# Методология высокоточного мониторинга институциональных позиций в протоколах Aave и Compound: Технологии пакетного извлечения данных и модели риск-коррекции LST-активов

Развитие экосистемы децентрализованных финансов (DeFi) привело к усложнению стратегий управления капиталом, особенно среди крупных держателей активов, так называемых «китов». Для построения эффективного трекера таких участников рынка недостаточно простого отслеживания балансов токенов в кошельках. Современные стратегии подразумевают активное использование кредитных плеч в протоколах денежных рынков, таких как Aave и Compound. Центральной метрикой для анализа деятельности таких субъектов является «Чистая позиция» (Net Position), которая определяется как разница между стоимостью предоставленного обеспечения (Collateral) и объемом заимствованных средств (Debt). Однако волатильность рынка и специфические риски производных инструментов, таких как токены ликвидного стейкинга (LST), вносят существенные искажения в расчеты. В частности, отклонение цены $stETH$ от $ETH$ (de-pegging) может создавать ложные сигналы о ликвидациях или притоках капитала. Данное исследование детально рассматривает механизмы извлечения данных через технологию Multicall и предлагает математический аппарат для фильтрации шума в сигналах за счет коррекции стоимости портфеля с учетом реальной рыночной ликвидности LST.

## Технологический стек извлечения данных: Архитектура Multicall3

Для реализации мониторинга в режиме реального времени критически важна атомарность данных. Запросы к блокчейну через стандартные JSON-RPC методы для каждого отдельного актива и каждого пользователя создают огромную нагрузку на инфраструктуру и, что более важно, приводят к получению данных из разных блоков. Это недопустимо при расчете Net Position, так как цена актива может измениться между запросом баланса залога и запросом суммы долга. Решением является использование контракта Multicall3, развернутого по универсальному адресу 0xcA11bde05977b3631167028862bE2a173976CA11.1

### Механика работы и преимущества Multicall3

Multicall3 позволяет объединить сотни вызовов view-функций в один запрос eth\_call. В контексте трекера китов это обеспечивает получение всех параметров позиции (залог, долг, коэффициенты ликвидности) в рамках одного и того же номера блока (Block Height), что гарантирует внутреннюю непротиворечивость данных.2

Основной функцией для работы является aggregate3, которая принимает массив структур Call3. Каждая структура включает адрес целевого контракта, флаг allowFailure и закодированные данные вызова (calldata).2

| **Параметр Call3** | **Тип** | **Описание** |
| --- | --- | --- |
| target | address | Адрес контракта (например, Aave Pool или Compound Comet) |
| allowFailure | bool | Если true, ошибка в одном вызове не отменяет весь пакет запросов |
| callData | bytes | Закодированная сигнатура функции и её аргументы |

Для извлечения данных о «чистой позиции» необходимо агрегировать вызовы к основным контрактам протоколов. Использование allowFailure: true критично, так как кит может иметь позиции в Aave V3, но не иметь их в Compound, и наоборот. Пакетный запрос позволит получить данные по всем возможным рынкам за один раз, игнорируя пустые результаты.2

## Глубокий анализ механики Aave V3: Извлечение Net Position

Протокол Aave V3 представляет собой сложную систему смарт-контрактов, где логика хранения данных разделена между основным пулом (Pool) и вспомогательными провайдерами данных. Для получения полной картины по конкретному адресу необходимо взаимодействовать с контрактом AaveProtocolDataProvider.5

### Иерархия контрактов и функции запросов

Для первичной оценки здоровья портфеля используется функция getUserAccountData в основном контракте Pool. Она возвращает агрегированные значения в базовой валюте протокола (обычно USD в эквиваленте Chainlink или ETH).6

Комплексный результат getUserAccountData включает:

1. totalCollateralBase: Общая стоимость всех активов, используемых в качестве залога.
2. totalDebtBase: Общая сумма задолженности (стабильной и переменной).
3. availableBorrowsBase: Доступный лимит для заимствования.
4. currentLiquidationThreshold: Порог ликвидации (в процентах).
5. ltv: Максимальное отношение кредита к стоимости залога.
6. healthFactor: Коэффициент здоровья позиции. Если он опускается ниже 1, позиция подлежит ликвидации.6

Однако для качественного анализа «китов» требуется детализация по каждому активу. Здесь вступает в дело AaveProtocolDataProvider по адресу 0x497a1994c46d4f6C864904A9f1fac6328Cb7C8a6 (для Mainnet).5

Через Multicall необходимо вызвать функцию getUserReserveData(address asset, address user) для каждого поддерживаемого актива. Список активов можно предварительно получить через getAllReservesTokens().5

| **Поле в getUserReserveData** | **Значение для Net Position** |
| --- | --- |
| currentATokenBalance | **Collateral**: Текущий баланс залога с учетом процентов 5 |
| currentStableDebt | **Debt**: Сумма долга с фиксированной ставкой 5 |
| currentVariableDebt | **Debt**: Сумма долга с переменной ставкой 5 |
| usageAsCollateralEnabled | **Status**: Флаг использования актива как залога 5 |

Формула чистой позиции для конкретного актива $i$:

$$NetPosition\_i = currentATokenBalance\_i - (currentStableDebt\_i + currentVariableDebt\_i)$$

Важно учитывать, что Aave использует внутреннюю индексацию для расчета процентов. Балансы aTokens и variableDebtTokens динамически увеличиваются в каждом блоке. Вызов через Multicall возвращает уже актуализированное значение на момент блока, что избавляет от необходимости вручную вычислять накопленные проценты.7

### Продвинутые аспекты: Индексы и Scaled Balance

Для удаления «шума», вызванного ежесекундным начислением процентов, профессиональные трекеры отслеживают scaledBalance. В Aave V3 scaledBalance — это сумма депозита, деленная на текущий индекс ликвидности (liquidityIndex) на момент совершения транзакции.8

Если scaledBalance пользователя не меняется, значит, изменение его чистой позиции обусловлено исключительно начислением процентов, а не реальными действиями (депозитом или выводом средств). Это позволяет фильтровать сигналы, оставляя только те, которые вызваны активными операциями кита.8

## Механика Compound: Специфика версий V2 и V3

Протокол Compound прошел путь от модели множественных рынков в V2 до модели изолированных рынков с базовым активом в V3 (Comet). Для каждого поколения требуются свои методы извлечения данных.9

### Compound V2: Модель cTokens

В версии V2 каждый актив представлен своим cToken (например, cETH, cUSDC). Чистая позиция здесь рассчитывается путем запроса данных у контракта Comptroller и индивидуальных cToken.

1. **Список рынков**: Функция getAssetsIn(address account) в Comptroller возвращает массив адресов cToken, в которые вошел пользователь.11
2. **Снимок позиции**: Для каждого cToken вызывается функция getAccountSnapshot(address account). Она возвращает кортеж:
   * Ошибка (0 — успех).
   * Баланс cToken.
   * Баланс заимствования (Borrow Balance).
   * Курс обмена (Exchange Rate).11

Расчет позиции в базовом активе:

$$Collateral = cTokenBalance \times exchangeRate$$

$$Debt = borrowBalance$$

### Compound V3 (Comet): Современная архитектура

Comet кардинально меняет подход. В каждой инстанции Comet есть один «базовый актив» (например, USDC), который можно заимствовать, и набор «залоговых активов», которые можно только поставлять.9

Для извлечения данных используется структура UserBasic, получаемая через функцию userBasic(address account).14

| **Поле UserBasic** | **Описание** |
| --- | --- |
| principal | Баланс базового актива (положительный — депозит, отрицательный — долг) 14 |
| assetsIn | Битовая маска (bitmap), указывающая, какие залоговые активы внесены 15 |

Чтобы рассчитать Net Position в Compound V3, необходимо:

1. Получить principal из userBasic. Если он отрицательный, это значение $Debt\_{base}$.
2. Итерировать по битовой маске assetsIn. Для каждого установленного бита получить адрес залогового актива через getAssetInfo(uint8 id).14
3. Вызвать collateralBalanceOf(address account, address asset) для получения суммы залога по каждому активу.13

## LST-коррекция: Математика дисконта и фильтрация рыночного шума

Токены ликвидного стейкинга, такие как $stETH$, являются наиболее популярным видом залога у китов в стратегиях «плечевого стейкинга» (yield looping).17 В нормальных условиях $stETH$ торгуется почти 1:1 к $ETH$, но в моменты кризисов ликвидности (например, крах Terra или FTX) привязка может нарушаться.18

### Проблема оракулов в Aave и Compound

Большинство лендинговых протоколов используют защищенные оракулы (Chainlink) или жестко заданные коэффициенты для оценки LST.

* В Aave V3 для $stETH$ часто используется адаптер, который считает курс как 1:1 к $ETH$, доверяя безопасности протокола Lido.18
* Для $wstETH$ используется формула: $Price\_{wstETH} = Price\_{ETH} \times Ratio\_{Lido}$, где Ratio — внутренний курс обмена Lido, который только растет.20

**Риск для трекера китов:** Если рыночная цена $stETH$ на биржах (Curve, Uniswap) падает до 0.95 ETH, а оракул Aave всё еще считает её как 1.00 ETH, то реальная стоимость портфеля кита завышена на 5%. Это создает «шум»: кит может быть уже на грани ликвидации по рыночным ценам, но трекер, использующий только данные протокола, покажет, что позиция стабильна.

### Методология корректировки на основе ликвидности (LST-Correction)

Для избавления от шума и получения «истинной» стоимости позиции (Real Net Position) необходимо интегрировать данные о рыночном дисконте.

#### 1. Использование ценовых фидов Chainlink для stETH/ETH

Chainlink предоставляет прямой фид курса $stETH/ETH$ (адрес 0x86392dC15406f859677340913993073040222812 на Ethereum Mainnet).22 Этот фид отражает агрегированную цену с вторичных рынков и учитывает депег.

Алгоритм коррекции:

$$Collateral\_{adjusted} = \sum (Amount\_i \times Price\_{Oracle, i} \times DiscountFactor\_i)$$Где $DiscountFactor$ для $stETH$ рассчитывается как:

$$DiscountFactor\_{stETH} = \frac{Price\_{stETH/ETH}^{Chainlink}}{1.0}$$

Если значение фида составляет 0.998, мы применяем этот коэффициент к залоговой части портфеля.

#### 2. Оценка через глубину ликвидности Curve

Для сверхкрупных позиций («мега-китов») даже цена Chainlink может быть нерелевантной, так как при попытке выхода из позиции они вызовут значительное проскальзывание (slippage). Крупнейшим пулом ликвидности для $stETH$ является пул StableSwap на Curve.23

Математика Curve использует инвариант StableSwap, который зависит от коэффициента усиления (A-factor).25

| **Фактор ликвидности** | **Значение** | **Влияние на Net Position** |
| --- | --- | --- |
| **A-factor** | 50-200 | Определяет «плоскость» кривой. Чем выше A, тем меньше проскальзывание у китов 23 |
| **Коэффициент резервов** | % ETH в пуле | Если доля ETH падает ниже 20%, выход кита из stETH вызовет обвал цены 23 |

Трекер должен запрашивать через Multicall функцию get\_dy(int128 i, int128 j, uint256 dx) у контракта пула Curve.27 Это позволит рассчитать реальную сумму $ETH$, которую получит адрес при немедленной продаже своего залога.

Формула реальной чистой позиции с учетом проскальзывания:

$$NetPosition\_{Real} = get\\_dy(stETH, ETH, Collateral\_{stETH}) - Debt\_{ETH}$$

## Устранение шума в сигналах: Статистический подход

Для того чтобы отличить временную волатильность (шум) от реального изменения тренда, необходимо применять статистические фильтры к дисконту LST. Исследования Chaos Labs показывают, что пары $stETH/ETH$ обладают свойствами возврата к среднему (mean-reversion), но скорость этого возврата может варьироваться.28

### Тесты на возврат к среднему (ADF и Hurst)

Для фильтрации ложных сигналов депега рекомендуется использовать два показателя:

1. **Тест Дики-Фуллера (ADF)**: Если значение ADF отрицательное и значимое, привязка стабильна, и текущее отклонение — это шум.28
2. **Показатель Херста (Hurst Exponent)**: Значение $H < 0.5$ подтверждает, что цена стремится вернуться к паритету.28

**Практическое применение в трекере:**

* Если $stETH/ETH < 0.99$, но тесты подтверждают сильный возврат к среднему, сигнал о «выходе кита» помечается как «низкоприоритетный шум».
* Если дисконт сохраняется более 24 часов ($\tau > 24h$) и корреляция падает, это сигнал о реальном изменении рыночной стоимости портфеля.28

### Сравнение параметров риска E-Mode

Aave V3 использует режим эффективности (E-Mode) для активов с высокой корреляцией, таких как LST и ETH. Это позволяет китам использовать LTV до 90% и выше.7

| **Параметр** | **Обычный режим** | **E-Mode (ETH Correlated)** |
| --- | --- | --- |
| **Max LTV** | ~70-80% | 90% + 7 |
| **Liquidation Threshold** | ~85% | 93-95% 7 |
| **Liquidation Penalty** | 5% | 1% 7 |

Высокий LTV в E-Mode делает позиции китов крайне чувствительными к малейшим изменениям дисконта. Шум в 0.5% при плече 10x превращается в 5% изменение маржинального обеспечения. Именно поэтому LST-коррекция через Multicall обязательна для точного мониторинга.17

## Практическая реализация: Пошаговый алгоритм для трекера

Для построения системы, свободной от шума, предлагается следующая архитектурная схема запросов и обработки данных:

### Шаг 1: Сбор сырых данных через Multicall3

Единый запрос aggregate3 должен включать:

* AaveV3.Pool.getUserAccountData(whale\_address) — для быстрой проверки здоровья.
* AaveV3.DataProvider.getUserReserveData(stETH, whale\_address) — для детального баланса.
* CompoundV3.Comet.userBasic(whale\_address) — для проверки долга в USDC/ETH.
* Chainlink.stETH\_ETH\_Feed.latestRoundData() — для получения текущего рыночного курса.
* Curve.stETH\_Pool.balances(0) и balances(1) — для оценки баланса пула.1

### Шаг 2: Расчет теоретической и реальной позиции

На бэкенде трекера вычисляются два значения:

1. **Protocol Net Position**: На основе цен оракулов протокола (где 1 stETH = 1 ETH).
2. **Adjusted Net Position**: С применением дисконта Chainlink и проскальзывания Curve.

### Шаг 3: Фильтрация сигналов

Сигнал считается валидным («сигнал»), если:

* Изменение scaledBalance в Aave составило $> 1\%$ (кит совершил транзакцию).8
* Или Adjusted Health Factor опустился ниже 1.05, в то время как Protocol Health Factor остается стабильным (скрытый риск ликвидации из-за депега).

## Заключение: Интеграция данных для принятия решений

Эффективный трекер китов должен объединять техническую точность извлечения данных (через Multicall3) с глубоким пониманием экономики протоколов и рисков ликвидности. Чистая позиция — это не статичная цифра, а динамический показатель, который в случае с LST-активами требует постоянной калибровки. Использование жестких оракулов протокола дает чистый, но часто ложный сигнал, в то время как учет рыночного дисконта и глубины пулов Curve позволяет увидеть реальные риски и намерения крупных игроков. Внедрение предложенной методологии позволит избавиться от шума, вызванного краткосрочной волатильностью LST, и обеспечит высокую точность прогнозирования действий «китов» в условиях нестабильности DeFi-рынка.

#### Works cited

1. How to Optimize Ethereum RPC Usage with Multicall | Quicknode Guides, accessed on January 19, 2026, <https://www.quicknode.com/guides/ethereum-development/transactions/how-to-optimize-ethereum-rpc-usage-with-multicall>
2. mds1/multicall3: Multicall: Aggregate multiple constant function call results into one - GitHub, accessed on January 19, 2026, <https://github.com/mds1/multicall3>
3. Multicall3 | Berachain Core Docs, accessed on January 19, 2026, <https://docs.berachain.com/developers/contracts/multicall3>
4. Multicall - Filecoin Docs, accessed on January 19, 2026, <https://docs.filecoin.io/smart-contracts/advanced/multicall>
5. Aave: Protocol Data Provider V3 | Address: 0x497a1994 ... - Etherscan, accessed on January 19, 2026, <https://etherscan.io/address/0x497a1994c46d4f6C864904A9f1fac6328Cb7C8a6>
6. Pool | Aave Protocol Documentation, accessed on January 19, 2026, <https://aave.com/docs/aave-v3/aptos/smart-contracts/pool>
7. Aave V3 Overview | Aave Protocol Documentation, accessed on January 19, 2026, <https://aave.com/docs/aave-v3/overview>
8. Specification - Lido Docs, accessed on January 19, 2026, <https://docs.lido.fi/integrations/aave/specification/>
9. Compound III Documentation, accessed on January 19, 2026, <https://docs.compound.finance/>
10. What is Compound V2? Complete Guide to Decentralized Lending - Nansen, accessed on January 19, 2026, <https://www.nansen.ai/post/what-is-compound-v2>
11. Comptroller - Compound v2 Docs, accessed on January 19, 2026, <https://docs.compound.finance/v2/comptroller/>
12. Compound v2 Docs | cTokens, accessed on January 19, 2026, <https://docs.compound.finance/v2/ctokens/>
13. Collateral & Borrowing - Compound III Docs, accessed on January 19, 2026, <https://docs.compound.finance/collateral-and-borrowing/>
14. Helper Functions - Compound III Docs, accessed on January 19, 2026, <https://docs.compound.finance/helper-functions/>
15. Understanding Collateral, Liquidations, and Reserves in Compound V3 | By RareSkills, accessed on January 19, 2026, <https://rareskills.io/post/compound-finance-liquidation>
16. Comet - Compound.js Docs, accessed on January 19, 2026, <https://docs.compound.finance/compound-js/comet/>
17. Aave v3 stETH | Summer.fi Knowledge Base, accessed on January 19, 2026, <https://docs.summer.fi/summer.fi-pro/products/earn/aave-v3-steth>
18. aave-proposals/src/AaveV2-V3PriceFeedsUpdate\_20230613/PRICE-FEEDS-UPDATE-20230613.md at main - GitHub, accessed on January 19, 2026, <https://github.com/bgd-labs/aave-proposals/blob/main/src/AaveV2-V3PriceFeedsUpdate_20230613/PRICE-FEEDS-UPDATE-20230613.md>
19. A Deep Dive into the Leading AMM Platform: Curve Finance - Mint Ventures, accessed on January 19, 2026, <https://research.mintventures.fund/2023/04/10/a-deep-dive-into-the-leading-amm-platform-curve-finance/>
20. Exchange rate for stETH/ETH hardcoded? - Learning Center - Aave, accessed on January 19, 2026, <https://governance.aave.com/t/exchange-rate-for-steth-eth-hardcoded/22693>
21. BGD. Correlated-asset price oracle - Development - Aave, accessed on January 19, 2026, <https://governance.aave.com/t/bgd-correlated-asset-price-oracle/16133>
22. STETH / ETH Price Feed | Chainlink, accessed on January 19, 2026, <https://data.chain.link/ethereum/mainnet/crypto-eth/steth-eth>
23. Recommendation for changing the Curve stETH:ETH pool A Factor from 30→200, accessed on January 19, 2026, <https://research.lido.fi/t/recommendation-for-changing-the-curve-steth-eth-pool-a-factor-from-30-200/6672>
24. [Finance] Dynamic Peg #1 — Overview of the stETH Crisis: Causes and Impacts - A41.io, accessed on January 19, 2026, <https://medium.a41.io/dynamic-peg-1-overview-of-the-steth-crisis-causes-and-impacts-4b171f2cf4da>
25. Cryptoswap: In Depth - Curve Finance Docs, accessed on January 19, 2026, <https://docs.curve.finance/cryptoswap-exchange/in-depth/>
26. Deep dive into Curve Finance: Core Mechanics, Security, and Integration Insights - Zealynx, accessed on January 19, 2026, <https://www.zealynx.io/blogs/curve-finance-core-mechanics>
27. get\_D() and get\_y() in Curve StableSwap | By RareSkills, accessed on January 19, 2026, <https://rareskills.io/post/curve-get-d-get-y>
28. On Aave's E-mode Classes and their Risk Parameters - Chaos Labs, accessed on January 19, 2026, <https://chaoslabs.xyz/resources/chaos_aave_e_mode.pdf>