

WHITE PAPER

Space out in cloud



With the introduction of spaces and the content lifecycle we have created a number of concepts and a way of working with content that we think a lot of our customers will be happy with. But not all customers are alike, we will always have customers that want to do more.

Luckily the spaces concept with some of its more advanced space functionality allow for this flexibility to adapt to customers' needs to work with and protect their data.

If we start with the movements of apps and its relations to data, we have introduced the concept of space aware data files and data connections. This means that data files and data connections can be stored in spaces and referenced in different ways to its location. The permissions are always applying to the current user, but this might be used to build out some interesting structures.

So first let's go through how this works in the data load editor.

When referring to a data file or data connection in the script it looks something like this

LOAD

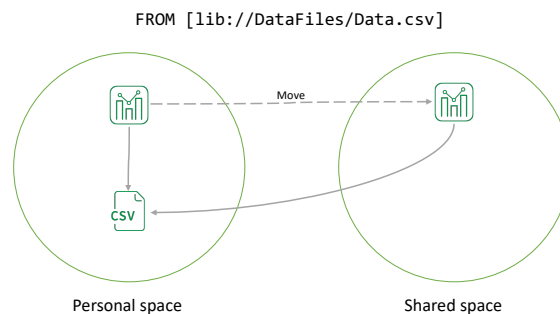
*

```
FROM [lib://DataFiles/Data.csv];
```

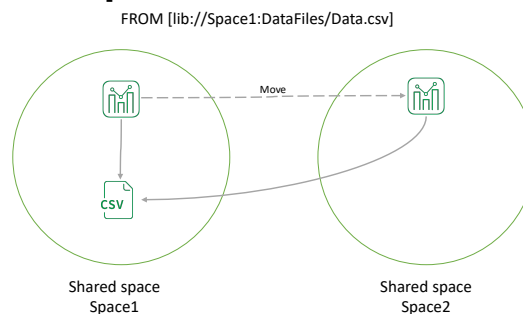
The example above will load a load the Data.csv file from the user executing the reloads personal space.

But if this is not what I want to happen, I have these three options to change the behavior

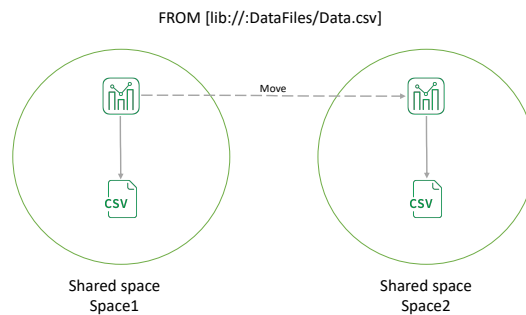
`FROM [lib://DataFiles/Data.csv]` would mean load from your private space



`FROM [lib://[Space name]:DataFiles/Data.csv]` would mean that you always load from the data files in the space called [Space name]



`FROM [lib:///DataFiles/Data.csv]` means that you load from the same space as the app resides in.



Loading from a static space would come in handy if you have team of analytic developers, but you want to separate data access between the them. You can now create shared spaces that only contain data files and data connections. The developer will only be able to use the data in the spaces they have permissions to and you get central governance of data in one place using FROM [lib://[Space name]:DataFiles/Data.csv] syntax to reference the data from your script. So even if the app is moved around it can still only be reloaded with the same permissions to the data.

But let's say that we instead want to have test data and production data. With the use of FROM [lib://:DataFiles/Data.xlsx] we can now let the app load different data based on which space the app resides in. An example would be to let developers have test data in their shared spaces and when you move or publish the app, it will now load the data from the new space's data files and connections (given they have the same name) all without having the need to edit the script.

You can now combine these concepts in the way your company need to secure and work with the data.

An example of putting this together with separation of concerns through roles in spaces would be to build out a quality process. Let a developers collaborate developing with test data in their shared spaces, allow them to publish into a managed space where the app can be reloaded and tested with production data before being published into the managed space for production where users can start consuming or self-serv, all without having to edit or change the script.

As an extra bonus starting in the April 2020 release all of the data files and data connections in a space can be managed from the hub.

If you have questions or ideas, please do not hesitate to put a comment below.