

Distributed Services Architecture (DSA)

DSA brings together entities like devices, services, and applications into one data model that updates in real time. By representing these entities as one system, DSA makes possible or simplifies various tasks, such as analytics, inter-device communication, distributed computing, and application development.

You can read more about DSA on the [DSA home page](#).

Structure of DSA

DSA represents each of the entities in the system as one of the following types:

- **Broker**—A broker does a wide variety of management tasks. For example, a broker manages security, links, subscriptions, and node permissions. Other entities in the system can operate only as permitted by a broker. A broker also saves configuration data to disk. A broker also routes data—that is, it moves data from a source to a destination.
 - **DSLink**—A DSLink connects to a broker. A DSLink creates, publishes, and interacts with data. A DSLink can also subscribe to data in the DSA system—that is, it can receive data whenever the data changes. You can find DSLinks in [IOT-DSA on Github](#).
 - **Node**—Some nodes are brokers, DSLinks, metrics, actions, or attributes. Other nodes, such as folders, do not fit into any of these categories.
 - **Metric**—A metric can exist on any broker, DSLink, or other organizational node. A metric is a key/value pair, in which the value can be any of the data types listed in Supported Data Types, including an arbitrary value map.
 - **Data Node**—Data nodes allow data to be stored on the broker's host server.
 - **Action**—An action can exist on any broker, DSLink, other node, or metric. An action is an invocable command that can affect an entity. For example, an action might create a node or set a metric value.
 - **Attribute**—An attribute can exist on any broker, DSLink, other node, or metric. An attribute is metadata for the selected entity, represented as a key/value pair.
-

Upstream and Downstream Connections

Upstream and *downstream* connections are important concepts for permissions configuration when you work with multiple servers or with multiple brokers on one server. A downstream entity requests permission, and an upstream entity either grants or refuses that permission. A broker is always upstream from its DSLinks. A broker can be either upstream or downstream from another broker.

Supported Data Types

DSA supports these data types:

- **String**—A sequence of characters or an empty string.
- **Number**—A number or a null value.
- **Bool**—A true or false value.
- **Array**—An array object or a null value. The values in the array are of the dynamic data type. An example array of number

values is `[2,3,5,7,11]`. An example array of map values is `[{"hello":"world","number":1}, {"hello":"world"}]`.

- **Map**—A map object containing key/value pairs, or a null value. The key is always a string, and the value is the dynamic data type. An example value is `{"hello":"world","primes":[2,3,5,7,11]}`.
- **Binary**—A byte array expressed as a string, or a null value. The string begins with `\u001Bytes:` and ends with a byte array encoded in base 64.
- **Dynamic**—A value that can be any of the above types.

Installing DSA

To install DSA on Mac OS X, Microsoft Windows, Linux, or other platforms, follow the steps in the relevant video on [this page](#), depending on your platform.

If you are installing DSA on a Raspberry Pi, BeagleBone Black, or DGBBox, you can also follow the steps in [this blog post and video](#).

If you install DSA with DGLux5, and do not have a DGLux5 license, follow the steps to request a license when you are prompted to do so.

Security and Permissions

Security and permissions information will go here.

Installing and Updating a DSLink

Installing and Updating DSLinks will go here.

Orientation to the DGLux5 Data Panel with DSA

DGLux5 Data panel information will go here.

Orientation to the DGLux5 Metrics Panel with DSA

DGLux5 Metrics panel information will go here.

Remote Dataflow

DGLux5 remote dataflow information will go here.

From:

<https://wiki.dglogik.com/> - **DGLogik**

Permanent link:

https://wiki.dglogik.com/dsa_wiki:home?rev=1475088029

Last update: **2021/09/20 14:19**

