

OMAG Server Package

ATLAS-2422 - OMAG Server

RESOLVED

The **Open Metadata and Governance (OMAG) Server Package** provides a server runtime for the Open Metadata and Governance components. This includes:

- [Open Metadata Types](#)
- [Open Metadata Access Services \(OMAS\)](#)
- [Open Metadata Repository Services \(OMRS\)](#)
- [Open Connector Framework \(OCF\)](#) Connector Installation Manager
- [Open Metadata Graph Repository](#)

Through the OMAG server configuration, the OMAG Server is able to perform combinations of the following roles in the metadata highway.

- Access Layer - supports the [Caller Integration Pattern](#)
- Repository Proxy - supports the [Adapter Integration Pattern](#)
- Metadata Repository - supports all integration patterns where a local store of metadata is required.

The OMAG Server has a REST API and UI for administering the OMAG Server. This includes setting up the OMAG server roles it is supporting and managing its connectivity on to the metadata highway. It is also able to dynamically load OCF connectors and their configuration (connections). Through this service, the OMAG Server supports the [Plug-in Integration Pattern](#) for OCF Connectors.

The section below describes the operation of the OMAG Server in each of its roles and how it supports metadata tools and repositories connecting onto the metadata highway.

OMAG Server Roles

When the OMAG Server is configured to provide an Access Layer it means it is hosting the [Open Metadata Access Services \(OMASs\)](#) APIs and Topics for external tools and applications. In the example shown in figure 1, there is a tool called ABC that is using the [Caller Integration Pattern](#) to access open metadata and related services.

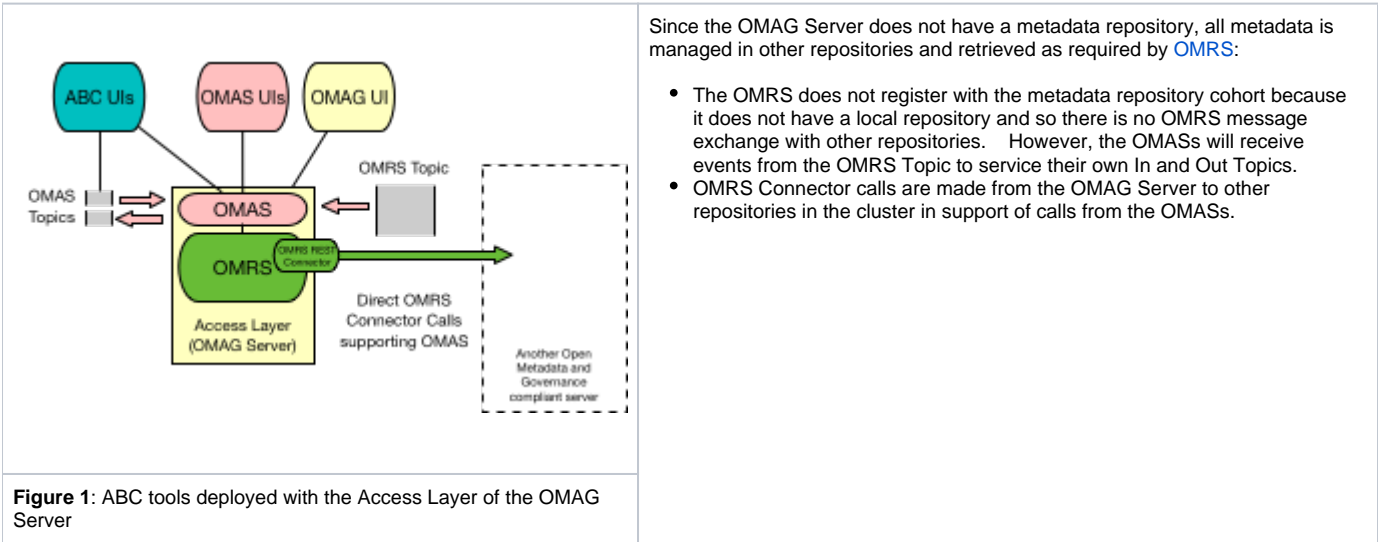


Figure 2 shows the OMAG Server with both the Access Layer and a local metadata repository enabled. Now metadata from ABC is stored in its local OMAG Server's repository. This is blended with metadata from other repositories by OMRS. With a metadata repository in play, ABC can operate independently and connect into the open metadata repository cohort and is therefore now an open metadata [Native](#).

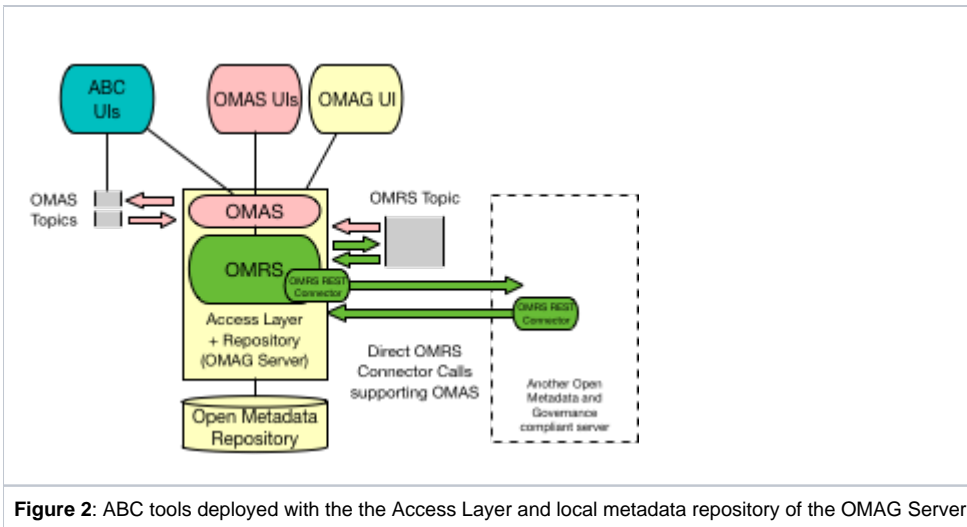
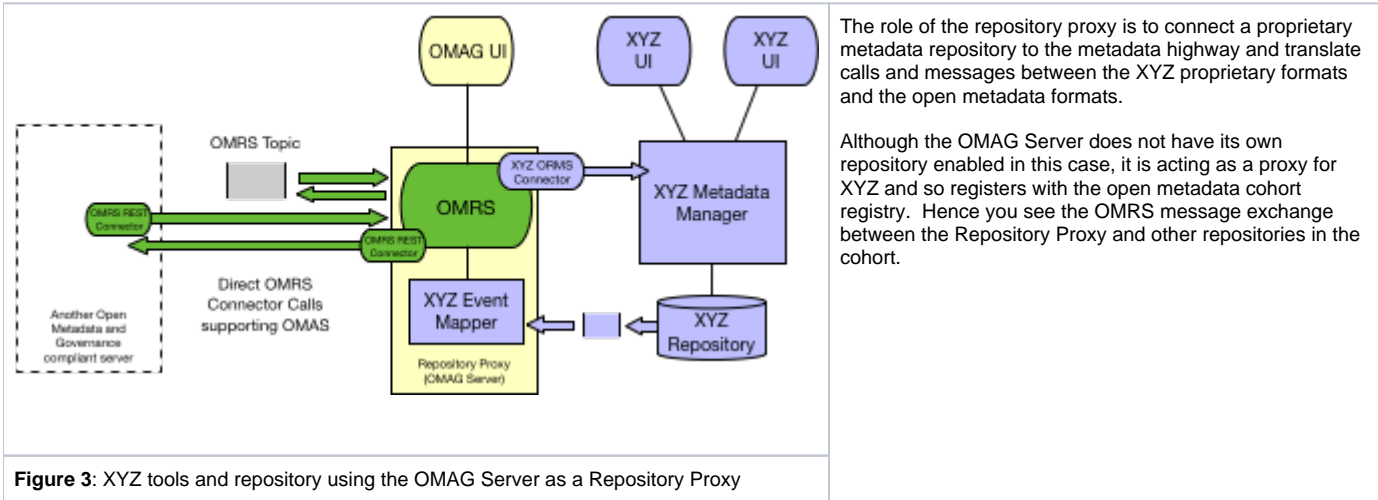


Figure 3 shows a tool and repository product called XYZ connected to the Open Metadata and Governance Ecosystem through an OMAG Server configured as a Repository Proxy.



In figure 4, the repository proxy has a local metadata repository. The repository is being used to store metadata that augments the metadata in the XYZ repository. This is used when the XYZ metadata model does not support all of the metadata types it needs, or there is a mismatch between the XYZ metadata metamodel and the open metadata types, requiring metadata to be cached locally in order to build up metadata content that can be stored in the XYZ repository.

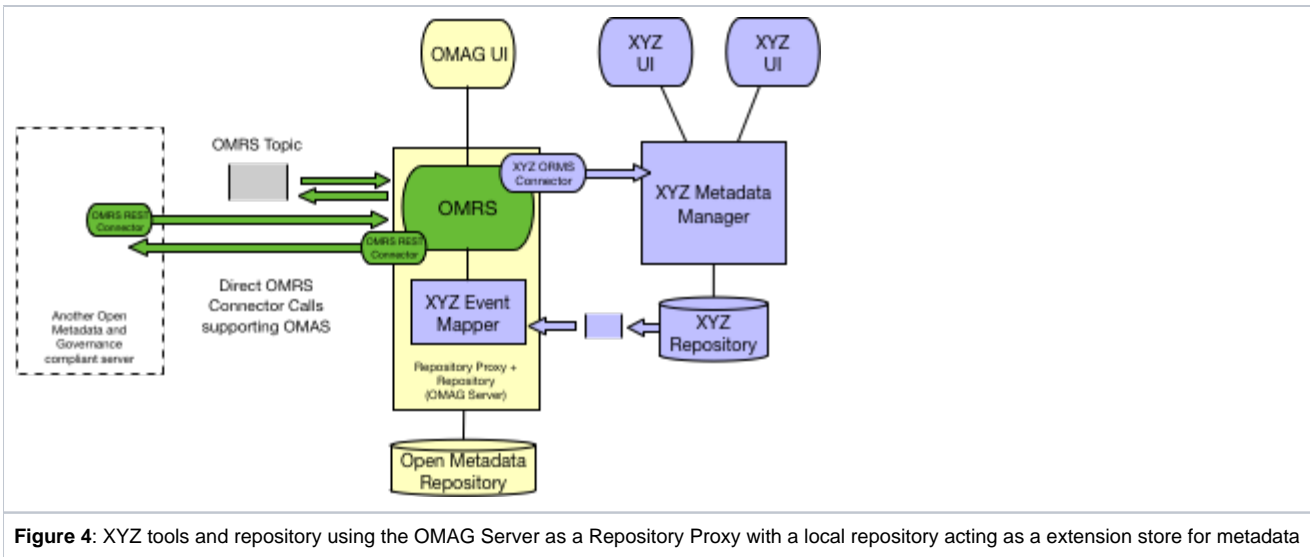
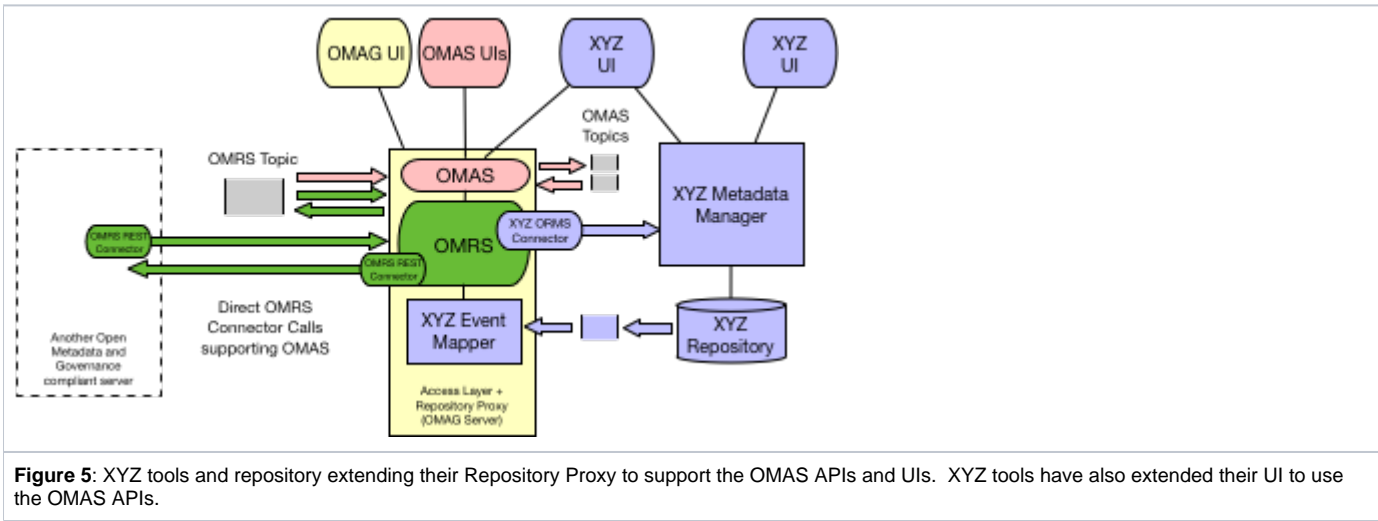
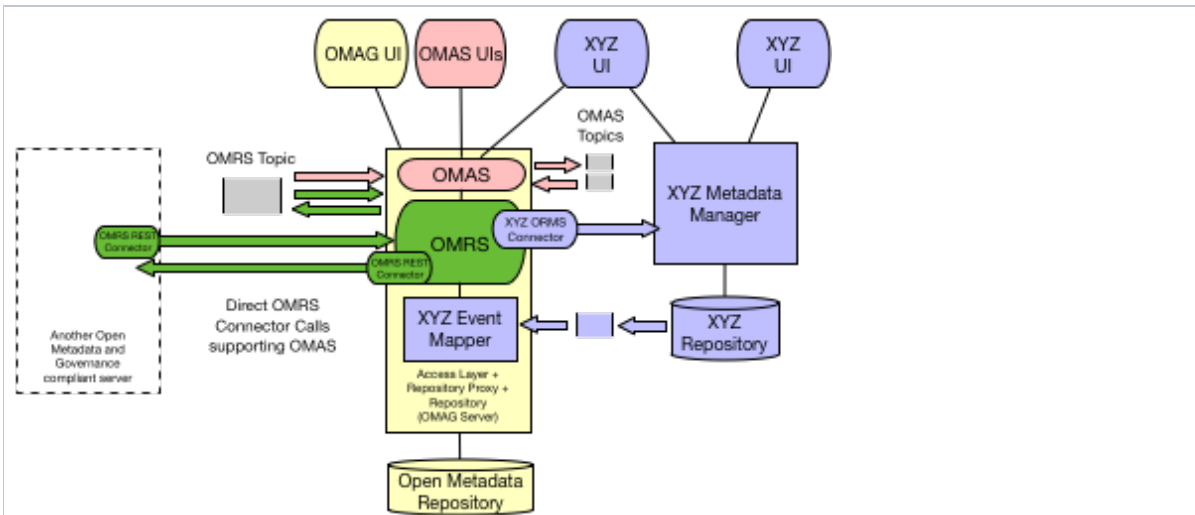


Figure 5 shows a further development where the users of XYZ are given access to the OMAS functions, both through the OMAS UIs and the XYZ UIs. Thus, this is an example of the Access Layer and the Repository Proxy in action together.



Finally figure 6 adds the metadata repository so the repository proxy has a local store as well as the XYZ store. Thus all 3 roles are enabled.



The OMAG Server, with its frameworks, and the fact that it is available as open source, can be extended with more advanced function through well defined plug-points. The [Apache Atlas Server](#) is an example of the [Native Integration Pattern](#) built on top of the [OMAG server](#) to give it native support for the open metadata and governance APIs. The Apache Atlas server then provides value-add services or managing metadata and governing data, such as the Hooks and Bridges for Hadoop technology.