

2.3.2 Network Access Control Taxonomy

[return to Taxonomic Views](#)

Network access control taxonomy classifies the types of access individuals (i.e., nodes) have from outside and from within the node network. The two main classes of access control are [permissionless](#) and [permissioned](#).¹⁾

Within each of these two classifications it is possible to have [public](#) and [private](#) access. Public and private access define who is able to write data onto a network or ledger. In contrast, open (i.e., permissionless) and closed (i.e., permissioned) determine who is able to read the data. Networks are classified as²⁾:

- public and open
- public and closed
- private and open
- private and closed

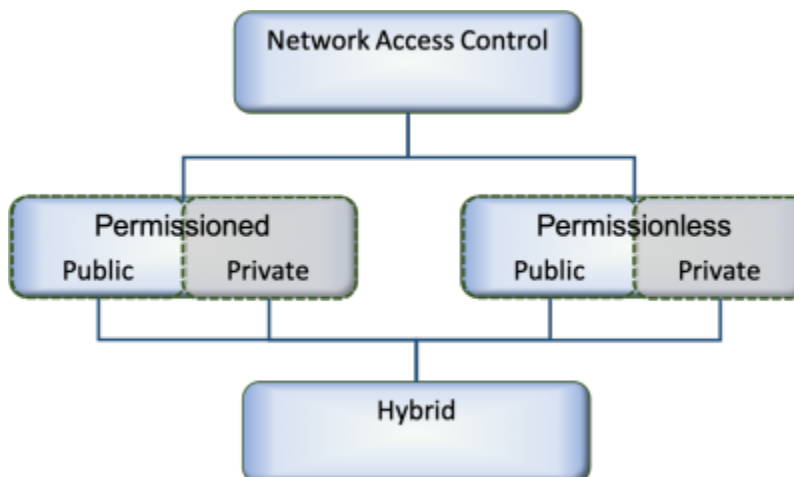


Figure 1: The Node Network Access Taxonomy

Another category of networks is a [hybrid network](#), which makes it possible to restrict the visibility of information on the network using a combination of [public](#), [private](#), [permissionless](#) and [permissioned](#) networks. Therefore, hybrid networks are appealing to regulated markets because they offer the benefits of public blockchain and private blockchain together.³⁾

Table 1: Taxonomy of DIDO Access Control

Permissionless Networks	Permissioned Networks
Public Networks	Private Networks
Hybrid Networks	

¹⁾ , ²⁾

“Public Vs Private Blockchain In A Nutshell”, Demiro Massessi, 12 December 2018,
<https://medium.com/coinmonks/public-vs-private-blockchain-in-a-nutshell-c9fe284fa39f>

³⁾

“Hybrid Blockchain: Decentralized Option for Highly Regulated Markets - Few players in highly regulated markets have adopted blockchain technology. However, hybrid blockchain will change this.”, Mina Down,

14 November 2018

<https://blog.goodaudience.com/hybrid-blockchain-decentralize-highly-regulated-markets-900f30a37903>

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

Permanent link:
https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:1.2_views:3_taxonomic:2_network_access_ctrl:start&rev=1607393169

Last update: **2020/12/07 21:06**

