Now that you have an understanding of the objectives of Solid and the legal framework supporting it. Let's dive deeper into how Solid actually works. Solid focuses on sharing information in a way that preserves privacy. It achieves this by storing personal data in "Personal Online Data Stores" or "pods".

A pod can contain a variety of information. It could contain your personal profile data, your financial data, your health data, your travel plans or any other kind of information you want. Pods are Web accessible storage services, which can either be deployed on personal servers by the users themselves, or on servers by pod providers similar to the current cloud storage providers.

Solid itself can be best defined as a set of standards, including those related to linked data and advanced access control.

The technology behind Solid is based on the Resource Description Framework, RDF, and Semantic Web technologies. Solid uses "vocabularies" which describe the data in a standardized way so that applications know how to access specific types of data relevant for the application.

Basically, a Solid pod is a server with data and a user interface. The data pod is compliant with the HyperText Transfer Protocol, HTTP, with the Linked Data Platform, and with the Linked Data Notifications.

The Solid applications are client-side Web or mobile applications that read and write data directly from the pods. Access to specific pieces of data on the pod is granted by the pod owners. The pod owners retain complete ownership and control over the data in their pods: what data each pod contains, where each pod is stored, and which applications have permission to use the data.

This data or resource extraction procedure needs to be secure of course. Solid's authentication architecture is concentrated on three actors: the client application, the resource server and the identity provider.

The flow works as follows: the client application requests a resource from the resource server. The resource server reacts by asking for the required access token.

The client application will present its client identifier with its associated secret to the identity provider. And requests an authorization code.

If granted, the Client presents the authorization code to the Token Endpoint. The client application will get an access token and OpenID Connect ID token, which it then will present to the resource server. After which the resource server will validate the signature on the access token with the public key from the identity provider.

And only when this is valid, the resource server will return the requested resource.

But let's go back one level and have a broader view on the Solid ecosystem. There are 3 main actors involved: Citizens, Government and the Third-party applications. The actors are interconnected through the personal data pod and they can have the role of 'storing data' and 'reading data'.

In a more complex ecosystem, the Citizens, Government and Third-party applications each have their own personal data pods, and access for storing and reading data is granted between each other.