

hsadminNg Glossary

This is a collection of terms used in this project, which either might not be generally known or unclear in meaning. If you miss something, please add it with a **TODO** marker.

Blackbox-Test A blackbox-test does not know and not consider such internals of an implementation, it just tests externally observable behaviour.

Business Object Used in the RBAC-system to refer to an object from the business realm. The usual term is *domain object* but in our context, the term *domain* could be too easily confused with a DNS *Internet domain*.

Dummy A *dummy* is a kind of *Test-Double* which replaces a real dependency which is not really needed in the test case.

Fake A *fake* is a kind of *Test-Double* without using any library, but rather a manual fake implementation of a dependency.

Mock A *mock* is a kind of *Test-Double* which can be configured to behaviours as needed by a test-case.

Often the term “mock” is used in a generic way, because typical mocking libraries like *Mockito* can also be used as dummies or spies and can replace fakes.

RBAC Abbreviation for *Role Based Access Control*. A system to control access to business objects by defining users, roles, and permissions. See also The INCITS 359-2012 Standard.

In our case we are implementing a hierarchical RBAC for a hierarchical and dynamic business object structure. More information can be found in our RBAC Architecture Document.

Tenant *Tenant* is one of the standard roles of Hostsharing’s RBAC system. It is assigned as a sub-role to those who have rights on sub-objects of a business object. Usually, tenants can only view the contents.

Generally, tenant roles only apply for the mere existence, id and name of a business object, not for internal details. E.g. a tenant of a customer could be the administrator of a hosting package of that customer. They can view some identifying information of that customer, but not view their billing and banking information.

Whitebox-Test A whitebox-test knows and considers the internals of an implementation, e.g. it knows which dependencies it needs and can test special, implementation-dependent cases.

Test-Double A “double” is a general term for something which replaces a real implementation of a dependency of the unit under test. This can be a “dummy”, a “fake”, a “mock”, a “spy” or a “stub”.

Test-Fixture Generally a test-fixture refers to all code within a test which is needed to setup the test environment and extract results, but which is not part of the test-cases.

In other words: The code which is needed to bind test-cases to the actual unit under test, is called test-fixture.