Elements of DeFi

https://web3.princeton.edu/elements-of-defi/

Professor Pramod Viswanath

Princeton University

Lecture 11

Lending protocols

Last Lecture: Oracles and attacks

- On-chain oracles
  - DeFi applications as oracles : AMMs
- Price feed stabilization mechanisms
  - TWAP, VWAP feeds
- Security
  - Case study of a specific attack
  - Cost and Profit of oracle corruption
- Open problems

This lecture: Lending protocols

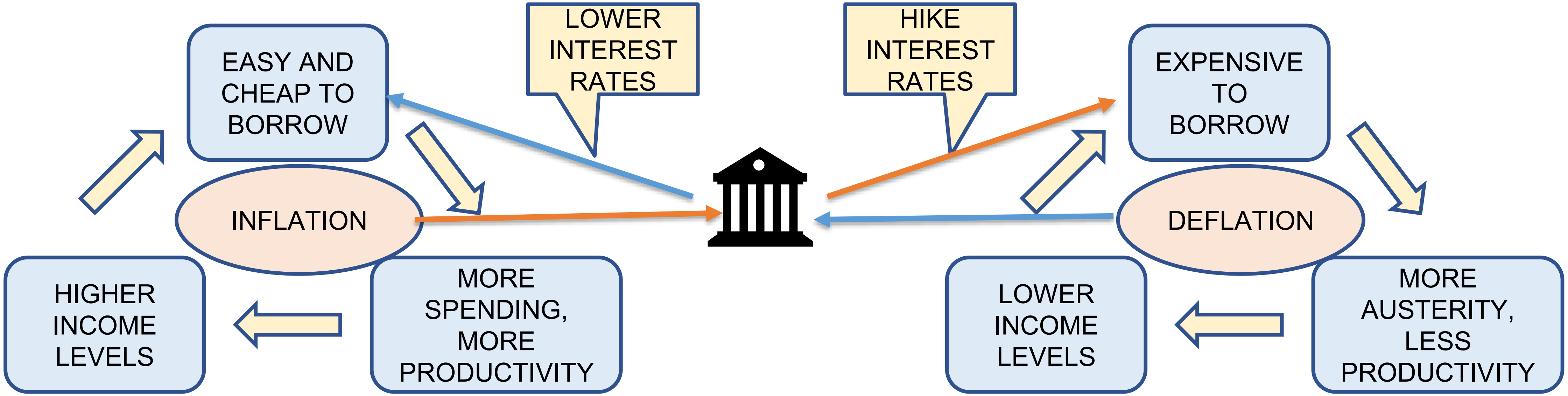
- Need for lending in any economy
  - Driven by trust in lenders
  - Track reputation of borrowers

- Decentralized Lending
  - Need for over-collateralization
  - Agents involved and their incentives – lender, borrower, liquidator, oracle
  - Action space – shorting, arbitrage, changing interest rates

- Under-collateralized lending: proposals

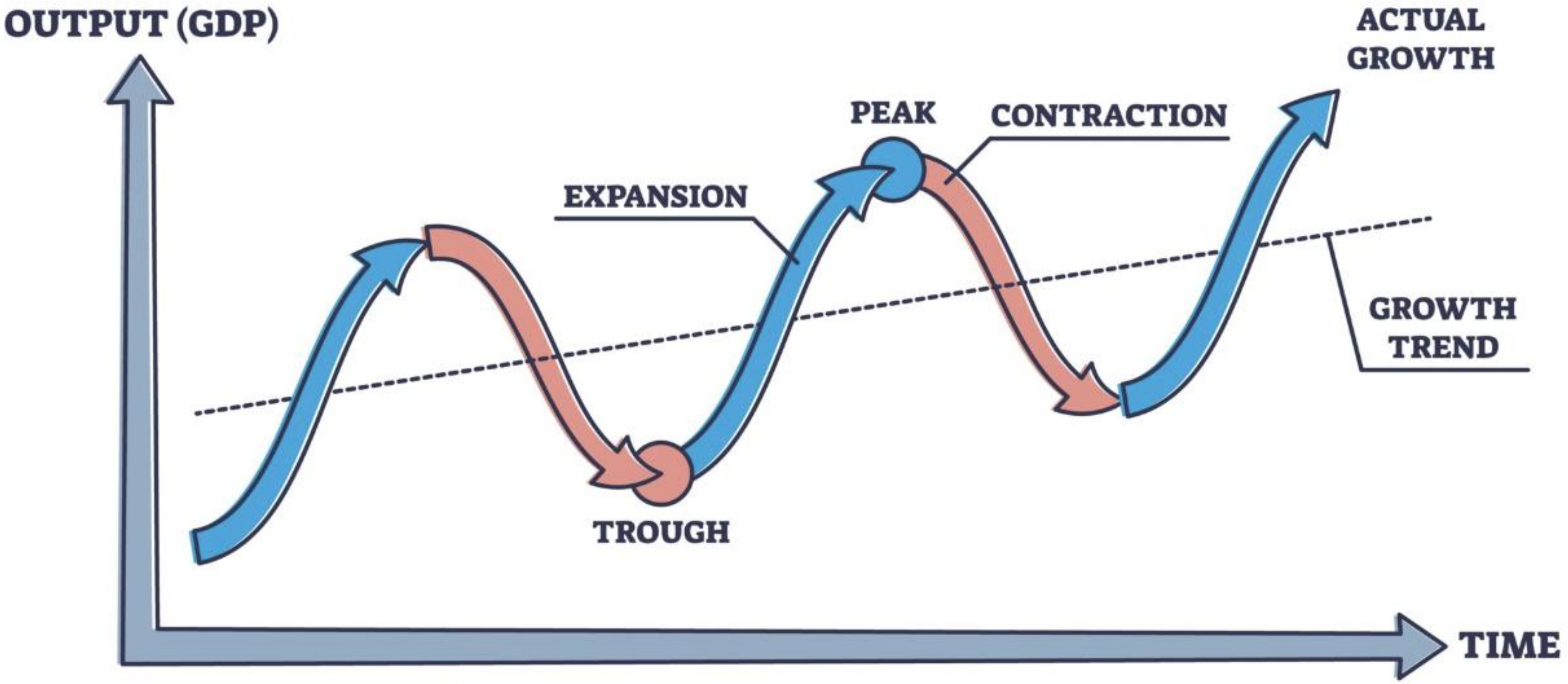
Need for lending

- Taking on debt enables borrowers to create value in future
- Created value is then used to pay back the debt
- Interest rates act as the “temperature control”



Need for lending

- Leads to “debt cycles” in an economy



Lending in TradFi

- Banks lend out money
  - Trusted to lend out responsibly
  - Trusted to balance risk of defaults with enough assets/deposits

WHAT IF BANKS FAIL TO DO THIS?

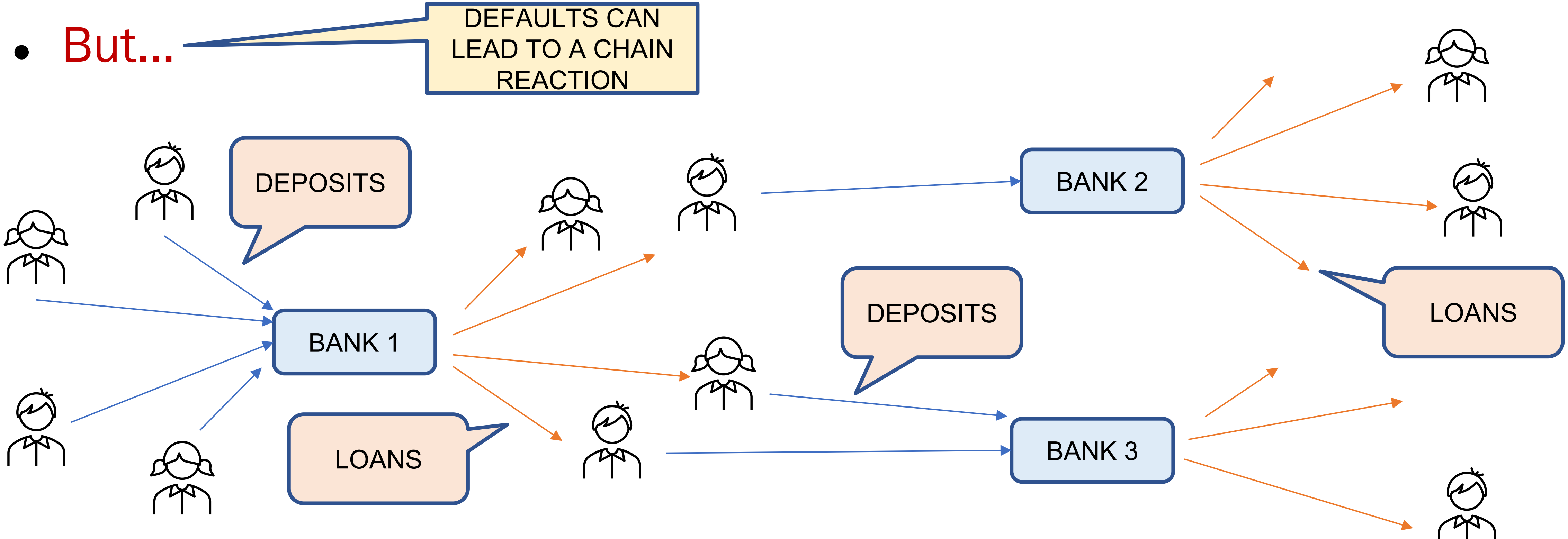
- Customers borrow money
  - Based on prior reputation or “credit score”
  - Based on income, assets

WHAT IF CREDIT RATING AGENCIES ARE MISTAKEN?

Lending in Tradfi: Monetary Multiplication

- Amount of active money in the economy gets multiplied by lending : $1 in the economy can be used to create $10 of productive value
- Leads to more capital efficiency
- But…

DEFAULTS CAN LEAD TO A CHAIN REACTION



DEPOSITS

BANK 1

LOANS

BANK 2

DEPOSITS

BANK 3

LOANS

Lending in TradFi: Vulnerability

2008 Financial Crisis: the perfect storm

- Banks did not lend out responsibly
- Credit rating agencies were mistaken about the credit scores
- Defaults did start a chain reaction

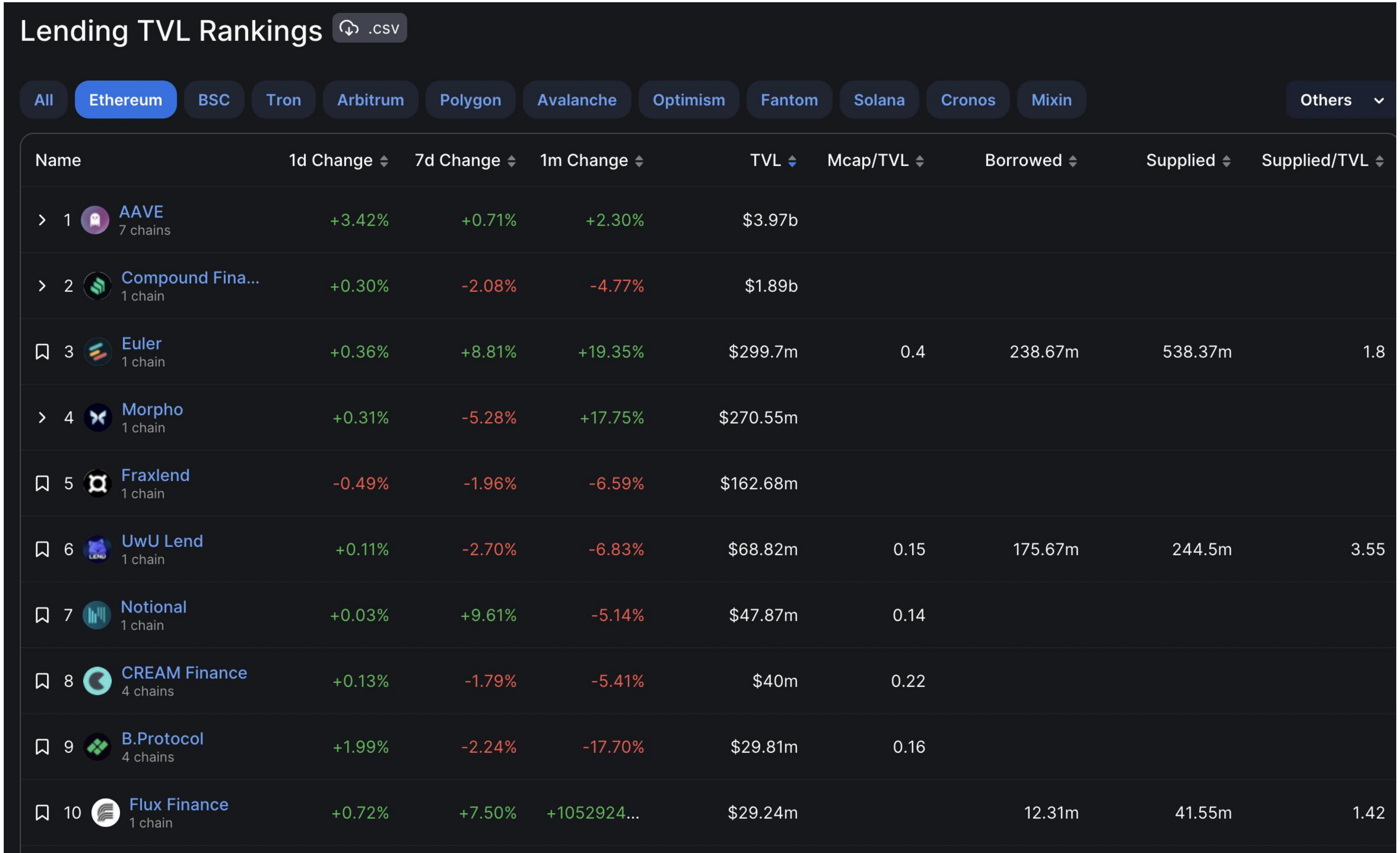
Need more transparency in

- The risks that lenders are taking on – deposits and collateral
- Metrics used to judge quality of loans – loan health factors

Decentralized lending

- How can lending be enabled in a trustless + permissionless setting?
- Borrower required to lock in collateral in asset 2 for every asset 1 borrowed
- Protocol is on-chain and transparent – health of loans can be assessed easily
- Other agents incentivized to check if the assets and liabilities of a lending contract are balanced - liquidators

Decentralized lending: TVL



Lending TVL Rankings

.csv

All | Ethereum | BSC | Tron | Arbitrum | Polygon | Avalanche | Optimism | Fantom | Solana | Cronos | Mixin | Others

	Name	1d Change	7d Change	1m Change	TVL	Mcap/TVL	Borrowed	Supplied	Supplied/TVL
1	AAVE 7 chains	+3.42%	+0.71%	+2.30%	$3.97b				
2	Compound Fina... 1 chain	+0.30%	-2.08%	-4.77%	$1.89b				
3	Euler 1 chain	+0.36%	+8.81%	+19.35%	$299.7m	0.4	238.67m	538.37m	1.8
4	Morpho 1 chain	+0.31%	-5.28%	+17.75%	$270.55m				
5	Fraxlend 1 chain	-0.49%	-1.96%	-6.59%	$162.68m				
6	UwU Lend 1 chain	+0.11%	-2.70%	-6.83%	$68.82m	0.15	175.67m	244.5m	3.55
7	Notional 1 chain	+0.03%	+9.61%	-5.14%	$47.87m	0.14			
8	CREAM Finance 4 chains	+0.13%	-1.79%	-5.41%	$40m	0.22			
9	B.Protocol 4 chains	+1.99%	-2.24%	-17.70%	$29.81m	0.16			
10	Flux Finance 1 chain	+0.72%	+7.50%	+1052924...	$29.24m		12.31m	41.55m	1.42

Decentralized lending: setup

- Most DeFi lending currently is “over-collateralized”

- Every token to be lent out has a liquidation threshold
  - e.g. Liquidation threshold = 75% means that every $100 worth of collateral posted can allow at most $75 worth of token to be borrowed
  - More collateral posted = Healthier loan

- Health Factor in terms of collateral, threshold and borrowed amount:

$$H_f = C_{USD} \frac{L_{Threshold}}{B_{USD}}. \qquad \text{Example:} \, H_f = 100 \, \frac{0.75}{75} = 1$$

Decentralized lending: agents

Decentralized lending sets up incentives for the following agents that interact with each other to create a healthy lending environment

Protocol smart contract

Lender

Borrower

Liquidator

Oracle

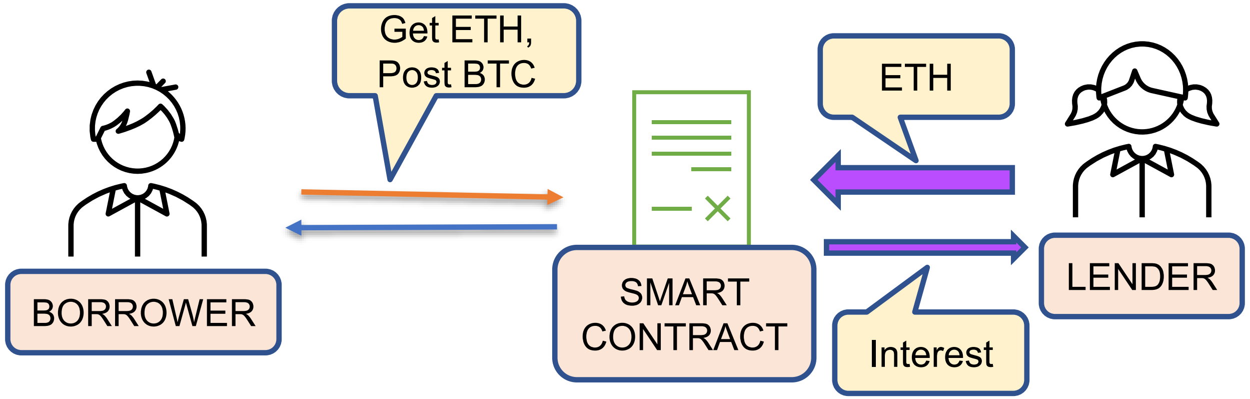
Agent: Smart Contract

- Smart Contract specifies
  - when borrowing can be done
  - when a loan can be liquidated
  - how much collateral needs to be posted to keep a loan afloat
  - which assets are accepted as collateral

All these parameters can be changed by protocol “governance” – users who have some “stake” in the protocol – usually lenders

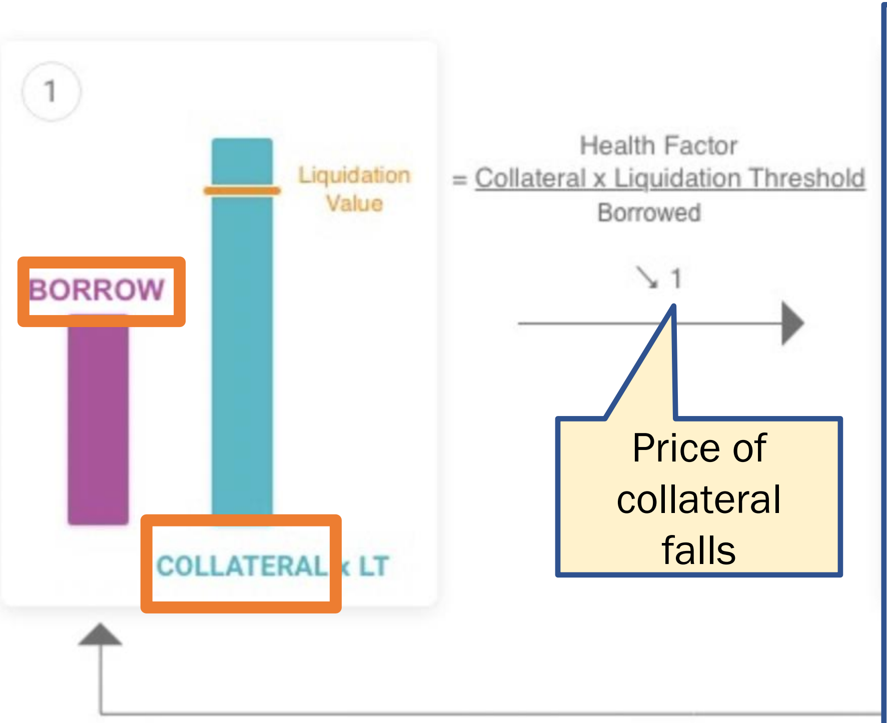
Agents: Lender and Borrower

- Lender
    - Has some capital lying around and wants it to generate yield/interest
- Borrower
    - Has a need for capital that would generate value in the borrowed token
    - Incentivized to keep loan afloat with enough collateral, gets penalized o/w



Agent: Liquidator

- If Health Factor H_f goes below 1, then liquidation is triggered
- Part of the loan that is underwater is repaid by selling corresponding collateral at a discount – who does this? - liquidator



Price of collateral falls

Agent: Liquidator

- Liquidator
  - can liquidate the loan if value of collateral posted goes below threshold
  - needs to pay back part of the loan and gets corresponding part of collateral at a discount

e.g.

Alice borrows $50 worth of ETH by posting $100 worth of BTC. Liquidation threshold is 2/3. Liquidator reward is 5%.

- If price of BTC falls, when will it trigger a liquidation?
- Assume BTC collateral worth only $60 after price fall – How much of the loan in supported? How much is underwater?

Liquidation in Action

e.g. (contd.)

Alice borrows $50 worth of ETH by posting $100 worth of BTC. Liquidation threshold is 2/3. Liquidator reward is 5%.

- Assume BTC collateral worth only $60 after large price fall
- Liquidator decides to liquidate $40 worth of loan
- Liquidator pays smart contract $40 in ETH
- Smart contract pays liquidator $42 in BTC
  - $2 = reward for liquidator
- New loan position?
  - $10 in ETH loaned out, $18 in BTC collateral
  - $2 = penalty for borrower

Is liquidating only $10 of the loan possible?

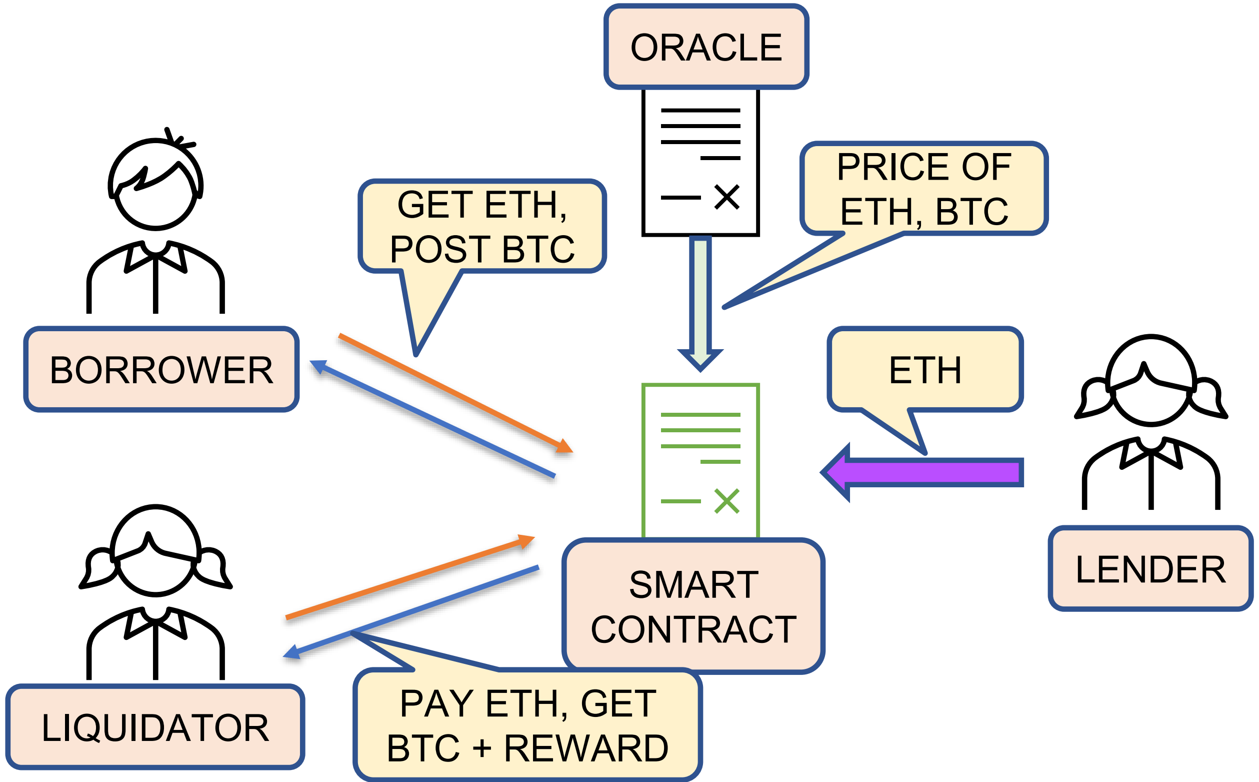
Why not?

Agent: Oracle

How does smart contract know value of collateral vs loan?

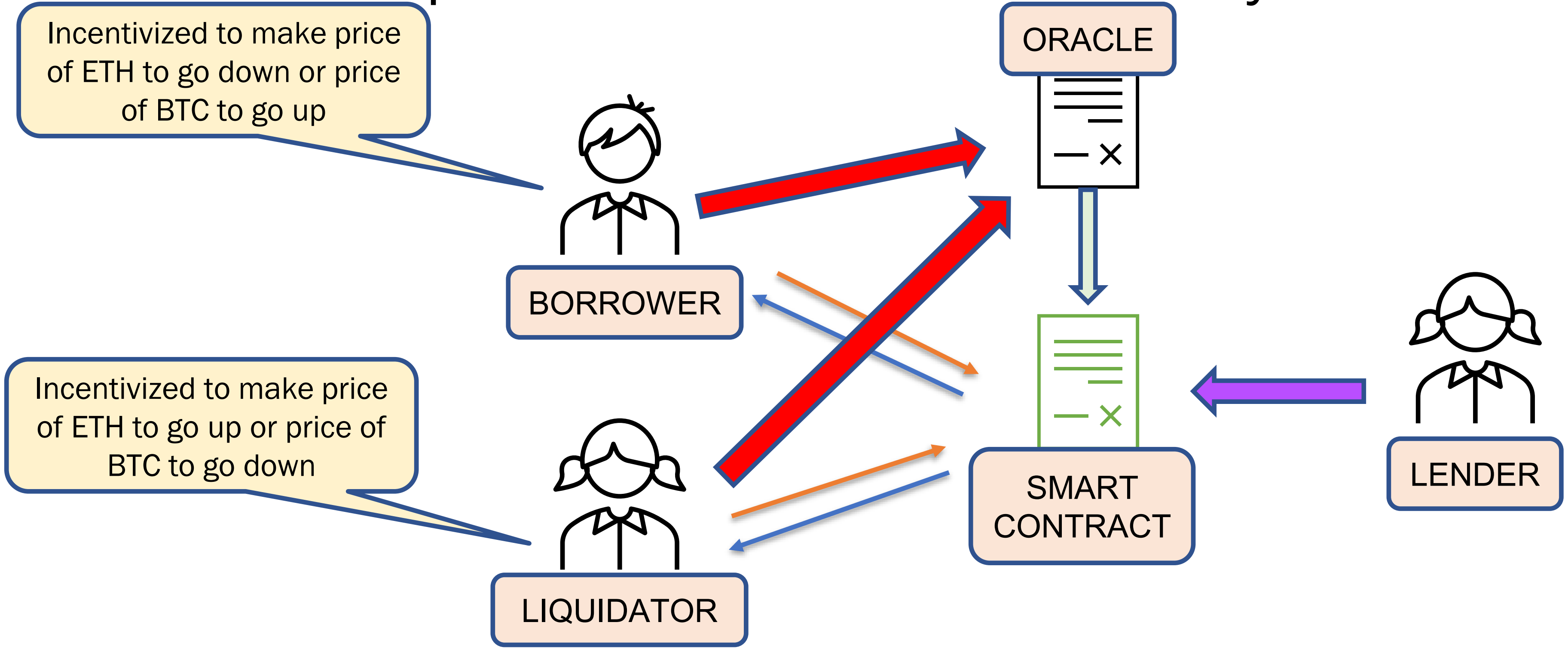
If liquidator wants to trigger liquidation, how does smart contract verify if value of the collateral is actually below threshold?

- Oracle



Incentives of Oracle

- Risk: Oracles can be a point of failure - can be manipulated
- Make collateral price fall to trigger false liquidations
- Inflate price of collateral and run away with loans at a lower price



Incentivized to make price of ETH to go down or price of BTC to go up

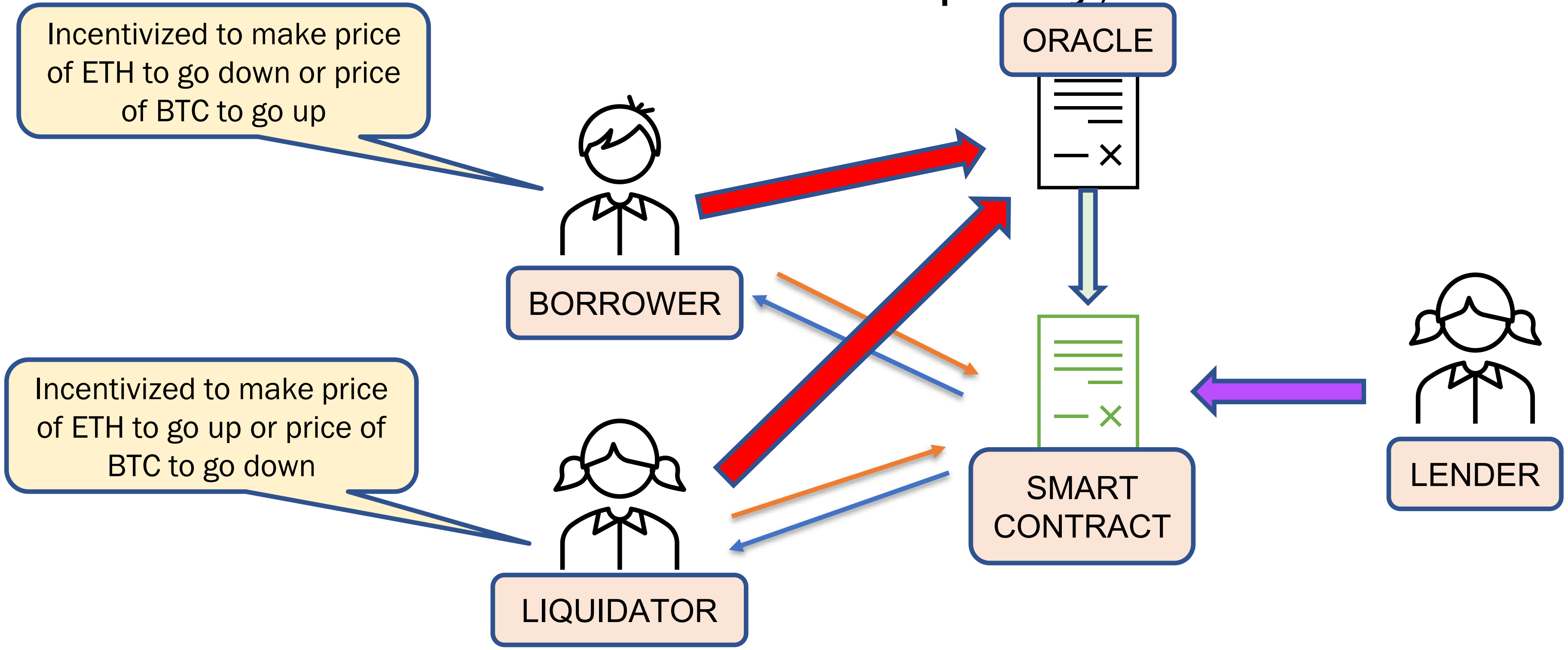
Incentivized to make price of ETH to go up or price of BTC to go down

Vulnerabilities

Main flaws exploited in such attacks?

Recall : bZx and Mango attacks

- Lending relied on only one AMM as oracle
- Oracle lacked sufficient liquidity, tokens obscure or rarely traded



Incentivized to make price of ETH to go down or price of BTC to go up

Incentivized to make price of ETH to go up or price of BTC to go down

Lenders' incentives: Interest rates

- Each liquidity pool has utilization rate which is used to decide the interest rate

$$U_t = \frac{TotalBorrows}{TotalLiquidity}$$

- As U_t goes to 100%, liquidity becomes scarcer, need to increase interest rate
- Tradeoff: as liquidity becomes scarcer, lenders get higher rate of return, but might not be able to withdraw all of it in case of a bankrun
- Protocol tries to keep value of U_t near a fixed $U_{optimal}$ by controlling the interest rate

Algorithmic Stabilization via Interest rates

- Interest rate R_t is changed in the following way $(R_{slope1} \ll R_{slope2})$

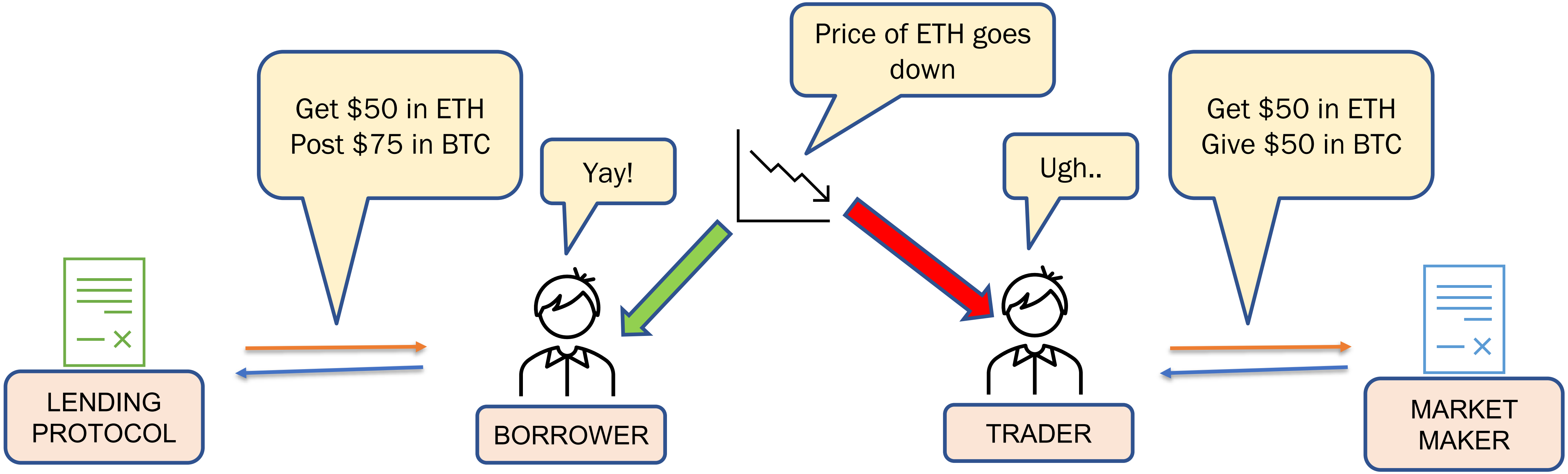
$$if\ U < U_{optimal}: \qquad R_t = R_0 + \frac{U_t}{U_{optimal}} R_{slope1}$$

$$if\ U \geq U_{optimal}: \qquad R_t = R_0 + R_{slope1} + \frac{U_t - U_{optimal}}{1 - U_{optimal}} R_{slope2}$$

- Parameters such as $R_0, U_{optimal}, R_{slope1}, R_{slope2}$ are decided by the protocol “governance” – chosen differently for each token
- Intuition:
  - More volatile assets are kept at low $U_{optimal}$ because lenders withdraw very often
  - Less volatile assets (stablecoins) are kept at high $U_{optimal}$ because lenders do not withdraw as often

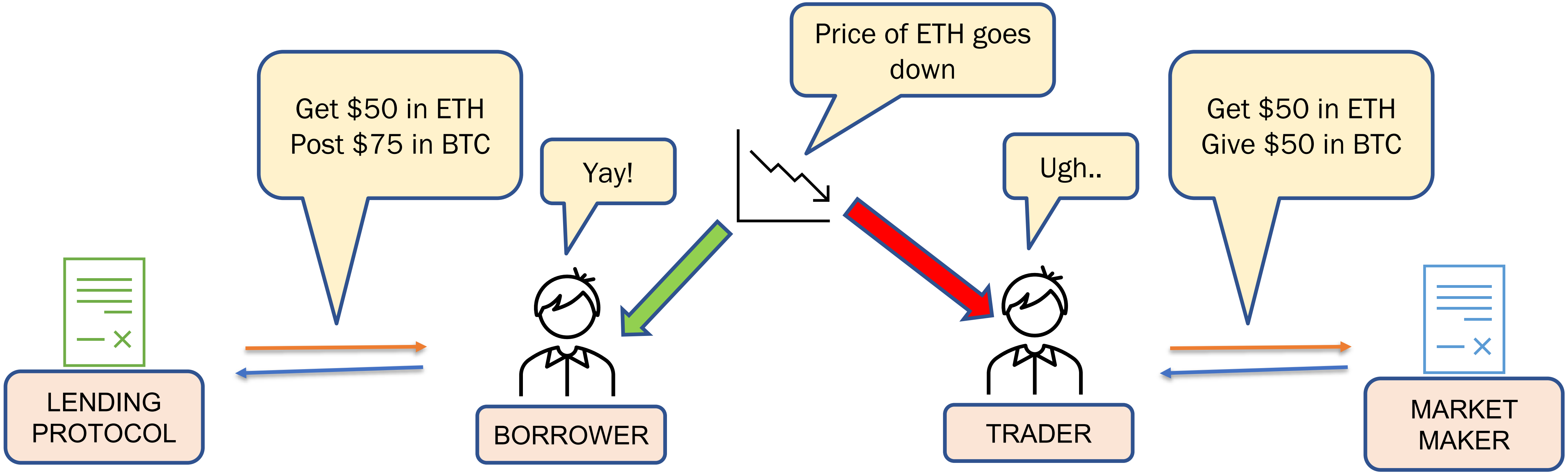
Lending enables shorting

- Why borrow a token A by posting collateral token B when you can just swap A for B on AMM?
- Think about what happens when price of A goes down



Lending enables shorting

- Lending protocols enables traders to short tokens, do margin trading
- However, need to make sure expected cost incurred from interest rate and posting collateral < expected profit from price falling



Under-collateralized lending

- Over-collateralized lending do not enable credit markets
- Need to enable under-collateralization to improve capital efficiency further
- Many centralized lenders have failed to manage risk
  - Celsius
  - Three Arrows Capital
  - FTX + Alameda

- Open problems
  - Decentralize under-collateralized lending?
  - Use ML models to compute a “credit-score” on-chain?

LECTURE ENDS