

## DOCUMENTATION

- ◆ Introduction
- ◆ Whitepaper
- ◆ **Technical Specifications**
  - Oracle Binary Encoding (OBI)
  - Oracle WebAssembly (Owasm)
  - Remote Data Source Executor
  - BandChain CLI & REST Endpoint

## BAND STANDARD DATASET

- ◇ Introduction
- ◇ Supported Blockchains
- ◇ Supported Price Data
- ◆ **Using the Data**
  - Using the Dataset
  - Using the Dataset in the Frontend
  - Using the Reference Prices in Smart Contracts**

Band Protocol Documentation / Introduction / Using the Dataset /  
Using the Reference Prices in Smart Contracts


## ON THIS PAGE

**Example Usage**

# Using the Reference Prices in Smart Contracts

To query prices from Band's oracle through smart contracts, the contract looking to use the price values should reference Band's `StdReference` contract. This contract exposes `getReferenceData` and `getReferenceDataBulk` functions.

`getReferenceData` takes two strings as the inputs, the base and quote symbol, respectively.

By using this website, you agree to our [Cookie Policy](#).  he

latest rates for those two tokens, and returns a `ReferenceData` struct, shown below.

```
struct ReferenceData {  
    uint256 rate; // base/quote exchange rate, multiplied by 1018  
    uint256 lastUpdatedBase; // UNIX epoch of the last update  
    uint256 lastUpdatedQuote; // UNIX epoch of the last update  
}
```

`getReferenceDataBulk` instead takes two lists, one of the `base` tokens, and one of the `quotes`. It then proceeds to similarly queries the price for each base/quote pair at each index, and returns an array of `ReferenceData` structs.

For example, if we call `getReferenceDataBulk` with `['BTC', 'BTC', 'ETH']` and `['USD', 'ETH', 'BNB']`, the returned `ReferenceData` array will contain information regarding the pairs:

- `BTC/USD`
- `BTC/ETH`
- `ETH/BNB`

# Example Usage

The contract code below demonstrates a simple usage of the new `StdReference` contract and the `getReferenceData` function.

```
pragma solidity 0.6.11;
pragma experimental ABIEncoderV2;

interface IStdReference {
    /// A structure returned whenever someone requests price data
    struct ReferenceData {
        uint256 rate; // base/quote exchange rate,
        uint256 lastUpdatedBase; // UNIX epoch of last update to base
        uint256 lastUpdatedQuote; // UNIX epoch of last update to quote
    }

    /// Returns the price data for the given base/quote pair
    function getReferenceData(string memory _base,
        string memory _quote,
        uint256 _timestamp)
        external
        view
        returns (ReferenceData memory);

    /// Similar to getReferenceData, but with multiple pairs
    function getReferenceDataBulk(string[] memory _base,
        string[] memory _quote,
        uint256 _timestamp)
        external
        view
        returns (ReferenceData[] memory);
}
```

```
contract DemoOracle {
    IStdReference ref;
```

By using this website, you agree to our [Cookie Policy](#). 

```
    constructor(IStdReference _ref) public {
```

```

        constructor(IStdReference _ref) public {
            ref = _ref;
        }

function getPrice() external view returns (uint256) {
    IStdReference.ReferenceData memory data = IStdReference(ref).getPrice();
    return data.rate;
}

function getMultiPrices() external view returns (uint256[]) {
    string[] memory baseSymbols = new string[](2);
    baseSymbols[0] = "BTC";
    baseSymbols[1] = "BTC";

    string[] memory quoteSymbols = new string[](2);
    quoteSymbols[0] = "USD";
    quoteSymbols[1] = "ETH";
    IStdReference.ReferenceData[] memory data = IStdReference(ref).getMultiPrices(baseSymbols, quoteSymbols);

    uint256[] memory prices = new uint256[](2);
    prices[0] = data[0].rate;
    prices[1] = data[1].rate;

    return prices;
}

function savePrice(string memory base, string memory quote) external {
    IStdReference.ReferenceData memory data = IStdReference(ref).getPrice(base, quote);
    price = data.rate;
}
}

```



### Found an Issue?

Help us improve this page by suggesting edits on GitHub.

By using this website, you agree to our [Cookie Policy](#).



PREVIOUS



## Using the Dataset in the Frontend



BAND PROTOCOL



[bandprotocol.com](https://bandprotocol.com)

This website is maintained by [Band Protocol](https://bandprotocol.com). The contents and opinions of this website are those of Band Protocol.

By using this website, you agree to our [Cookie Policy](#).

