

## Functionality of an application

**Question:** How can you model externally visible functionality of applications with ArchiMate?

**Solution:** The ArchiMate concept that was designed explicitly for modeling externally visible functionality is the application service. This is connected with a realization relationship to an application function belonging to an application. Corresponding interface(s) can be visualized if required. The figure below shows this modeling pattern:

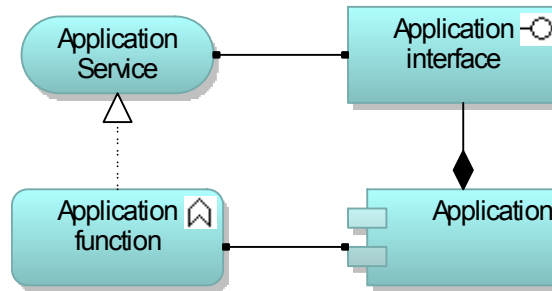


Figure: Externally visible functionality

This way of modeling can be used both for internal and external application services.

**Consequences:** If an application produces a lot of functionality, a model like this can become cluttered by the great number of realization relationships.

The suggestion can be easily induced that a service oriented architecture is in place while in reality this is not so.

**Alternatives:** This model pattern can be simplified by leaving out the application function and deduce the realization relationship to the application component (figure below to the left). If the interface has no relevance for the model, it can be left out (figure below to the right)

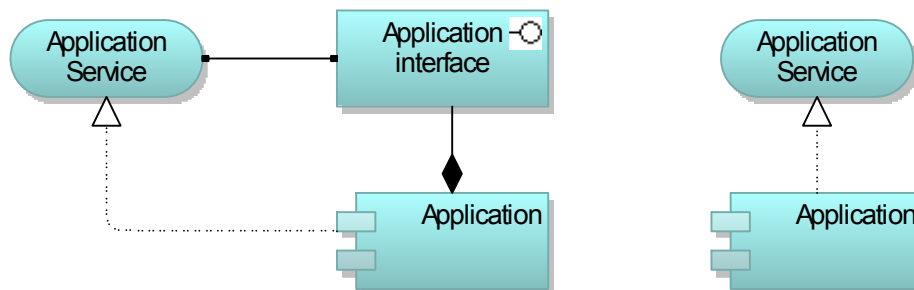


Figure: Externally visibly functionality - simplified

When no services can be distinguished, the externally visible functionality of an application can be visualized by delivering the application function via an interface to the 'consumer'. In the example below, a Business process is supported by an Application function, which is exposed by an Application interface.

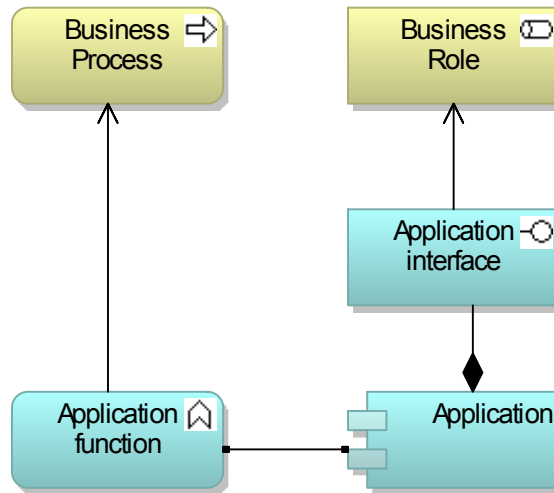


Figure : Externally visible functionality coupled to a process role

In the last view the application interface can also be left out. The result is a 'used by' relationship between the application and the business role.

**Relationships with other good practices:** For functionality that is not interactive there is a relationship with good practice 'Interfaces between applications'. Visualizing the functionality of an application resembles visualizing applications within an application portfolio. The good practice 'Application landscape' is also applicable for the functionality of an application.