

f. Preface

[return to Front Matter](#)

The Distributed Immutable Data Objects (DIDO) [Reference Architecture \(RA\)](#) is meant to be used as a resource to guide the design, use, or selection of [Blockchain](#), [Distributed Ledger Technology \(DLT\)](#), or other Distributed Computing solutions such as [InterPlanetary File System \(IPFS\)](#), [Data Distribution Service \(DDS\)](#) and [Common Object Request Broker Architecture \(CORBA\)](#).

The purpose of DIDO RA 1.0 was to create a better understanding of the Blockchain and DLT, which were exploding in the [Information Technology \(IT\)](#) world after the publication of Satoshi Nakamoto's paper "Bitcoin: A Peer-to-Peer Electronic Cash System"¹⁾ and the subsequent success of the [Bitcoin](#). Since the publication of Nakamoto's paper, this excitement has grown way beyond the original Bitcoin. It has led to the promise/emergence of many other new cryptocurrencies, as well as the application of the well known and established concepts of distributed, peer-to-peer [applications](#) to supply chains, the [Industrial Internet of Things \(IIoT\)](#), natural resources, environmental sciences, and even the monetization of data.

In DIDO RA 2.0, the goal was to focus less on cryptocurrencies and more on generalizing peer-to-peer, distributed computing. As a parallel effort to the publication of DIDO RA 2.0, several products have been developed to work in parallel with and complement this paper:

- **DIDO Data Model (DIDO-DM)**: captures the conceptual data constructs described in the DIDO-RA including the [Community of Interest \(CoI\)](#) and testing.
- **DIDO Testing Environment (DIDO-TE)**: creates an environment that allows for virtualized testing of a [Distributed Application \(DApp\)](#) before it can be released into the "wild" using real hardware and networks.
- **DIDO Command Line Interface (CLI)**: defines a high level command language with which to send commands to each node covering the configuration, definition, and manipulation of data on distributed nodes.
- **DIDO Reference Implementation (DIDO-RI)**: provides a working interface to the DIDO-DM, DIDO-TE and DIDO-CLI.

The major enhancement of the DIDO RA 3.0 is the addition of a section on Requirements for DIDO Implementations from the "End User" perspective. Even though these requirements are for DIDO platform developers (e.g., Ethereum, Hyperledger, etc.) the intent is to enable a broad range of "End Users", e.g., financial, retail, supply chains, etc., to employ one or more DIDO platforms. This new section defines a Governance Requirements Model, which is comprised of three processes: Regulations (i.e., requirements specification), Execution (i.e., lifecycle from design to maintenance), and Compliance (i.e., oversight of a product or service, as well as, the processes of Regulation and Execution). This model ensures DIDO implementations are well-governed and moves DIDO Governance from its current "Execution Centric" state (i.e, DIDO platform developers) to the overall DIDO [Communities of Interest](#).

¹⁾

S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 24 May 2009. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>.

From:

<https://www.omgwiki.org/dido/> - **DIDO Wiki**

Permanent link:

https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:0.front:5_preface



Last update: **2021/12/07 00:48**