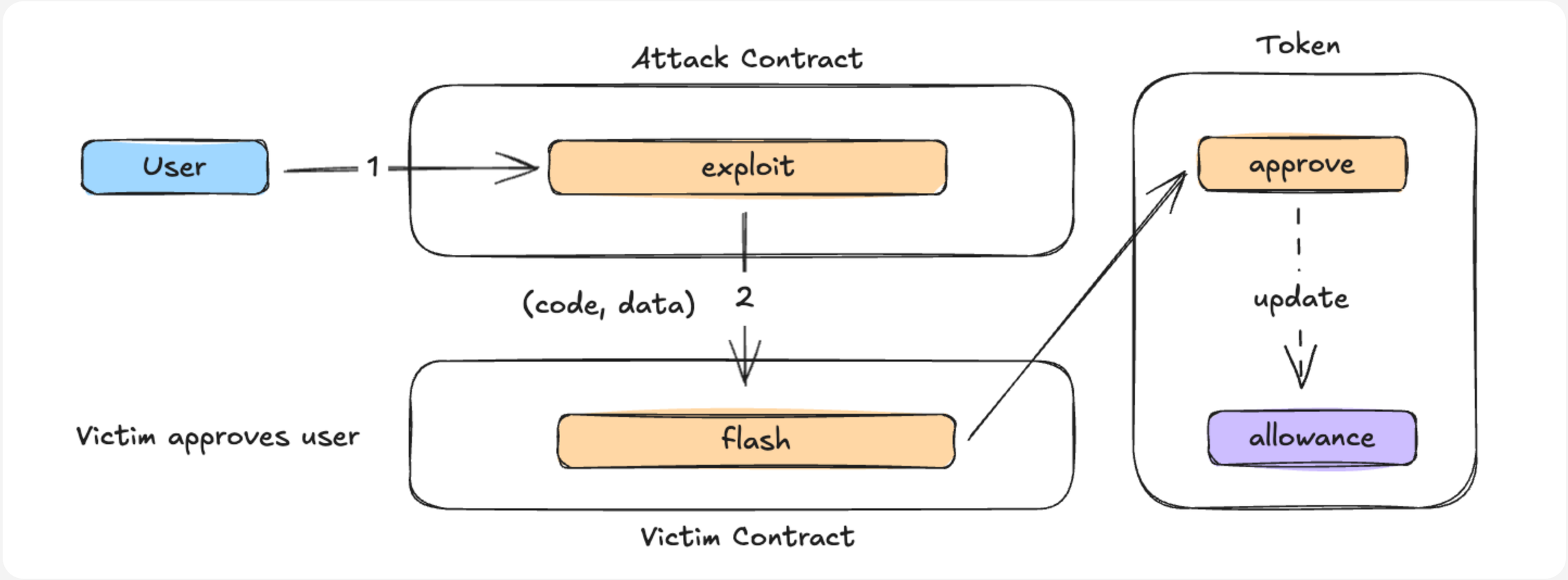
Arbitrary Call

```
// flash borrow
// locked with itself to avoid flashing more than MINT
function flash(address code, bytes calldata data)
  external payable returns (bytes memory result) {
    // lock->mint->call->burn->unlock
    VatStorage storage vs = getVatStorage();
    if (vs.flock == LOCKED) revert ErrLock();
    vs.flock = LOCKED;

    getBankStorage().rico.mint(code, _MINT);
    bool ok;
    (ok, result) = code.call(data);
    if (!ok) bubble(result);
    getBankStorage().rico.burn(code, _MINT);

    vs.flock = UNLOCKED;
}
```

The attacker provides a target address and call data to the victim contract to execute malicious actions.



Arbitrary Call



Flash Loan

Application: Arbitrage

Pool A ETH Price: 2800 USD

Pool B ETH Price: 2900 USD

- Purchase ETH from a pool with a lower price and sell it in a pool with a higher price to make a profit from the price difference.

Flash Loan

Application: Loan Refinancing

- On-chain lending: To borrow 1000 USDC, you need to deposit more than 1200 USDC - Overcollateralization
- You deposit S1200 worth of ETH into lending protocol A and borrow 1000 USDC.
- The borrowing interest rate at protocol A is 5%, but you find a lower rate of 4% at lending protocol B.
- You want to migrate your loan to the new protocol. How can you achieve this?

Lending Overview

Undercollateralization vs Overcollateralization vs Uncollateralization

- Flash loans are uncollateralized, while standard lending protocols require overcollateralization.
- Collateral Factor (Loan-to-value): the percentage that determines how much a user can borrow against the value of their supplied collateral.
- If an asset has a collateral factor of 75%, users can borrow up to 75% of the value of that asset as a loan.
- For example, if you provide $1,000 worth of ETH as collateral, you can borrow up to $750, calculated as $1,000 × 0.75.

Lending Overview

Undercollateralization vs Overcollateralization vs Uncollateralization

- Flash loans are uncollateralized, while standard lending protocols require overcollateralization.
- Long on ETH: Deposit ETH as collateral to borrow USDC, then use the borrowed USDC to buy back ETH
- Short on ETH: Deposit USDC as collateral to borrow ETH, then use the borrowed ETH to purchase USDC.

Lending Overview

Undercollateralization vs Overcollateralization vs Uncollateralization

- The maximum amount of assets that can be borrowed in this way is:

$$1 / (1 - LTV)$$

- If the LTV is 0.8, you can borrow up to 5 times the value of your collateral.
- How can you maximize the number of tokens you hold?

Lending Overview

Undercollateralization vs Overcollateralization vs Uncollateralization

- The maximum amount of assets that can be borrowed in this way is:

$$1 / (1 - LTV)$$

- If the LTV is 0.8, you can borrow up to 5 times the value of your collateral.
- How can you maximize the number of tokens you hold? Short on ETH
  - Deposit USDC, borrow ETH, swap the borrowed ETH for USDC, redeposit the USDC, and borrow ETH again.
  - Repeat this process again and again

Lending Overview

Undercollateralization vs Overcollateralization vs Uncollateralization

- If you have $1000 ETH, and the collateral factor is 0.8, you can borrow maximum $5000 USDC
- Use a flash loan to borrow $5,000 worth of stablecoins.
- Swap the $5,000 in stablecoins for an equivalent amount of ETH.
- Deposit the ETH into a lending pool as collateral.
- Borrow $4,000 worth of stablecoins against the ETH collateral.
- Combine $1,000 of your own stablecoins with the $4,000 borrowed from the lending pool to repay the flash loan.

Flash Loan

Application: Loan Refinancing

- The typical process involves:
  - Repaying 1000 USDC to lending protocol A to reclaim your 1200 USD worth of ETH collateral.
  - Depositing the 1200 USD worth of ETH as collateral into lending protocol B to borrow 1000 USDC.
- However, if your borrowed assets have been deposited into other DeFi applications. What do you do then?

Flash Loan

Application: Collateral Swap

- You deposit $1200 worth of ETH into lending protocol A and borrow 1000 USDC. But what if a better borrowing interest rate is available for wBTC as collateral?
- How can you switch your collateral to wBTC?

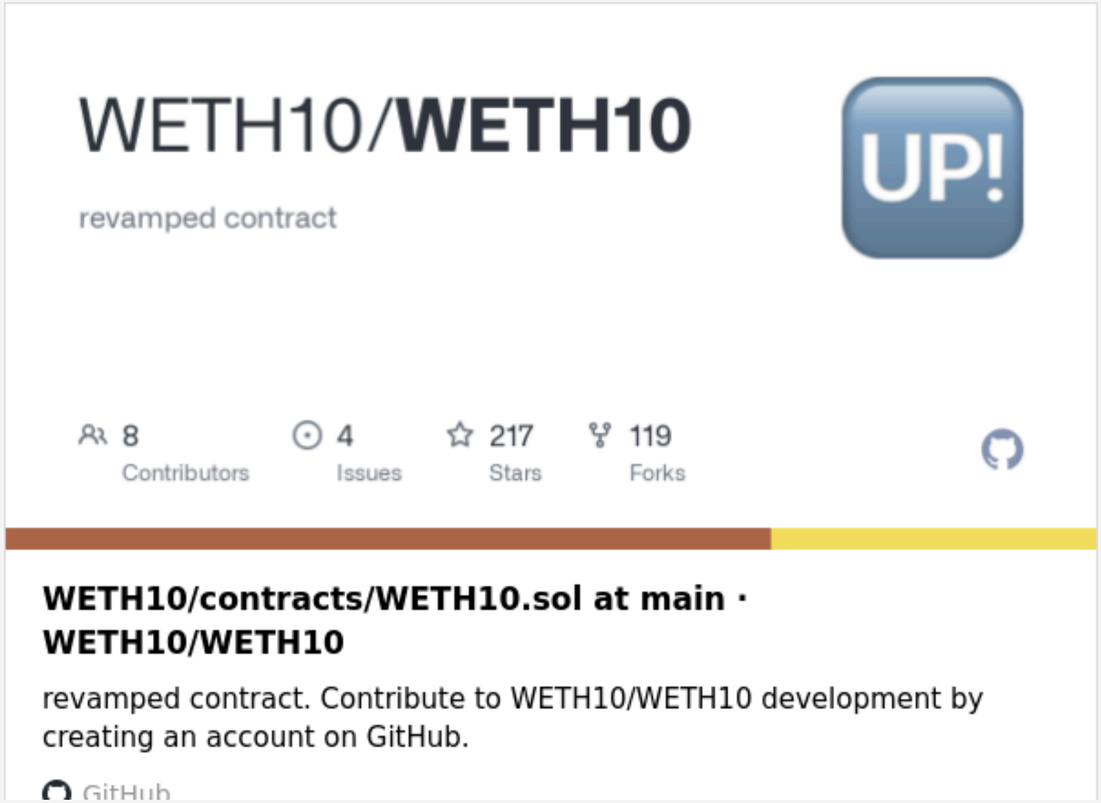
Flash Loan

What types of DeFi application would provide flash loan as service?

- Uniswap generates income for depositors through trading fees.
- Their large capital reserves enable them to offer flash loans, creating an additional revenue stream.
- This maximizes capital efficiency by allowing the same funds to be used for multiple purposes.

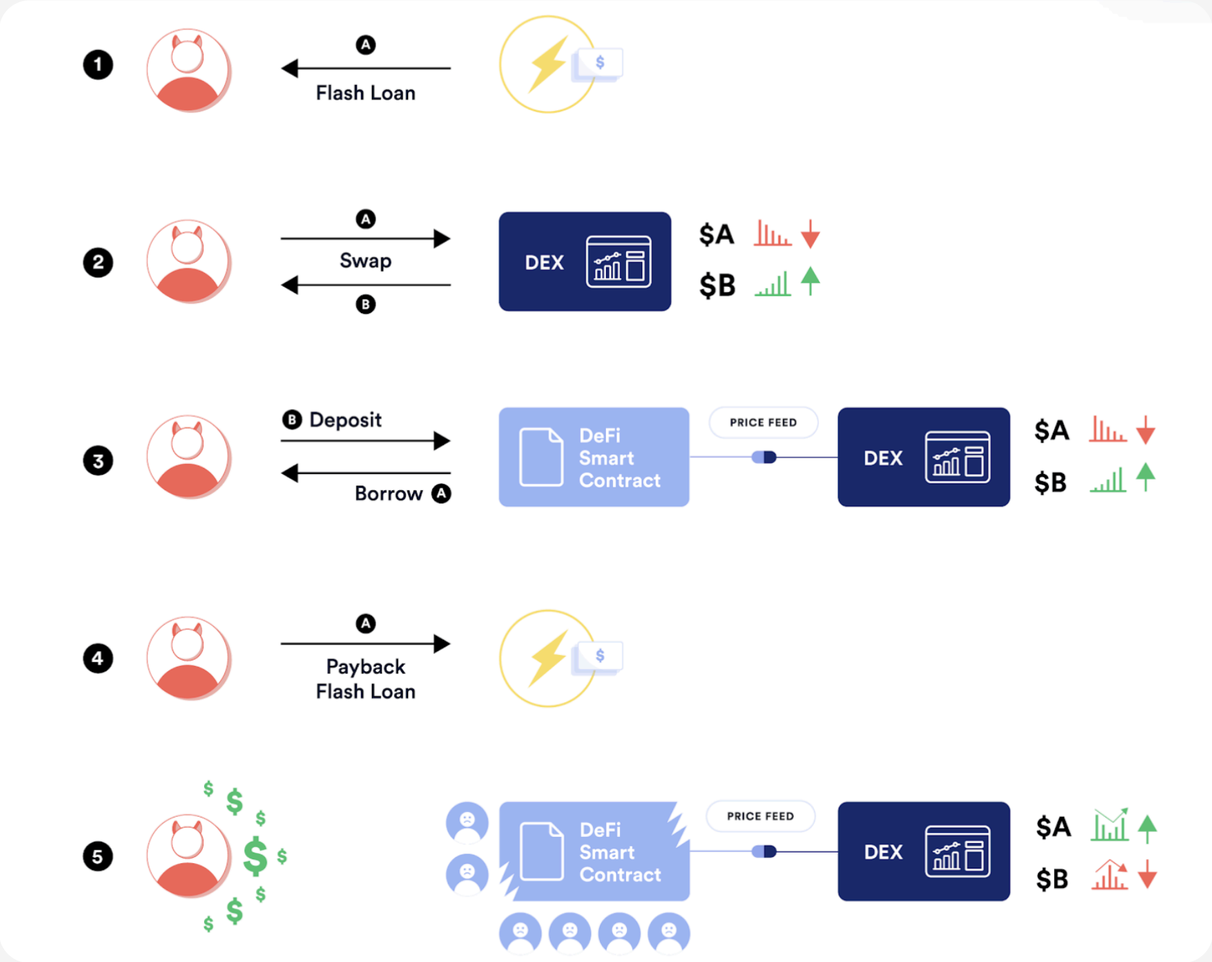
Some Invariant

Service similar to flash loan



- Flash Mint
- Flash Borrow
- Flash Swap

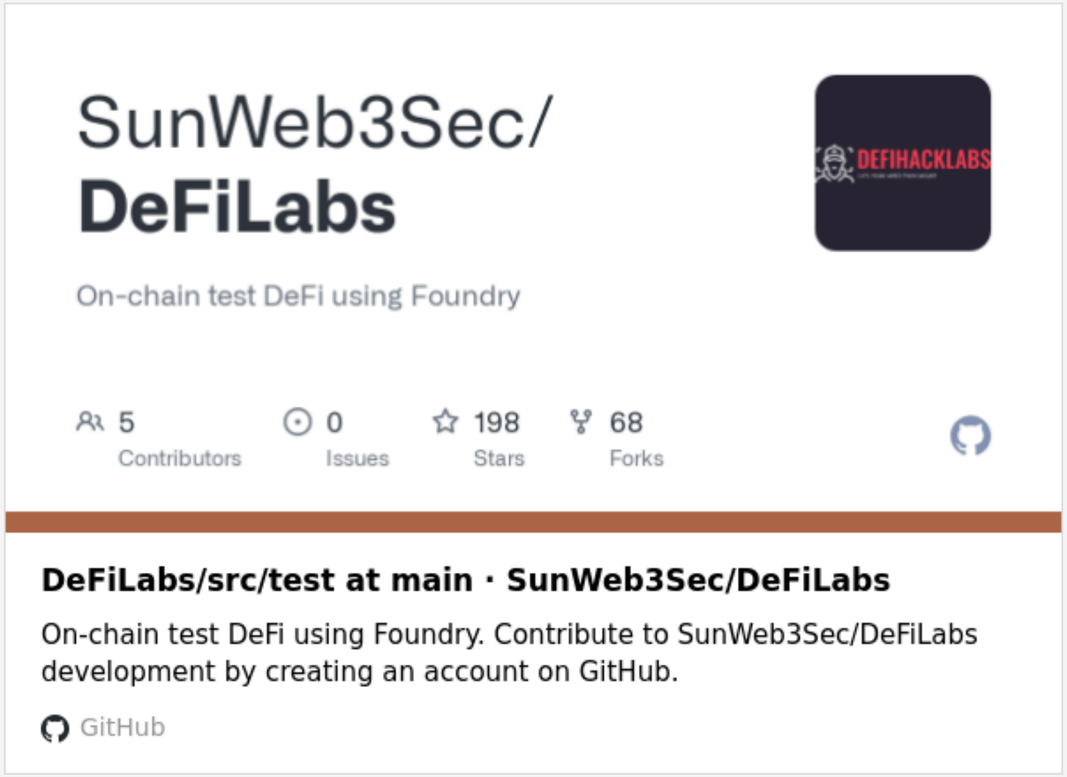
Oracle Manipulation



Flash loan can affect the spot price

Flash Loan Provider

Often, DEX, lending protocol will provide flash loan service



- Aave
- Balancer
- DoDo Finance
- Uniswap

Decentralized Exchange

How can you swap token on chain?

COUNT	AMOUNT	TOTAL	PRICE	PRICE	TOTAL	AMOUNT	COUNT
1	0.9873	0.9873	21,679	21,682	0.4697	0.4697	1
1	1.101	2.088	21,678	21,684	1.176	0.7071	2
1	0.0232	2.111	21,677	21,685	1.955	0.7784	2
2	0.9011	3.012	21,676	21,686	2.070	0.1152	1
3	1.301	4.313	21,674	21,687	2.301	0.2305	1
5	1.543	5.857	21,673	21,688	2.762	0.4610	1
3	0.8241	6.681	21,672	21,689	5.430	2.668	3
4	0.6837	7.365	21,671	21,690	6.591	1.161	3
4	1.508	8.873	21,670	21,691	7.355	0.7642	3
3	0.6397	9.512	21,669	21,692	7.896	0.5404	3
3	0.6518	10.16	21,668	21,693	7.993	0.0976	1
3	2.451	12.62	21,667	21,694	10.60	2.610	3
3	1.062	13.68	21,666	21,695	11.53	0.9287	3
4	9.307	22.99	21,665	21,696	14.72	3.190	3
3	0.6663	23.65	21,664	21,699	15.00	0.2746	3
5	2.334	25.99	21,663	21,700	15.00	0.0020	1
1	1.854	27.84	21,662	21,701	15.10	0.1010	2
4	0.6109	28.45	21,661	21,702	15.10	0.0030	2
3	0.5875	29.04	21,660	21,703	16.01	0.9020	5
7	2.781	31.82	21,659	21,704	17.25	1.240	5
2	1.200	33.02	21,658	21,705	26.81	9.560	4
9	23.98	57.00	21,657	21,706	28.07	1.259	5
3	2.626	59.63	21,656	21,707	28.07	0.0044	2
1	1.178	60.81	21,655	21,708	28.29	0.2178	2

- What data structures would need to be maintained?
- This could consume excessive gas on-chain.
- Imagine implement order book on Ethereum mainnet.
- It may not be a feasible idea at this point.

Reference: Uniswap V3 Book

AMM

AMM (Automated Market Makers): Use algorithms to manage liquidity in an automated way, operating similarly to traditional market makers.

Uniswap

Balancer

Curve

$x * y = k$

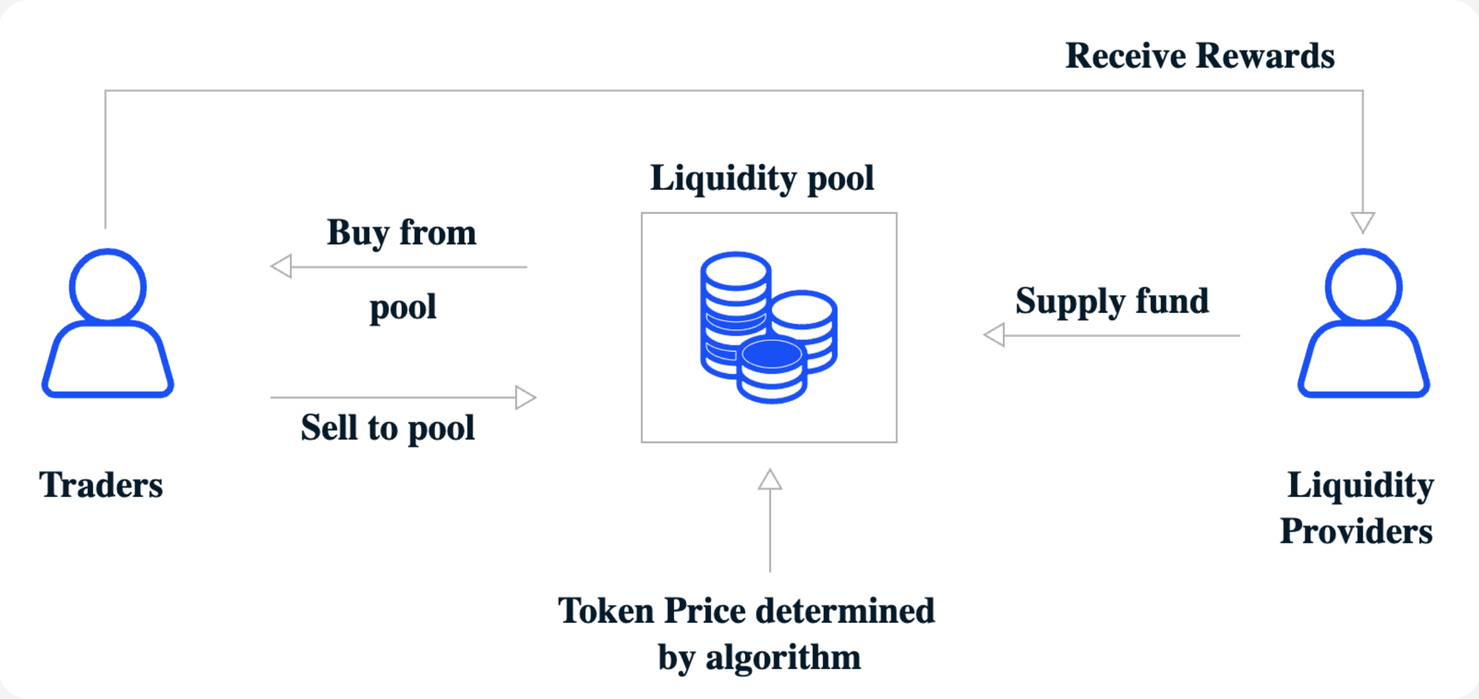
$$\prod_{i=1}^{n} x_i^{w_i} = k$$

$$An^n \sum x_i + D = ADn^n + \frac{D^{n+1}}{n^n \prod x_i}.$$

The good news is, we will only teach Uniswap
The bad news is, we will only teach Uniswap

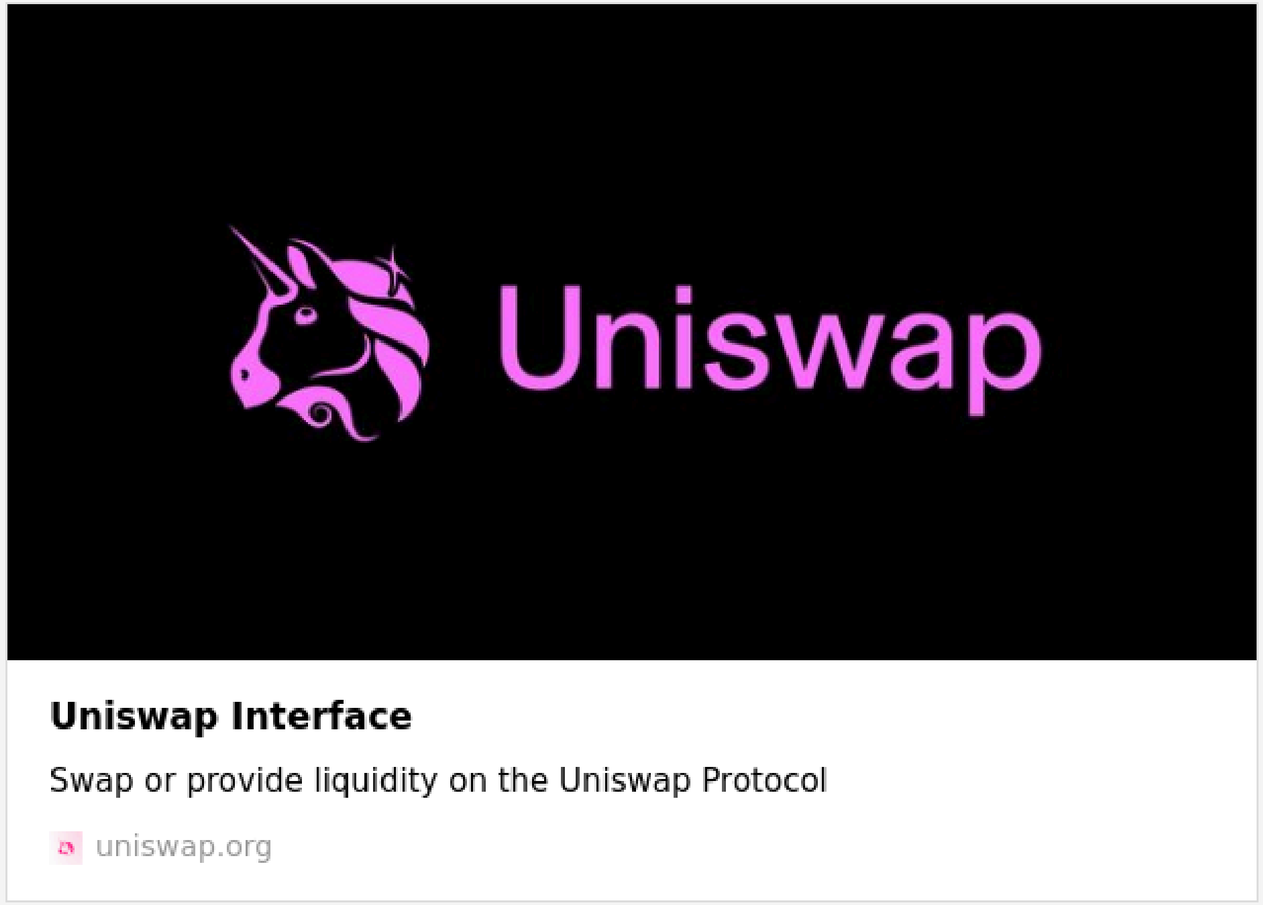
AMM

AMM (Automated Market Makers): Use algorithms to manage liquidity in an automated way, operating similarly to traditional market makers.



Uniswap

Before diving into the underlying mathematics and mechanisms, let's explore how to interact with Uniswap on their website.



Uniswap V2

Algorithm: x * y = k, where k is a constant

- x: the amount of token0 in the pair
- y: the amount of token1 in the pair

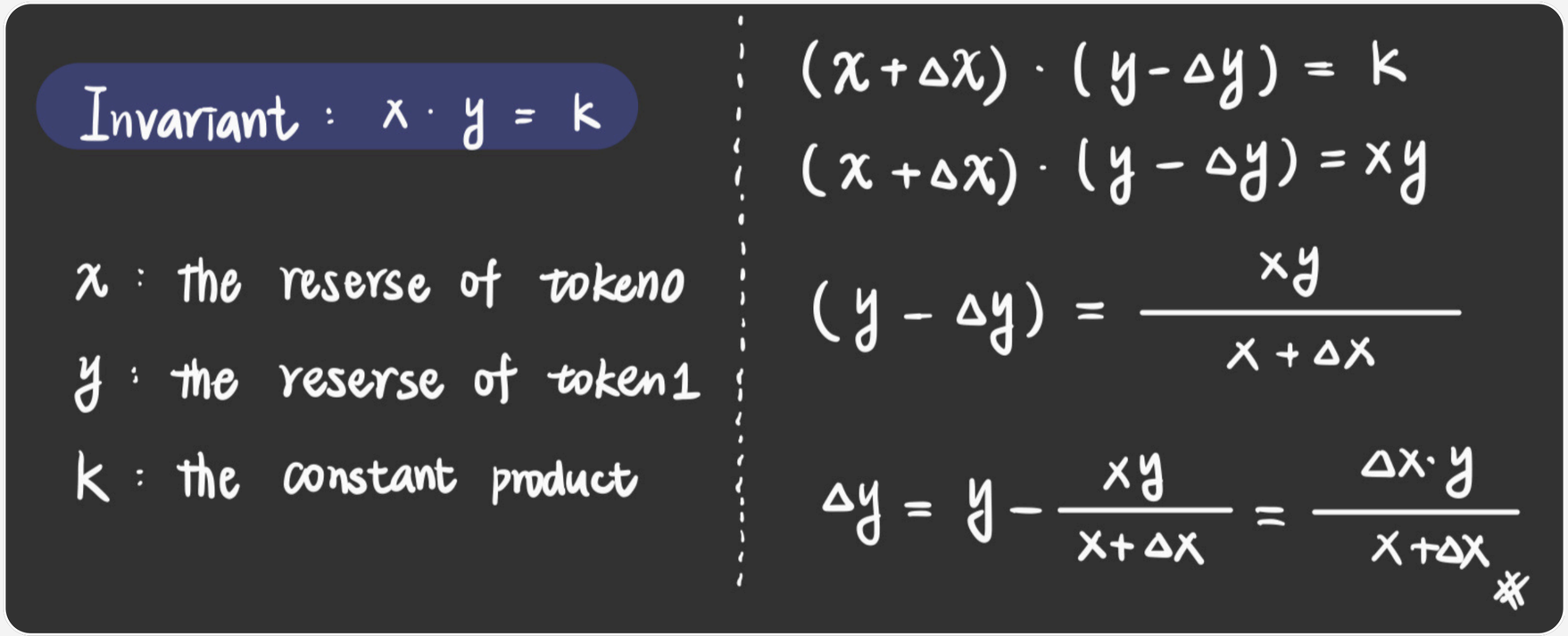
Assumption: There is no swap fee.

Case 1: We want to swap x' token0 for token1, how much will I get?

Case 2: We want to swap y' token1 for token0, how much will I get?

Swap Without Fee

Case 1: We want to swap x' token0 for token1, how much will I get?



Uniswap V2

Algorithm: $x * y \leq k$, where k is a constant

- x: the amount of token0 in the pair
- y: the amount of token1 in the pair

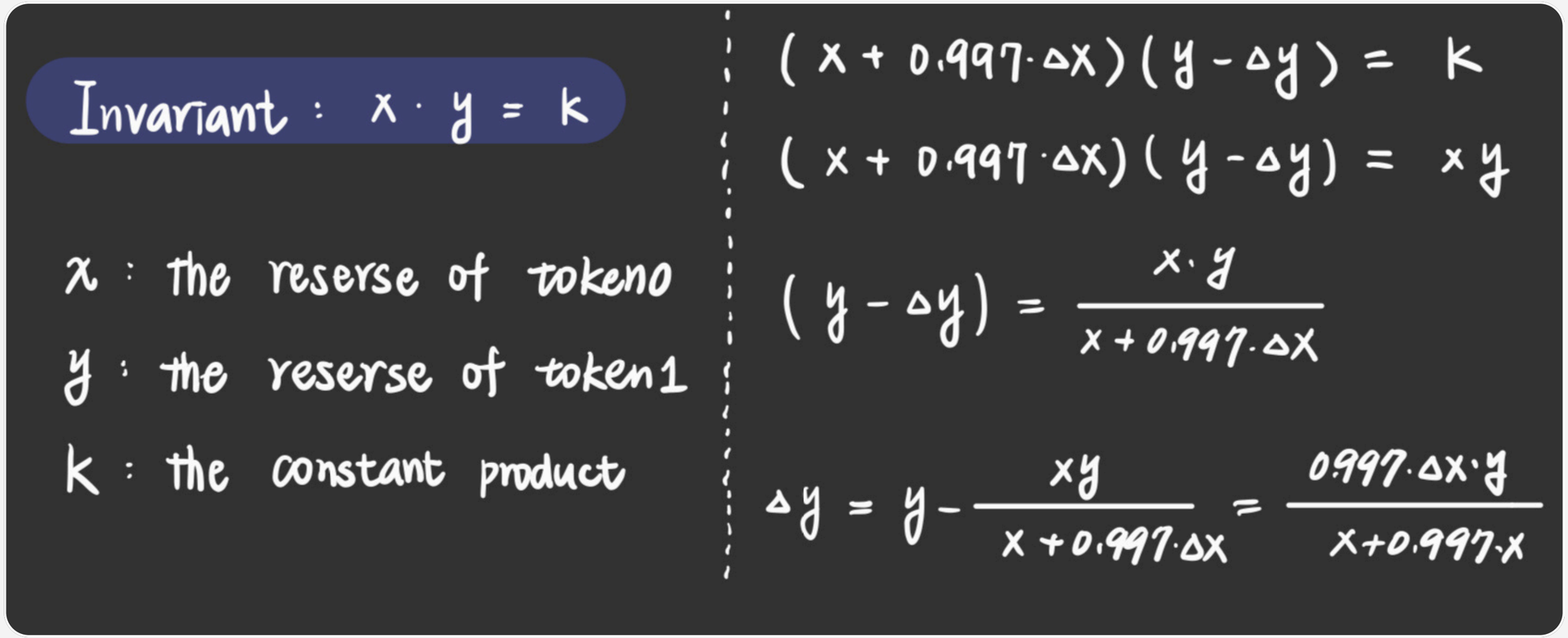
Assumption: There is **0.3%** swap fee.

Case 1: We want to swap x' token0 for token1, how much will I get?

Case 2: We want to swap y' token1 for token0, how much will I get?

Swap With Fee

Case 1: We want to swap x' token0 for token1, how much will I get?



Router::swapExactTokensForTokens

Given input token, amount and the path, conduct the swap and get output result

- amountOutMin: specify the minimum output token amount
- deadline: specify the last timestamp the transaction should be executed

```
function swapExactTokensForTokens(
    uint amountIn,
    uint amountOutMin,
    address[] calldata path,
    address to,
    uint deadline
) external override ensure(deadline) returns (uint[] memory amounts) {
    amounts = UniswapV2Library.getAmountsOut(factory, amountIn, path);
    require(amounts[amounts.length - 1] >= amountOutMin, 'UniswapV2Router: INSUFFICIENT_OUTPUT_AMOUNT');
    TransferHelper.safeTransferFrom(path[0], msg.sender, UniswapV2Library.pairFor(factory, path[0], path[1]), amounts[0]);
    _swap(amounts, path, to);
}
```

Slippage Check, this is very important when integration

Router::swapExactTokensForTokens

Given input token, amount and the path, conduct the swap and get output result

- First validate the amountOutMin and deadline
- Transfer the token to the router: the user should first approve the router

```
function swapExactTokensForTokens(
    uint amountIn,
    uint amountOutMin,
    address[] calldata path,
    address to,
    uint deadline
) external override ensure(deadline) returns (uint[] memory amounts) {
    amounts = UniswapV2Library.getAmountsOut(factory, amountIn, path);
    require(amounts[amounts.length - 1] >= amountOutMin, 'UniswapV2Router: INSUFFICIENT_OUTPUT_AMOUNT');
    TransferHelper.safeTransferFrom(path[0], msg.sender, UniswapV2Library.pairFor(factory, path[0], path[1]), amounts[0]);
    _swap(amounts, path, to);
}
```

Router::swapExactTokensForTokens

Given input token, amount and the path, conduct the swap and get output result

- Loop over the path and conduct swap for each token pair

```
// requires the initial amount to have already been sent to the first pair
function _swap(uint[] memory amounts, address[] memory path, address _to) private {
    for (uint i; i < path.length - 1; i++) {
        (address input, address output) = (path[i], path[i + 1]);
        (address token0,) = UniswapV2Library.sortTokens(input, output);
        uint amountOut = amounts[i + 1];
        (uint amount0Out, uint amount1Out) = input == token0 ? (uint(0), amountOut) : (amountOut, uint(0));
        address to = i < path.length - 2 ? UniswapV2Library.pairFor(factory, output, path[i + 2]) : _to;
        IUniswapV2Pair(UniswapV2Library.pairFor(factory, input, output)).swap(amount0Out, amount1Out, to, new bytes(0));
    }
}
```

Router Swap Operation

There are other variation for router to conduct swap operation

- swapTokensForExactTokens: Given output token, amount, and path, returns the token required for the swap.
- swapExactETHForTokens: Given **ETH**, amount and path, the router will first convert ETH to WETH and later conduct the swap.
- swapExactTokensForTokensSupportingFeeOnTransferTokens

DeFi News

DeFi News

區塊鏈跨境支付來了！瑞銀推出 UBS Digital Cash：支援美/歐元/人民幣..多種法幣轉帳

瑞銀（UBS）在 11 月 7 日發佈公告表示，其已經成功開發並試點了 UBS 數位現金，並基於區塊鏈實現了多幣種的跨境支付。 （前情提要：瑞銀推出首個代幣化基金「上鏈以太坊」，TradFi選擇ETH的理由是什麼？） （背景補充：瑞銀集團香港首發「代幣化股權」...

動區動趨-最具影響力的區塊鏈新聞媒體 / Nov 8, 2024

瑞銀推出首個代幣化基金「上鏈以太坊」，TradFi選擇ETH的理由是什麼？

瑞士聯合銀行（UBS）在強調投資者對代幣化金融資□的興趣日益濃厚後，推出了基於以太坊的旗下首個代幣化基金。加密評論員稱，此舉相當於將 ETH 置於傳統金融的核心，看好對 ETH 幣價帶來正面影響。 （前情提要：SmartCon演講全文》資□管理的未來-...

動區動趨-最具影響力的區塊鏈新聞媒體 / Nov 2, 2024

DYOR