

Morpho Midnight Whitepaper

Bhargav Nagaraja Bhatt Morpho Association	Paul Frambot Morpho Association	Quentin Garchery Morpho Association
Mathis Gontier Delaunay Morpho Association	Adrien Husson Morpho Association	Paul-Adrien Nicole Morpho Association
Adrien Laversanne-Finot Morpho Association	Matthieu Lesbre Morpho Association	

May 2026

Abstract

Morpho Midnight¹ is a non-custodial fixed-rate lending protocol implemented for the Ethereum Virtual Machine. It is organized around isolated, immutable, permissionlessly created markets with fixed-maturity. Lending and borrowing are implemented through the trading of credit and debt units, whose payoff structure is analogous to that of zero-coupon obligations, settling at the market's maturity. Participants trade by posting or consuming offers that do not lock capital and source liquidity only at settlement, allowing makers to quote across multiple markets at once. Markets can range from single to multi-collateral configurations, and gates can be used to implement access-control policies.

¹<https://github.com/morpho-org/midnight>

Disclaimer

This paper is published for general information purposes only and describes the Midnight smart contracts at a technical level. It does not constitute, and must not be relied on as, investment, financial, tax, accounting or legal advice, nor as any recommendation or solicitation to buy, sell, supply, borrow against, or otherwise transact in any digital asset, token, or smart contract position. Nothing in this paper should be construed as an offer of securities or of any other financial instrument.

The terminology used throughout this paper is used solely in a descriptive and functional sense to refer to the mechanical behaviour of the Midnight smart contracts. None of these terms is intended to denote, and none should be construed as denoting, a regulated financial instrument, a security, a deposit, a loan within the meaning of any applicable banking, securities, or financial-services regulation, a regulated trading venue, a clearing facility, or any other regulated product, activity or status. In particular, the terms “trade”, “trading”, “settlement”, “settle”, “offer”, “maker”, “taker”, “quote”, “order”, “obligation”, “credit”, “debt”, “market”, “liquidity”, “liquidation”, “maturity”, “tick”, “zero-coupon” and any cognate terms are used in a descriptive and functional sense to refer to the mechanical behaviour of the Midnight smart contracts. None of these terms is intended to denote or characterise the smart contracts, the market units, or the actions of any participant as a security, a financial instrument, a regulated derivative, a bond, a fixed-income obligation, a commodity contract, a deposit, a loan within the meaning of any applicable banking, securities or financial-services regulation, or a regulated trading, clearing, or settlement venue. Any resemblance between the vocabulary used herein and concepts defined under applicable laws or regulations is incidental and does not reflect the legal characterisation of the protocol, its components, or the activities of its users.

The Midnight source code is made available as open-source and permissionless software on the basis of the licence(s) set out in the relevant code repository, and is provided “as is”, without warranty of any kind, express or implied, including without limitation any warranty of merchantability, fitness for a particular purpose, non-infringement, or accuracy. “Morpho” is a trademark of the Morpho Association; no licence to use it is granted by this paper, and any third-party use of these marks remains subject to applicable trademark law.

Any yields, returns, rates or other figures referenced in this paper or in related communications are variable, indicative only, not guaranteed, and subject to change without notice. Use of the protocol may be restricted in certain jurisdictions, and users are solely responsible for compliance with all laws and regulations applicable to them. The views and statements expressed in this paper are current as of its date of publication and are subject to change without notice and without any obligation to update.

To the maximum extent permitted by applicable law, the authors and the Morpho Association each disclaim any and all liability for any direct, indirect, incidental, consequential or special losses, damages, costs or claims arising from or in connection with any reliance on this paper or any use of the Midnight smart contracts.

1 Introduction

Lending and borrowing protocols are among the most widely used blockchain protocols, totaling about \$25b of active loans as of May 2026 [1]. Early lending protocols emerged in an environment constrained by thin and passive liquidity and high transaction costs. In that context, pool-based markets provided an effective design. To maximize liquidity concentration, protocols specified not only settlement and accounting, but also key pricing and risk parameters, aggregating all users into a single pool that can be entered and exited at will [2, 3]. This design works well when participant preferences are relatively homogeneous, but cannot accommodate multiple lending risk, liquidity, and compliance profiles without fragmenting the liquidity, as the range of assets, users, and credit use cases expands.

Morpho Blue [4] proposed a different architecture for onchain credit based on isolated and immutable markets with permissionless creation. The protocol itself makes no choices about which assets are credit-worthy or how capital should be allocated. Those decisions are deliberately left to lenders, who create and choose markets that match their needs. In practice, much of the supply comes from vaults built on top of the protocol. The market layer therefore stays minimal, while curation and allocation become a competitive layer above it.

Pool-based architectures go together with variable rates. The utilization of pools is controlled by an interest rate model [5], and symmetrically, the rate is “discovered” through the utilization. Although simple, this architecture comes with important tradeoffs. First, interest rate risk can be a direct barrier for borrowers who need predictable funding costs. It also makes new credit use cases harder to bootstrap, since modest flows can sharply move utilization in small markets. Symmetrically, lenders must closely monitor utilization to keep allocations in line with their risk-return preferences. Fixed-rate lending naturally addresses these limits. Although it has already been explored in DeFi [6], it has not yet emerged as the general foundation for onchain lending markets.

A central challenge in scaling onchain lending markets is bootstrapping liquidity. How effectively protocols make available liquidity accessible is a key determinant. This is especially important in isolated markets, where liquidity can fragment between collateral configurations and maturities, even though lenders are often willing to lend across several markets. In addition, participants often want to lend or borrow only at a specific rate, which might not be available at a given time or for the desired size. If such conditional liquidity must be provisioned upfront, it carries an opportunity cost that is likely to reduce the amount made available.

Midnight is a fixed-rate lending protocol for collateralized credit. Markets can accommodate a broad range of configurations, with arbitrary loan and collateral tokens (including multi-collateral), maturity, and access-controls using optional gates. As in Morpho Blue, markets are isolated, immutable, and permissionlessly created, serving as trustless primitives on top of which independent products can be built to serve various use-cases across different jurisdictions.

Lending and borrowing are implemented through the trading of fungible market *units*. These units have a fixed payoff structure analogous to that of zero-coupon obligations, representing credit for lenders and debt for borrowers. Trading occurs through executable *offers* published by *makers*, specifying the price at which they are willing to buy or sell these market units, without having to lock capital. *Takers* select offers offchain and submit them to the protocol, which settles if funds are available at execution time. Because liquidity is expressed through offers and capital is sourced only at execution, such markets can begin to function before consistent flow develops. Borrowers can seek liquidity against bespoke collateral configurations, while specialized

lenders can quote across many markets at the same time. Together, these properties expand the set of credit use cases that can be supported and bootstrapped onchain.

2 Fixed-maturity lending markets

2.1 Markets

Midnight is organized around isolated and immutable fixed-maturity lending markets, with configurations that cannot be altered after creation. Each market specifies a loan token, a maturity, and a set of accepted collateral assets, with their respective parameters.

Within a market, positions are accounted in units. One debt unit is an obligation to repay one loan token before maturity, and one credit unit is a claim on those repaid tokens. Buying units increases your credit, and selling units increases your debt. The user rate is naturally implied from the discount at which units are traded: for any traded price $P > 0$, the simple rate over the remaining term is $r = \frac{1}{P} - 1$.

2.2 Fungibility

Every trade involves a buyer and a seller, but the settlement results in fungible positions within the market rather than in a lasting bilateral relationship between those two parties. Credit and debt are accounted for at the market level, and positions are not tied to the particular trade that created them.

As markets mature on fixed calendar dates rather than on rolling tenors from origination, positions created at different times but with the same maturity belong to the same market and are fungible with one another.

2.3 Early exits

Within a market, existing lenders and borrowers can reduce their outstanding credit and debt at any time by selling and buying units respectively (see Figure 1). A buyer closes out any debt before accumulating credit and a seller closes out any credit before accumulating debt. Early exits make users' payoffs more flexible and deepen liquidity for all participants, since entries and exits occur within the same unified market.

Trading is notably possible after the maturity, except that one cannot increase their debt then, preventing paths 1 and 3 in Figure 1. This is enabled to facilitate unwinding in the event of unprofitable liquidations.

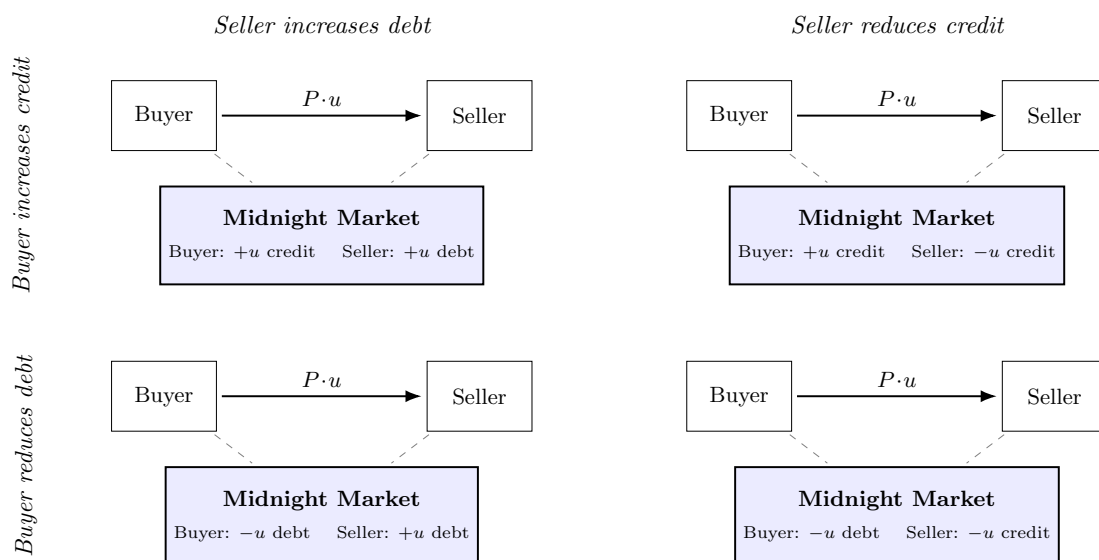


Figure 1: Illustration of the four possible trade cases depending on the initial positions of the buyer and seller.

3 Offer-based markets

3.1 Offers

Makers use *offers* to specify the price and maximum size they are willing to trade on a given market. Offers are not broadcasted at the protocol level and may be distributed through any external channel, either offchain or onchain.

A taker executes an offer by submitting it to the Midnight contract. Takes may be partial: any size up to the offer’s remaining capacity is permitted, and a single offer can be filled by multiple takers until exhausted. The contract settles atomically against the referenced market, creating, transferring, or burning the corresponding credit and debt units.

Each offer is attached with a *ratifier* contract that embeds the validation logic. It is called when an offer is taken. Typically, the ratifier verifies the validity of a signature of the offer against the maker’s public key. The modularity lets makers use different signature schemes such as passkeys or post-quantum schemes, or create more custom validation logic. Notably, this allows a single signature to ratify multiple offers (see next section).

3.2 Maker callbacks

Offers may specify a callback that executes at take time. This lets makers source the funds or collateral only when the offer is filled, instead of provisioning their positions in advance.

In particular, makers can keep the capital backing their offers deployed productively elsewhere until their offer is filled. For example, a lender may keep assets deployed in a Morpho Blue market while quoting a fixed-rate offer on Midnight. If the offer is taken, the callback withdraws

the necessary funds, provided sufficient liquidity is available, and completes settlement within the same transaction.

Callbacks are also particularly useful for rolling fixed-maturity exposure. A borrower approaching maturity may use a callback to buy back or repay debt in the current market and enter a later-maturity market atomically. Likewise, a lender may roll credit exposure from one maturity into another without first withdrawing into idle balances.

3.3 Multi-market offers

Callbacks allow makers to publish multiple offers backed by the same liquidity, which is key against liquidity fragmentation. To handle the liquidity available in different offers, offers can be assigned to a common consumption group that shares a fill budget. Executing any offer in the group decreases the remaining budget for all other offers in it; once the budget is exhausted, no further offer in the group can be filled. This keeps a maker’s exposure bounded by the budget rather than by the sum of all signed offers’ sizes.

To make this efficient at scale, ratifiers can support the ratification of a Merkle root of an offer set, allowing makers to quote multiple offers across many markets with a single signature or interaction. These offers can be taken later by presenting the corresponding Merkle proof.

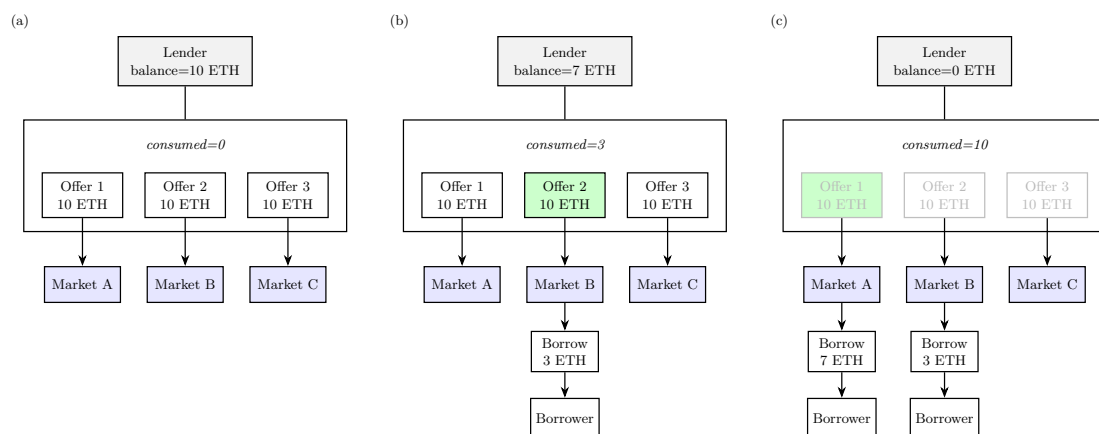


Figure 2: Example of the lifecycle of a multi-market offer.

3.4 Routing

The offer publication layer might not guarantee anything about the executability of an offer on Midnight. Therefore, a taker seeking the best available liquidity across outstanding offers faces a genuine search problem, required to take into account callback execution, consumption groups, and gas. This process, referred to as routing, takes place off protocol and can be performed by anyone.

This makes Midnight fundamentally different from a central limit order book logic: the protocol maintains no canonical queue of resting orders, no protocol-level price-time priority, and no protocol-level reservation of capital.

3.5 Tick structure

While the protocol does not impose a queue for offers, routers will naturally compare them in price order. Without a minimum increment, makers could undercut one another by economically insignificant amounts and the incentive to provide meaningful size would deteriorate.

To prevent this, Midnight enforces a discrete price tick grid for offers. Because a fixed price increment maps to different rate increments depending on time to maturity and participants typically reason in rate, the grid is defined such that consecutive ticks correspond to a constant relative change in implied total return. Therefore, for a given time-to-maturity, each tick step corresponds to the same relative change in rate.

Let δ denote the target relative increment in implied return, and let N denote the tick range. The unquantized price of tick n is given by:

$$\tilde{P}(n) = \frac{1}{1 + (1 + \delta)^{N/2-n}}, \quad (1)$$

It follows that consecutive ticks satisfy the following:

$$\underbrace{\frac{1}{\tilde{P}(n-1)} - 1}_{\text{return associated to tick } n-1} = (1 + \delta) \underbrace{\left(\frac{1}{\tilde{P}(n)} - 1 \right)}_{\text{return associated to tick } n}. \quad (2)$$

The appropriate tick size depends on the depth and activity of the market, which can vary significantly between markets. Midnight therefore allows tick grids to be configured per market by a *tickSpacingSetter* role. Each market starts with ticks corresponding to $\delta = 0.02$ (a 2% relative change in implied return per tick), and the value of δ may be decreased to $\delta = 0.01$ (1%) or $\delta = 0.005$ (0.5%). The definition of ticks ensures that when finer ticks are enabled, all previously valid ticks remain valid and outstanding offers remain executable. A market can therefore begin with coarse increments and move toward tighter quoting as depth and participation grow.

As maturity approaches, rate becomes less relevant as the quoting abstraction. To handle this regime under the same immutable grid, encoded prices are quantized to a granularity $\epsilon = 1 \times 10^{-6}$ by rounding each unquantized price to the nearest multiple of ϵ , with ties rounded downward. Away from par, quantization has no meaningful effect. Near $P = 1$, adjacent ticks collapse onto the same quantized price, so the effective minimum price increment becomes ϵ rather than the rate-based spacing.

4 Liquidations

4.1 Maximum debt

A market in Midnight may accept multiple collateral assets. Each collateral within a market has its own oracle and liquidation loan-to-value ratio (LLTV)². The maximum debt capacity of a borrower is the sum of the values of the collateral assets weighted by their respective LLTVs. For a borrower with collateral amounts c_1, \dots, c_k deposited against collaterals with prices p_1, \dots, p_k

²Possible LLTV values are constrained to a finite set to avoid liquidity fragmentation across markets that are not economically distinct.

(expressed in loan token per collateral token) and respective liquidation loan-to-value ratios $LLTV_1, \dots, LLTV_k$, the maximum debt capacity is:

$$\text{maxDebt} = \sum_{i=1}^k c_i \cdot p_i \cdot LLTV_i \quad (3)$$

A borrower cannot increase their debt above the `maxDebt` threshold, and their position is considered *healthy* as long as their maximum debt capacity does not fall below their debt. Unhealthy positions can be liquidated: a third-party (the *liquidator*) repays a portion of the debt and seizes a portion of the collateral. Repaid debt is available to be withdrawn for lenders in that market.

4.2 Liquidation incentive

During a liquidation, the ratio of collateral seized by debt repaid is based on the oracle price, with a discount; the value of this discount is determined by the liquidation incentive. In Midnight, in a given market, the maximum value of the liquidation incentive for each collateral is set by an immutable parameter, the liquidation cursor $\gamma \in \{0.25, 0.5\}$ ³:

$$\text{LIF}_{\text{max}}(LLTV, \gamma) = \frac{1}{1 - \gamma(1 - LLTV)}. \quad (4)$$

The appropriate value depends on the characteristics of the collateral. Decreasing γ preserves more of the overcollateralization of the borrower as a buffer against adverse price movements, reducing the risk of bad debt, provided the incentive still covers liquidators' costs. In contrast, increasing γ increases the incentive paid to liquidators, which may be necessary for collateral that is costly or risky to liquidate.

4.3 Liquidation of unhealthy positions

When a position becomes unhealthy, it becomes liquidatable. This prevents solvency deterioration that could lead to lender losses. In such cases, the liquidation incentive is equal to `LIFmax`. Unhealthy positions may be liquidated both before and after maturity.

The amount repaid by the liquidator is capped at the level required to restore the position to health, a mechanism referred to as *the recovery close factor*. It prevents liquidating entire positions as soon as they become unhealthy whilst still incentivizing restoring position's health. An exception applies when the residual collateral after a liquidation would fall below a dust threshold⁴. In that case, the position can be liquidated entirely, to avoid leaving residual positions too small to liquidate profitably.

The recovery close factor matters particularly as the borrower must collateralize the full amount due at maturity at any time. Allowing a liquidator to close the entire position on a minor breach would force the borrower to surrender collateral against the full debt, even though only part of the term had elapsed.

³On Morpho Blue the same liquidation cursor $\gamma = 0.3$ is used for all markets.

⁴The dust threshold is a parameter defined at the market level and denominated in the loan asset. It should be set such that positions above this value can be liquidated profitably given typical gas costs on the host chain.

4.4 Liquidation of overdue positions

After maturity, any outstanding debt makes a position liquidatable regardless of its health, to allow lenders to redeem at maturity, even if a healthy borrower fails to repay on time.

In that case, the issue is one of liquidity access for lenders rather than an immediate risk of loss. Midnight therefore softens the maturity-based liquidation incentive through a mechanism assimilable to a Dutch auction to avoid transferring excessive value from the borrower to the liquidator. The liquidation incentive factor starts at 1 at maturity and increases linearly to LIF_{\max} over a 15 minute window. Capping the auction at LIF_{\max} ensures that the incentive eventually reaches the same level as in standard liquidations. Health-based liquidations remain available at any time after maturity, preserving lender protection against solvency deterioration.

4.5 Bad-debt accounting

A drop in the collateral value of a position can create a shortfall in the credit of lenders. In particular, lenders may not recover the portion of the debt that cannot be repaid through a full liquidation. That portion is referred to as bad-debt.

In Midnight, bad-debt is accounted for by reducing lenders' credit proportionally. More specifically, liquidations compute the maximum liquidatable amount, and immediately realize any excess debt, regardless of how much collateral the liquidator chooses to seize⁵.

5 Access-control gates

Midnight is designed to support flexible access-control conditions. Markets may specify up to two optional gate contracts at creation, fixed thereafter, which the protocol calls when gated actions are attempted. This lets each market embed conditions specific to its intended use.

The enter gate restricts who may increase their credit or debt position. It applies only to entry: a participant can always exit by withdrawing credit, repaying debt, or withdrawing collateral, even if the gate reverts. Because the gate is an external contract that may evolve over time, this restriction ensures that it cannot trap funds inside a market and remains an access-control layer rather than a custody risk.

The liquidator gate restricts who may liquidate positions. This enables markets in which liquidation, and therefore bad-debt realization, is limited to a designated set of actors.

6 Authorizations

Midnight uses a single coarse-grained authorization primitive. An account may authorize another address to act on its behalf. Authorization is deliberately not scoped by action or by market: once granted, it gives the authorized address full control over the authorizer's Midnight state,

⁵On Morpho Blue, bad-debt is realized only after collateral has been fully seized, so a position can be observably insolvent for some time before lenders' credit is reduced, creating an incentive for informed lenders to exit the market before the loss is accounted. Here, the time to realize a loss is set by how quickly a liquidator first acts on the position, considerably narrowing that window.

including the ability to add or remove other authorizations. This choice gives a lot of flexibility, but comes with a trade-off: scoped delegation is not expressible natively. Integrations that want to grant a third party only a subset of actions must do so through an intermediate contract that holds the authorization and exposes a narrower interface to its own users. Granularity therefore lives on top of the protocol rather than inside it.

7 Fees

Midnight can charge two protocol-level fees on each market: a settlement fee on each trade and a continuous fee that accrues over time on outstanding credit. Both are bounded by immutable caps, giving participants a permanent upper bound on the fees the protocol can charge. Default fee values are configured per loan token and can be overridden per market. A designated role sets these values, while another role receives the accrued fees.

7.1 Settlement fee

The settlement fee is time to maturity dependent and is charged at take time. The fee inserts a spread between the buyer’s and seller’s settlement prices. Given a buy offer (resp. sell offer) at price P , the taker sells at $P - f(TTM)$ (resp. buys at $P + f(TTM)$).

The fee rate $f(\text{ttm})$ is a piecewise linear continuous function in time to maturity, with breakpoints at 0, 1, 7, 30, 90, 180 and 360 days, and constant after maturity and above 360 days:

$$f(\text{ttm}) = \frac{f_k(t_{k+1} - \text{ttm}) + f_{k+1}(\text{ttm} - t_k)}{t_{k+1} - t_k}, \quad (5)$$

where $\text{ttm} \in [t_k, t_{k+1}]$ and f_k, f_{k+1} are the fees at the surrounding breakpoints.

The fee value at each breakpoint is bounded by a hardcoded maximum. The maxima are calibrated so that the implied annualized rate $f(\text{ttm}) \cdot 360/\text{ttm}$ does not exceed 50 bps. $\text{ttm} = 0$ is an exception, its cap is 0.14 bps.

7.2 Continuous fee

The continuous fee accrues each second on outstanding credit and is borne by lenders. It materializes when the lender interacts with the market with a reduction in their credit. Lenders crystalize their future fees when increasing their credit; therefore, later changes in the continuous fee value do not impact existing lending positions. The continuous fee is capped at 1% annualized.

8 Acknowledgments

This white paper is the result of extensive collaboration and many fruitful discussions during the development of Midnight. The authors would especially like to thank members of the Morpho Association, as well as Charles Bertucci, Louis Bertucci, Guillaume Garchery, Colin González, and Olivier Guéant.

References

- [1] DefiLlama. [DefiLlama](#).
- [2] Aave. [Aave v1 Protocol Whitepaper](#), 2020.
- [3] Robert Leshner and Geoffrey Hayes. [Compound: The money market protocol](#), 2019.
- [4] Mathis Gontier Delaunay, Paul Frambot, Quentin Garchery, and Matthieu Lesbre. [Morpho Blue Whitepaper](#), 2023.
- [5] Charles Bertucci, Louis Bertucci, Mathis Gontier Delaunay, Olivier Gueant, and Matthieu Lesbre. Agents' behavior and interest rate model optimization in defi lending. *Mathematical Finance*, 36(2):374–396, 2026.
- [6] Dan Robinson and Allan Niemerg. [The yield protocol: on-chain lending with interest rate discovery](#), 2020.