

Introduction

When someone wants to request data from BandChain's oracle, they do not interact or call the data sources directly. Instead, they call an oracle script, which then proceeds to execute the necessary data sources.

An oracle script's job, unlike a data source, is to be responsible for compiling the results from various data sources to enable on-chain security. As such, an oracle script, similarly to a smart contract on other platforms such as Ethereum, Near and Solana, is executed on-chain rather than off-chain.

Oracle Script Specification

At a basic level, an oracle script itself is just an executable program that encodes two pieces of data:

- A set of raw data requests to the data sources it requires
- A method of aggregating those the raw data reports from the requests sent into a final result

Oracle scripts themselves are Turing-complete and can be written in any programming language that supports compilation into WebAssembly code. This composability and Turing-completeness make oracle scripts very similar to smart contracts.

Oracle Script Execution Flow

1. Preparation Phase
2. During the preparation phase, the oracle script outlines the data sources that are required for its execution. It then
3. sends out a request to BandChain's validators to retrieve the result from the required data sources. The content of this
4. request consists of the data sources' execution steps and the parameters required by said data sources.
5. Aggregation Phase
6. During the aggregation phase, the oracle script aggregates all the data reports returned by the validators. Each
7. report contains the values which the validator received from said data sources. The script then proceeds to combine
8. those values into a single final result. Note that the specifics of the aggregation process is entirely up to the design
9. of the oracle script. BandChain does not enforce any regulations regarding the aggregation method used and entirely
10. leaves that design decision to the creator of the script. Instead of a typical plain medianizer, the oracle script can
11. encode custom conditions such as a data deviation rule to ensure that all data points returned will stay within certain
12. percentage deviation from each other and to revert the transaction if otherwise.

For more information on oracle scripts and its execution, please refer to the corresponding page on our [wiki](#) . [Previous Deployment](#) [Next Creating an Oracle Script](#)