



# SOR Calculation Memory

Exploring the Calculation Memory: A Comprehensive Guide



Last Update: 23/11/2023



This guide aims to explain how the SOR (Smart Order Router) calculates the price of a quote, a must have knowledge to understand the Calculation Memory, a new feature set to launch on the release 2.24.

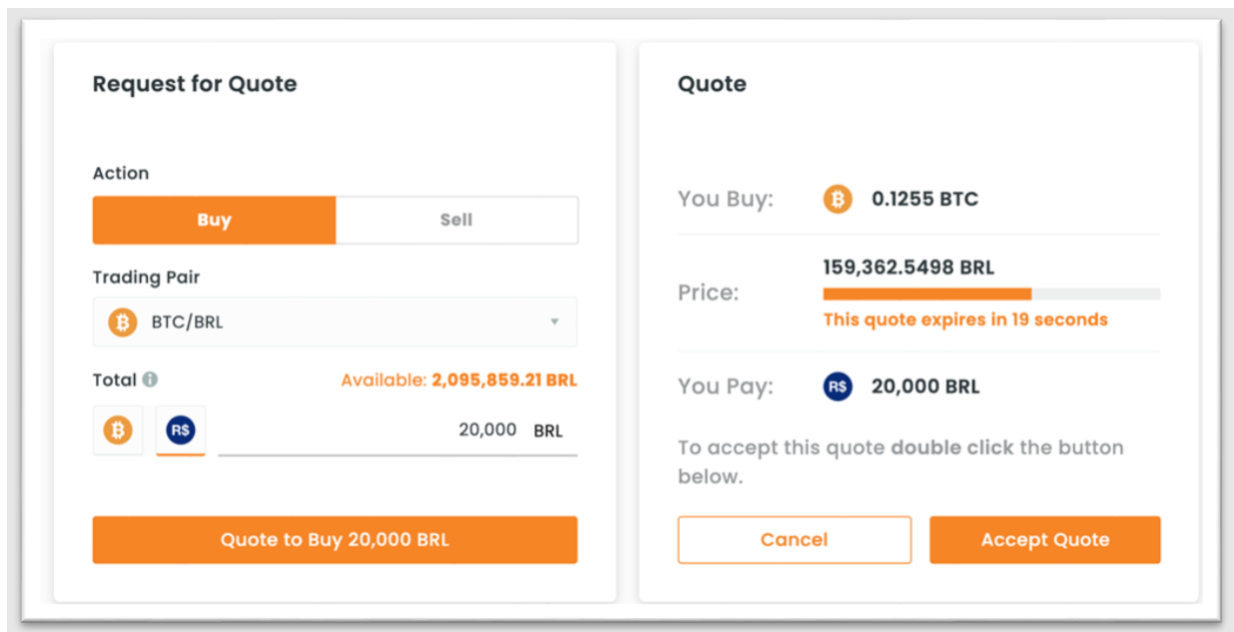
**DISCLAIMER:** Due to the technical nature of Parfin's SOR algorithm, a full exploration of all its aspects is beyond the scope of this article. This article simplified the algorithm down to its essential elements, offering enough clarity to understand its foundational logic.

## 1. THE RFQ MECHANISM

By using Parfin's RFQ (Request for Quote) system, users can submit RFQs to a wide network of liquidity providers. Upon receiving the best quote price, users can either accept or reject it at their discretion.

In the context of trading, an RFQ is a mechanism where the user specifies the details of the trade they're interested in. For example, if a user wants to buy a certain amount of BTC, they can initiate an RFQ indicating the quantity of BTC they wish to buy. Once the RFQ is created, it is typically sent to various counterparties, like CEXs (Centralized Exchanges) and OTCs (Over-The-Counter). These counterparties will then respond how much they would charge for fulfilling the user's request.

What sets these quotes apart is their locked-in nature, providing users with a guaranteed rate (referred as the **Quote Price**) for accepting the trade within a certain time frame. For example, users may have the option to buy BTC at a set price of 159,362 BRL, with the price held steady for a quick 30-seconds window. The image below illustrates this process.



The screenshot displays the Parfin RFQ interface, divided into two main sections: 'Request for Quote' and 'Quote'.

**Request for Quote Section:**

- Action:** A toggle switch with 'Buy' (selected, orange) and 'Sell' (grey) options.
- Trading Pair:** A dropdown menu showing 'BTC/BRL' with a Bitcoin icon.
- Total:** A section with a Bitcoin icon and a BRL icon. The BRL icon is selected, and the amount '20,000 BRL' is entered. Above this, it says 'Available: 2,095,859.21 BRL'.
- Quote Button:** An orange button labeled 'Quote to Buy 20,000 BRL'.

**Quote Section:**

- You Buy:** 0.1255 BTC (with a Bitcoin icon).
- Price:** 159,362.5498 BRL. Below the price is a progress bar and the text 'This quote expires in 19 seconds'.
- You Pay:** 20,000 BRL (with a BRL icon).
- Instructions:** 'To accept this quote double click the button below.'
- Buttons:** Two orange buttons: 'Cancel' and 'Accept Quote'.

To initiate an RFQ, a user needs to provide the following details:

<b>Trading Pair</b>	The specific asset pair intended for trade (e.g. BTC/BRL)
<b>Side</b>	The direction of the trade, either buy or sell
<b>Input Type</b>	The basis for the transaction, either total or quantity
<b>Total / Quantity</b>	The transaction amount, in either fiat or crypto

For the user, the RFQ process may appear as an enigmatic black box. The user provides the quote details and, almost as if by magic, receives the best quote price. The key to this seemingly magical process lies in Parfin's SOR algorithm.

## 2. THE SOR ALGORITHM

To demystify the SOR, this section will delve into the mechanics of the SOR's algorithm.

⚠ The complexity and technical nuances of the SOR's algorithm preclude a comprehensive discussion of all its facets. However, the ensuing discussion will distill the algorithm into its core principles, providing sufficient insight into the underlying logic.

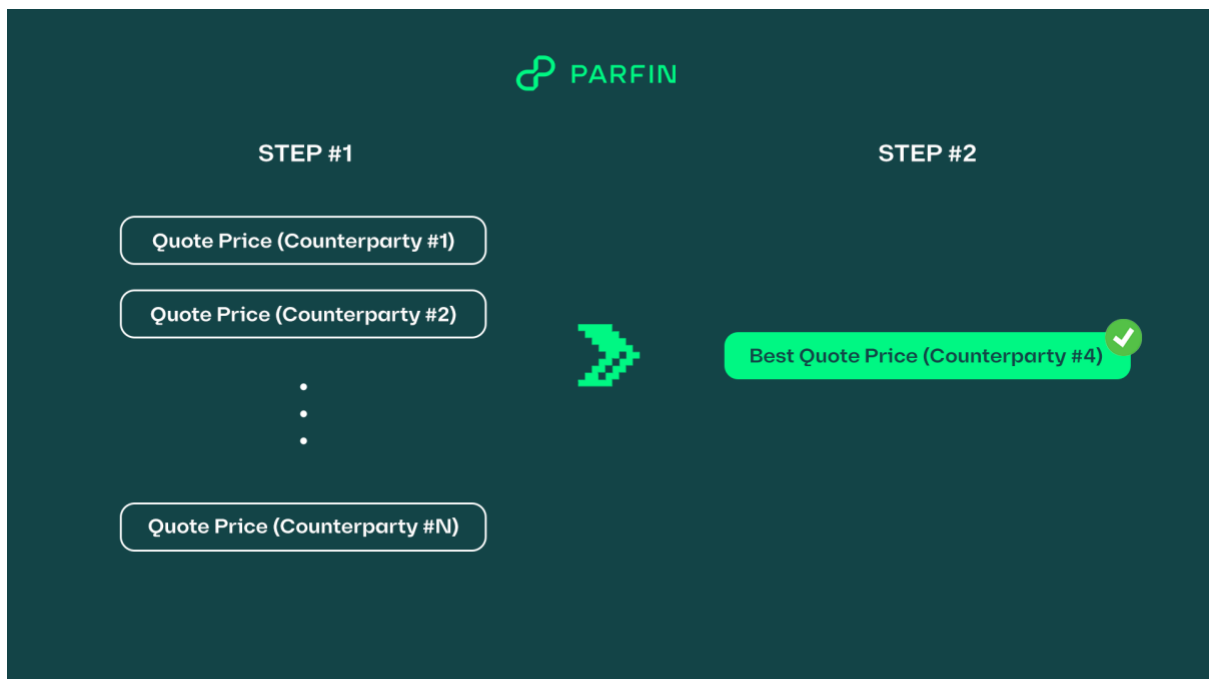
In simplified terms, the sole purpose of the SOR is to efficiently scan the complex crypto market to find the best quote price given a specific set of inputs provided by the user.

To achieve this:

**[STEP #1]** The algorithm first calculates the quote price for each counterparty that is connected to the RFQ Execution Settings.

**[STEP #2]** Once individual quote prices are established on a one-to-one basis with each counterparty, the algorithm proceeds to perform a comparative analysis to determine the best quote price.

Hence, as illustrated on the image below, the algorithm narrows down from a pool of quote prices to a single one, deemed the best.



To understand how the SOR calculates each quote price in **STEP #1**, it is necessary to understand the components that “form” the price of a quote. Each component is detailed in the subsequent subsections.

💡 Think of each component as a Lego brick; assembling them together enables the determination of the quote price.

## 2.1. ESTIMATED TRADE CLEAN PRICE

As previously stated, the SOR algorithm is tasked with identifying the best quote price based on:

- Prevailing market conditions
- The connected counterparties
- The inputs provided by the user

To achieve this objective, the algorithm must assess each counterparty connected to the SOR. Put simply, the SOR algorithm calculates a quote price for each *eligible* counterparty.

In line with this objective, once the user provides the necessary inputs for the RFQ, the SOR algorithm begins its operation by fetching the **Estimated Trade Clean Price** for each available counterparty.

⚠ The term “fetching” simplifies a multi-step procedure replete with calculations, all aimed at arriving at the Estimated Trade Clean Price for each available counterparty.

The **Estimated Trade Clean Price** represents the estimated price that the counterparty would charge for the trade.

At this point, three observations merit attention:

**[1]** The term “clean” is used to denote that the price does not incorporate any trading fees charged by the counterparty.

**[2]** The term “estimated” indicates that this is not the execution price for the trade against the counterparty.

**[3]** A counterparty is considered *available* if it is properly connected to the SOR. However, *availability* does not necessarily equate to *eligibility* for participation in the RFQ process. A counterparty may be excluded (also known as “ruled out”) for several reasons, the most common of which include:

- Trading pair not supported by the counterparty
- Insufficient funds available for the trade at the counterparty
- Decimal precision inputted exceeds the counterparty's capabilities

Following this process of ruling out certain counterparties, the SOR algorithm consolidates all the **Estimated Trade Clean Prices** of the counterparties that are *available* and *eligible*.

## 2.2. ESTIMATED TRADE PRICE

Subsequently, the algorithm proceeds to calculate the **Estimated Trade Price**, commonly referred to as the Estimated Trade *Dirty* Price. This price incorporates the trading fees that would be charged by the counterparty.

The determination of these trading fees for each counterparty relies on historical data and undergoes dynamic updates as new trading activities occur.

The following formulas are used for the calculations:

$\text{Trade Fee Price} = \text{Fee Rate} * \text{Estimated Trade Clean Price}$
$\text{Estimated Trade Price} = \text{Estimated Trade Clean Price} + \text{Trade Fee Price}$

The formulas outlined above are based on a "buy" RFQ. For a "sell" RFQ, the **Estimated Trade Clean Price** is reduced by the **Trade Fee Price** to determine the Trade Price.

At this point of the explanation, the SOR:

1	Fetches the <b>Estimated Trade Clean Price</b>
2	Calculated the <b>Trade Fee Price</b>
3	Calculated the <b>Estimated Trade Price</b> by adding or subtracting the <b>Trade Fee Price</b>

## 2.3. FX CONVERSION

Until now, the discussion has centred on counterparties that can directly execute trades. However, it is common to deal with a foreign counterparty – one that operates in a *quote currency* different from the one the user is allowed to trade.



For example, counterparty X may offer a BTC/**USD** trading pair, while the user is restricted to trading BTC/**BRL**.

Due to Parfin's SOR algorithm, executing such a trade remains feasible for the user. However, the incorporation of a foreign exchange (FX) conversion becomes necessary because the counterparty and the user (through your company) operate in different quote currencies.

Thus, when FX conversion is necessary, the SOR algorithm retrieves the **Estimated FX Clean Price** from a FX Provider, which could be either a financial institution or a market data provider. As explained before, the term "estimated" indicates that this is not the FX execution price.

Subsequently, the algorithm proceeds to calculate the **Estimated FX Price**, commonly referred to as the FX *Dirty* Price. This price incorporates taxes and any additional spread.



Typically, taxes refer to taxes such as the IOF. The additional spread refers to the spread that is applied when the FX Provider is unavailable, and the **FX Clean Price** must be sourced from a public market data provider. This additional spread is incorporated into the operation to mitigate associated risks.

The following formulas are used when FX conversion is necessary:

$FX\ Taxes\ Price = FX\ Taxes\ \% * Estimated\ FX\ Clean\ Price$
$FX\ Offline\ Spread\ Price = FX\ Offline\ Spread\ \% * Estimated\ FX\ Clean\ Price$
$Estimated\ FX\ Price = Estimated\ FX\ Clean\ Price + FX\ Taxes\ Price + FX\ Offline\ Spread\ Price$

The formulas outlined above are based on a “buy” RFQ. For a “sell” RFQ, the **Estimated FX Clean Price** is reduced by the **FX Taxes Price** and the **FX Additional Spread Price** to determine the **Estimated FX Price**.

At this point of the explanation, the SOR:

1	Fetches the <b>Estimated Trade Clean Price</b>
2	Calculated the <b>Trade Fee Price</b>
3	Calculated the <b>Estimated Trade Price</b> by adding or subtracting the <b>Trade Fee Price</b>
4	Fetches the <b>Estimated FX Clean Price</b> (if necessary)
5	Calculated the <b>FX Taxes Price</b> (if necessary)
6	Calculated the <b>FX Additional Spread Price</b> (if necessary)
7	Calculated the <b>Estimated FX Price</b> (if necessary)

## 2.4. QUOTE PRICE WITHOUT SPREAD

Upon estimating the **Estimated Trade Price** and the **Estimated FX Price**, the SOR proceeds to calculate the **Quote Price Without Spread**.

Basically, it is the **Estimated Trade Price** converted to the company’s quote currency using the **Estimated FX Price**. As the name suggests, the **Quote Price Without Spread** excludes the spread that your company may charge your users.

It is derived by multiplying the **Estimated Trade Price** with the **Estimated FX Price**:

$Quote\ Price\ Without\ Spread = Estimated\ Trade\ Price * Estimated\ FX\ Price$
--

If an FX conversion is not necessary:

$Quote\ Price\ Without\ Spread = Estimated\ Trade\ Price$
---

## 2.5. SPREAD PRICE

Next, the SOR calculates the **Spread Price**. This price component functions akin to brokerage fees and reflects either a markup or markdown on the **Quote Price Without Spread**, depending on whether the quote is a buy or a sell.

The formula to calculate the **Spread Price** also depends on the side of the quote – whether it is a buy or a sell quote.

For buy:

$$\text{Spread Price} = \text{Spread \%} * (\text{Estimated Trade Clean Price} * \text{Estimated FX Price})$$

For sell:

$$\text{Spread Price} = \left( \left( \frac{\text{Estimated Trade Clean Price}}{1 + \text{Spread \%}} \right) - \text{Trade Clean Price} \right) * \text{Estimated FX Price} * (-1)$$

It should be noted that the **Spread Price** is calculated based on the **Estimated Trade Clean Price**. This methodology is deliberately taken to avoid calculating spread on top of trading fees.

## 2.6. UNADJUSTED QUOTE PRICE

To calculate the **Unadjusted Quote Price**, the SOR will either markup or markdown the **Quote Price Without Spread** with the **Spread Price**.

Whether it's a markup or markdown depends on the side of the quote. For buy, the **Spread Price** will be a markup, while for sell, it will be a markdown.

For buy:

$$\text{Unadjusted Quote Price} = \text{Quote Price Without Spread} + \text{Spread Price}$$

For sell:

$$\text{Unadjusted Quote Price} = \text{Quote Price Without Spread} - \text{Spread Price}$$

One may wonder why this price is referred to as “unadjusted.” Subsequent sections will provide a detailed explanation.

At this point of the explanation, the SOR:

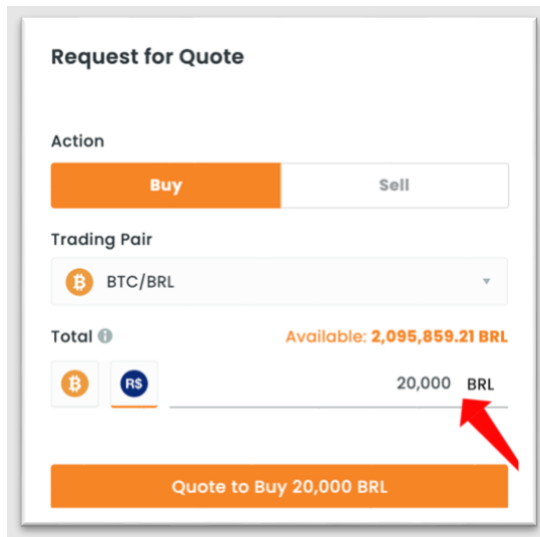
1	Fetches the <b>Estimated Trade Clean Price</b>
2	Calculated the <b>Trade Fee Price</b>



3	Calculated the <b>Estimated Trade Price</b> by adding or subtracting the <b>Trade Fee Price</b>
4	Fetches the <b>Estimated FX Clean Price</b> (if necessary)
5	Calculated the <b>FX Taxes Price</b> (if necessary)
6	Calculated the <b>FX Additional Spread Price</b> (if necessary)
7	Calculated the <b>Estimated FX Price</b> (if necessary)
8	Calculated the <b>Quote Price Without Spread</b>
9	Calculated the <b>Spread Price</b>
10	Calculated the <b>Unadjusted Quote Price</b>

## 2.7. UNADJUSTED QUANTITY

For this subsection, let's consider the RFQ shown on the image below.



The inputs of the RFQ are as follow:

<b>Trading Pair</b>	BTC/BRL
<b>Side</b>	Buy
<b>Input Type</b>	Total
<b>Total</b>	20,000 BRL

Now, based on this immutable equation:

$$Total = Price * Quantity$$

The SOR can derive the **Unadjusted Quantity** rearranging the equation:

$$\text{Unadjusted Quantity} = \frac{\text{Total}}{\text{Unadjusted Quote Price}}$$

Let's say, hypothetically, that the **Unadjusted Quote Price** is 159,362 BRL. Then:

$$\text{Unadjusted Quantity} = \frac{20,000}{159,362} = 0.125500432 \dots$$

In this scenario, the **Unadjusted Quantity** refers to the quantity that the user would receive for the trade, considering the expenditure of 20,000 BRL and the prevailing **Unadjusted Quote Price** of 159,362 BTC/BRL.

As noted from this example, when the input type is Total, the SOR has *initially* no control over the number of decimal places of the **Unadjusted Quantity**.

However, each counterparty has unique rules regarding decimal precision. When the SOR calculates the **Unadjusted Quantity**, this value may not necessarily align with the counterparty's requirements. To prevent the counterparty from rejecting the RFQ, adjustments to the quantity are essential.

For instance, consider that Counterparty A stipulates a maximum of 4 decimal places for BTC. If the SOR calculates an **Unadjusted Quantity** exceeding this limit — for example, 0.1255004321239... BTC — it becomes evident that an adjustment is necessary. These modifications ensure that Counterparty A is able to accept the quote.

After clarifying these concepts, one can better understand why certain calculations are termed "unadjusted."

## 2.8. ADJUSTED QUANTITY

As previously outlined, the **Unadjusted Quantity** must undergo adjustments in decimal places to be in compliance with each counterparty's decimal precision requirements. Therefore, after such adjustments, the SOR arrives at the **Adjusted Quantity**. In this manner, the **Adjusted Quantity** is the quantity that complies with the decimal precision rule of the counterparty. If no adjustments were made, the counterparty would not accept the RFQ.

It is important to understand that not all unadjusted quantities will undergo identical adjustments. This is true because:

**[1]** Each counterparty imposes unique decimal precision rules, which can further vary depending on the specific asset involved.

**[2]** Parfin's SOR algorithm employs different methodologies to adjust the **Unadjusted Quantity**, depending on whether the quote involves a buy or a sell operation.

In the case of a buy, the quantity is *truncated* to align with the counterparty's stipulated decimal precision.



To illustrate:

<b>Unadjusted Quantity</b>	0.123456789 BTC
<b>Counterparty Decimal Precision for BTC</b>	6
<b>Adjusted Quantity</b>	0.123456 BTC

In the event of a sell, the quantity undergoes a *ceil* transformation to conform to the decimal precision stipulated by the counterparty.

To illustrate:

<b>Unadjusted Quantity</b>	0.123456789 BTC
<b>Counterparty Decimal Precision for BTC</b>	6
<b>Adjusted Quantity</b>	0.123457 BTC

Simultaneously, Parfin's RFQ system can handle RFQs with Quantity as the input type. In such scenarios, the SOR doesn't need to calculate the quote's quantity, as it has already been provided by the user.

Furthermore, in such cases, the system is programmed to limit the number of decimal places a user can enter, ensuring compatibility with the specific requirements of the connected counterparties.

As a result, the term "unadjusted" is rendered irrelevant when the input type is Quantity. Thus, when the input type is Quantity, the **Unadjusted Quantity** directly becomes the **Adjusted Quantity**, or simply the final quantity of the quote. Likewise, the **Unadjusted Quote Price** also directly becomes the **Final Quote Price** since no adjustments are necessary.

## 2.9. QUOTE PRICE

As we approach the concluding phase in determining the **Final Quote Price**, let's summarize what the SOR has done so far:

1	Fetches the <b>Estimated Trade Clean Price</b>
2	Calculates the <b>Trade Fee Price</b>
3	Calculates the <b>Estimated Trade Price</b> by adding or subtracting the <b>Trade Fee Price</b>
4	Fetches the <b>Estimated FX Clean Price</b> (if necessary)
5	Calculates the <b>FX Taxes Price</b> (if necessary)
6	Calculates <b>FX Additional Spread Price</b> (if necessary)
7	Calculates the <b>Estimated FX Price</b> (if necessary)
8	Calculates the <b>Quote Price Without Spread</b>

9	Calculated the <b>Spread Price</b>
10	Calculated the <b>Unadjusted Quote Price</b>
11	Calculated <b>Unadjusted Quantity</b>
12	Calculated <b>Adjusted Quantity</b>

Then, to calculate the **Final Quote Price**, we must revisit this immutable equation:

$$Total = Price * Quantity$$

The SOR already has the Total, which was provided by the user when creating the RFQ. In addition, the SOR calculated the **Adjusted Quantity**.

Thus, the **Final Quote Price** can be determined using the following formula:

$$Final\ Quote\ Price = \frac{Total}{Adjusted\ Quantity}$$

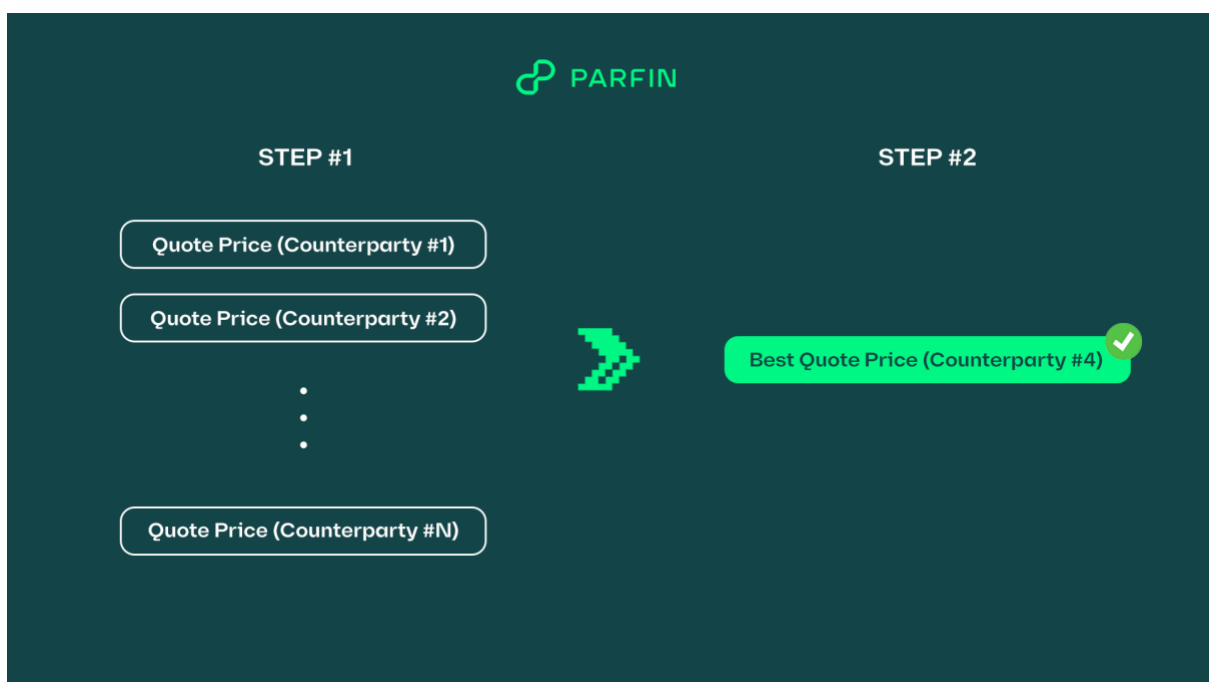
The **Final Quote Price** calculated using the formula above stands as the definitive price that would be presented to the user, should the SOR ascertain that this particular counterparty offers the best price.

### 3. DETERMINING THE BEST QUOTE PRICE

In a practical setting, a company could have multiple counterparties connected to the SOR, essentially “competing” against each other to offer the price.

For a "buy" RFQ, the counterparty offering the lowest quote price is deemed the best. Conversely, in a "sell" RFQ, the counterparty with the highest quote price takes that designation.

With that in mind, for a single RFQ, the SOR ultimate task is to pick the “winning” counterparty. To achieve this objective, the algorithm performs the calculations outlined in **STEP #1** for each counterparty that is connected to the SOR. This results in a compiled list of **Final Quote Prices**, each corresponding to a specific counterparty.

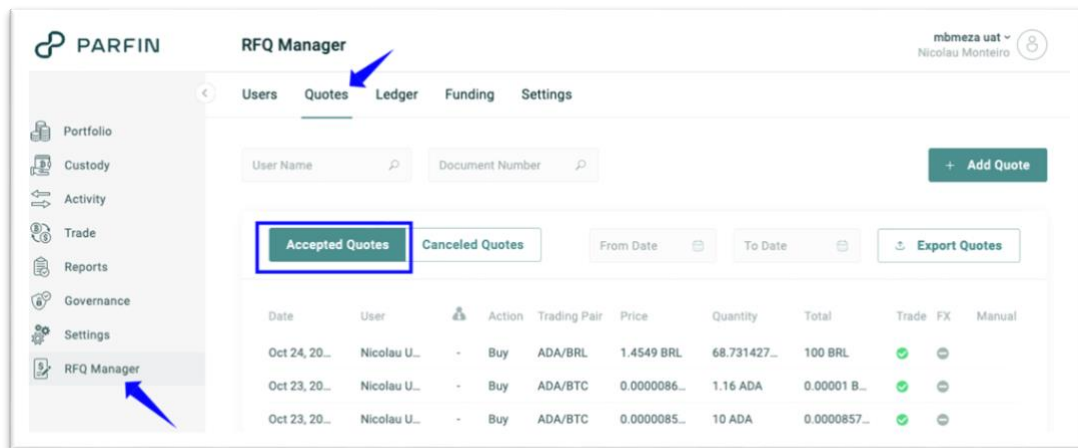


**STEP #2** is the concluding step, involving a comparative analysis of these Quote Prices to identify the counterparty that provides the best price. This selected price is then presented to the user upon the RFQ submission, leaving them the choice to accept or decline the quote.

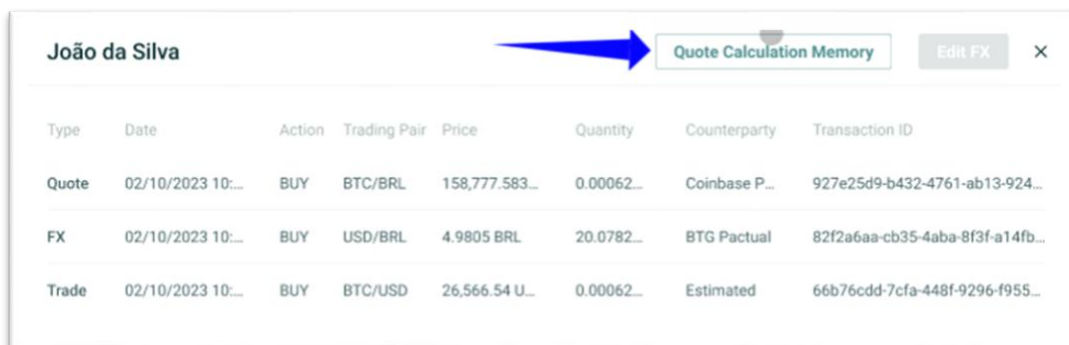
## 4. CALCULATION MEMORY – HOW TO USE IT

To utilize the Calculation Memory feature, follow these steps:

1. Navigate to the Parfin Platform
2. Access the RFQ Manager
3. Select the Quotes tab




4. Search for the specific quote on the "Accepted Quotes" table and click on its entry
5. After the modal appears (similar to the one below), click on "SOR Calculation Memory"



After clicking on "SOR Calculation Memory", the Calculation Memory section is displayed. This section is divided into four parts:

### [1] General Details

This part provides basic details about the quote, including the creation date (in UTC), the user's name, and the Quote ID.

General Details			
Date	User		Quote ID
02/10/2023 10:00:00	João da Silva	-	927e25d9-b432-4761-ab13-924bc7c0b3d4

## [2] Quote Details


This part offers key information about the quote, encompassing details such as the trading pair, the trading side (Buy or Sell), the input type (Total or Quantity), the asset quantity, and the total value of the quote.

Quote Details					
Trading Pair	Side	Input Type	Price	Quantity	Total
BTC/BRL	Buy	Total	158,777.5830 BRL	0.00062981182 BTC	100,00 BRL

## [3] Best Execution

This part presents a comprehensive list of all the counterparties that were *available* for the RFQ. It shows the name of the counterparty, the final quote price that was calculated by the SOR, and a flag indicating the counterparty with the best price.

In cases where a counterparty is *available* but deemed *ineligible*, the exclusion reason is displayed instead of the final quote price.

Best Execution		
Counterparty	Price	Best Price
Coinbase Prime	158,777.5830 BRL	
Foxbit	160,035.0098 BRL	-
Kraken	Counterparty response exceeded your RFQ timeout setting.	-


To check how each **Final Quote Price** was calculated, one can simply click on the desired entry and a modal with the components that "form" the **Final Quote Price** will be displayed.

This modal will contain all the components of the **Final Quote Price** that has been explained on the previous sections of this guide.

Price Details

Estimated Trade Prices

Counterparty

 B2C2


Trade Clean Price

1.0008 USD

Trade Fee Price


0 USD

Trade Price

 1.0008 USD

Estimated FX Prices

FX Provider

 FX Market Data Provider

FX Clean Price

4.8943 BRL


FX Taxes Price\*

0.0185 BRL (0.38%)

FX Offline Spread Price\*

0.0489 BRL (1.00%)

FX Price

 4.9618 BRL

Unadjusted Quote


Quote Price Without Spread

4.9658 BRL

Spread Price

0.0993 BRL (2.00%)

Unadjusted Quote Price

 5.0651 BRL

Adjusted Quote

Unadjusted Quantity

9.87142073840148 USDT

Counterparty Decimal Precision


5 (0.00000)

Adjusted Quantity

9.87142 USDT

Best Price

Final Quote Price

 5.0651 BRL

\*price per 1 USD

💡 Note that the FX Taxes %, FX Offline Spread % and the Spread % will be displayed where indicated on the image above.



## 5. CALCULATION MEMORY – PRACTICAL EXAMPLE

Consider the following RFQ:

<b>Side</b>	Buy
<b>Trading Pair</b>	ADA/BRL
<b>Input Type</b>	Total
<b>Total</b>	200 BRL

### Request for Quote

Action

Buy

Sell

Trading Pair

ADA/BRL

Total ⓘ Available: **2,095,359.21 BRL**

ADA

BRL

200 BRL

Quote to Buy 200 BRL

### Accepted Quote

You have successfully executed the following trade

You Buy: 121.6 ADA

---

Price: 1.6447 BRL

---

You Pay: 200 BRL

Your balances have been automatically updated. You may request a withdrawal of funds to a whitelisted address once the outstanding amounts are settled.

The SOR algorithm returned that the best quote price is **1.6447 BRL** which yield a quantity of **121.60 ADA**.

Now, with the Calculation Memory feature, you can discern the rationale for determining **1.6447 BRL** as the best price, based on the inputs provided in the quote.

When the quote was created, these were the configurations on the RFQ Manager:

<b>Spread</b>	3.00%
<b>FX Taxes</b>	0.38%
<b>FX Offline Spread</b>	0.00%

By checking at the SOR Calculation Memory, the **Estimated Trade Clean Price** is 0.283 USD, and the **Trade Fee Price** is 0.0004245 USD.

Then:

$$\text{Estimated Trade Price} = \text{Estimated Trade Clean Price} + \text{Trade Fee Price}$$

$$\text{Estimated Trade Price} = 0.283 + 0.0004245 = 0.2834245$$

Since the currency of the quote is BRL and the currency of the trade is USD, it is necessary to perform an FX conversion.

The SOR Calculation Memory indicates that the **Estimated FX Clean Price** is 5.6127 BRL.

Then:

$$\text{FX Taxes Price} = \text{FX Taxes \%} * \text{Estimated FX Clean Price}$$

$$\text{FX Taxes Price} = 0.38 \% * 5.6127 = 0.02132826$$

$$\text{FX Additional Spread Price} = \text{FX Additional Spread \%} * \text{FX Clean Price}$$

$$\text{FX Additional Spread Price} = 0 \% * 5.6127 = 0$$

$$\text{Estimated FX Price} = \text{FX Clean Price} + \text{FX Taxes Price} + \text{FX Additional Spread Price}$$

$$\text{Estimated FX Price} = 5.6127 + 0.02132826 + 0 = 5.63402826$$

The next step is to calculate the **Quote Price Without Spread**:

$$\text{Quote Price Without Spread} = \text{Estimated Trade Price} * \text{Estimated FX Price}$$

$$\text{Quote Price Without Spread} = 0.2834245 * 5.63402826 = 1.59682164257637$$

Since the side of the quote is *buy*, the formula to calculate the **Spread Price** is as follow:

$$\text{Spread Price} = \text{Spread \%} * (\text{Estimated Trade Clean Price} * \text{Estimated FX Price})$$

$$\text{Spread Price} = 3 \% * (0.283 * 5.63402826) = 0.0478328999274$$

Now, it is possible to arrive at the **Unadjusted Quote Price**:

$$\text{Unadjusted Quote Price} = \text{Quote Price Without Spread} + \text{Spread Price}$$

$$\text{Unadjusted Quote Price} = 1.59682164257637 + 0.0478328999274 = 1.64465454250377$$

As previously mentioned, for this quote, the user wants to buy an equivalent of 200 BRL (Total) of ADA. So, the **Unadjusted Quantity** is:

$$\text{Unadjusted Quantity} = \frac{\text{Total}}{\text{Unadjusted Quote Price}}$$

$$\text{Unadjusted Quantity} = \frac{200}{1.64465454250377} = 121.60608494445 \dots$$

According to the SOR Calculation Memory, the counterparty stipulates a decimal precision of 2 decimal places. Hence, it is necessary to *truncate* the **Unadjusted Quantity** to arrive at the **Adjusted Quantity**.

$$\text{Adjusted Quantity} = 121.60$$

Finally, the last step is to calculate the **Final Quote Price**:

$$\text{Final Quote Price} = \frac{\text{Total}}{\text{Adjusted Quantity}}$$

$$\text{Final Quote Price} = \frac{200}{121.60} = 1.6647$$

As noted, the **Final Quote Price** that has been calculated on this example is equal to the value that has been presented to the user (see the beginning of this example).