CAP

The CyberSpace Architecture Project

GLOSSARY

This glossary is currently very incomplete!

Aspect

> Aspects are kinds of viewing something, they characterize the points of view one can take on. An object (in the traditional meaning of the word) together with an aspect results in a *view*.

Aspect Dependency

> The property of a model, class, object or entity to depend not only on the captured thing itself but on some *aspect*. In contrast to *subject dependency*, aspect dependency is more abstract and coupled stronger to the object than to the viewer.

Accessory

> The *environment* of an object is composed of accessories. This includes all kinds of objects which are related very close to the *kernel*. They are not statically bound to the *kernel*, but form a dynamic and therefore adaptive and felixble completion of it.

Kernel

> The kernel mostly resembles what a traditional class instance is, but is more pure in that it is restricted to the object essentials: *persistance relevant data* to ensure the objects identity as well as a *primitve but complete set of methods* to allow manipulation. Methods and Data are as far as possible *subject independent*. The kernel represents the *identity* of the object.

Shell

> A shell is a hull around the *Kernel* and the *Accessories* in the *SKEM* architecture. It manages the inner components and provides sophisticated access to them under various aspects.
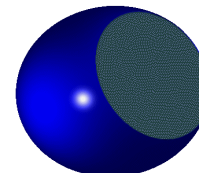
Subject Dependency

> The property of a model, class, object or entity to depend not only on the captured thing itself but on some *subject*. This emphasizes the role of the viewer in the modeling process. In contrast to *aspect dependency*, subject dependency is more concrete and coupled stronger to the viewer than to the object.