

2.2.2.3.4 Software

[return to Node Architecture](#)

By their very nature, DIDs are run on distributed, decentralized computers that require software to operate. The DID software itself can be classified as ancillary data and therefore can be distributed just like data to each DID [node](#). The DID software could include [Smart Contract](#), [Distributed Application \(dApp or DApp\)](#), scripts, containers, and DID [command line interface \(CLI\)](#) commands.

Smart contracts are programs actually stored on the [blockchain](#), and “triggered” to execute when a set of conditions are met. dApps are [applications](#) that don't reside on the blockchain but interact with the blockchain. dApps are used to communicate with smart contracts, and consequently with blockchain.

In other words, dApps can be considered as “blockchain-enabled” application, and smart contracts provide easy access the blockchain. This division of responsibilities is similar to the traditional separation used in [Web Application \(Web App\)](#) between the user-front-end (i.e., [Client](#)) and the backed (i.e., [Server](#)). Contract development is concerned with managing agreements or transactions.

There are many non-functional benefits of dApps such as:

- [Maintainability](#)
- [Portability](#)
- [Scalability](#)
- [Interoperability](#)

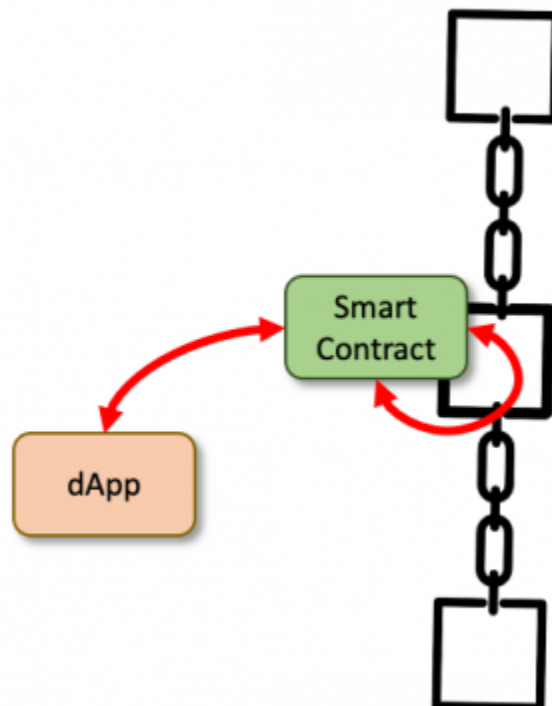


Figure 1: The Relationship Between the dApp, the Smart Contract and the blockchain

- **Note:** Not only is the data deployed on the blockchain, the smart contract itself is deployed on the blockchain. Therefore, before the Smart Contract can be used, it must be deployed to the blockchain. With slight modification, the dApp could interact with any system regardless of it is a [distributed system](#) or not as long as the [interface](#) remains the same.

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

Permanent link:
https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:1.2_views:2_tech_views:2-nodenet:3_nodearch:1_sw

Last update: 2022/02/03 16:41

