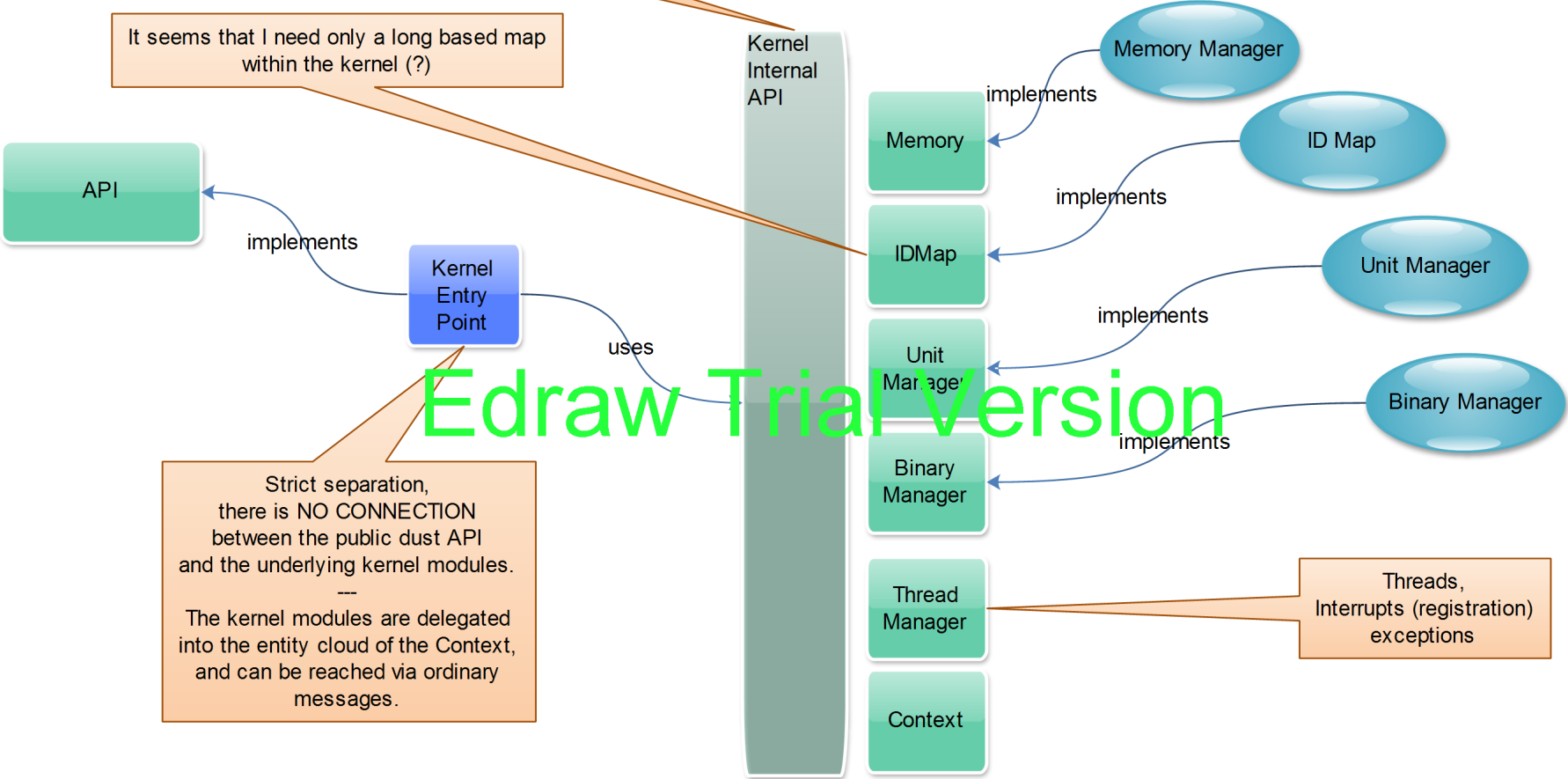


Edraw Trial Version

The access management is not yet displayed. I think for application with such features, the access protection must be built into the kernel (user, identification, access check), but the exact location is not clear yet. It should exist in core function declaration, and... well, it should be active: the initialization phase and routines should have a general protection shield, no custom solution. Perhaps it is that easy that the boot procedure runs as local admin (can set kernel entities, drivers), but not root (it can't change the configuration)

It seems that I need only a long based map within the kernel (?)



Legend

C Header

C Source

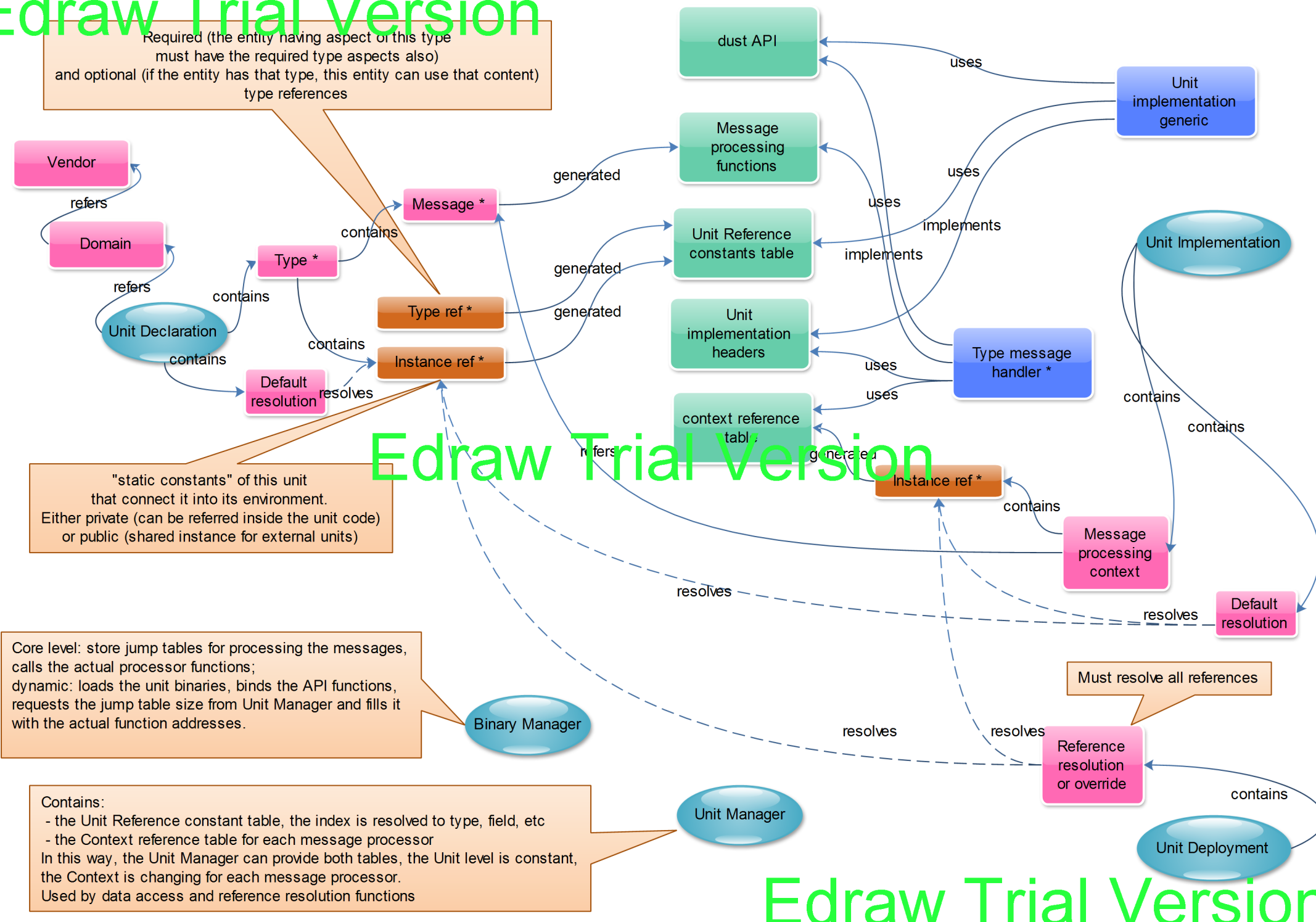
Mandatory Type
and handler code

Optional Type
and handler code

Kernel object instance
(memory block)
[of type]

Component

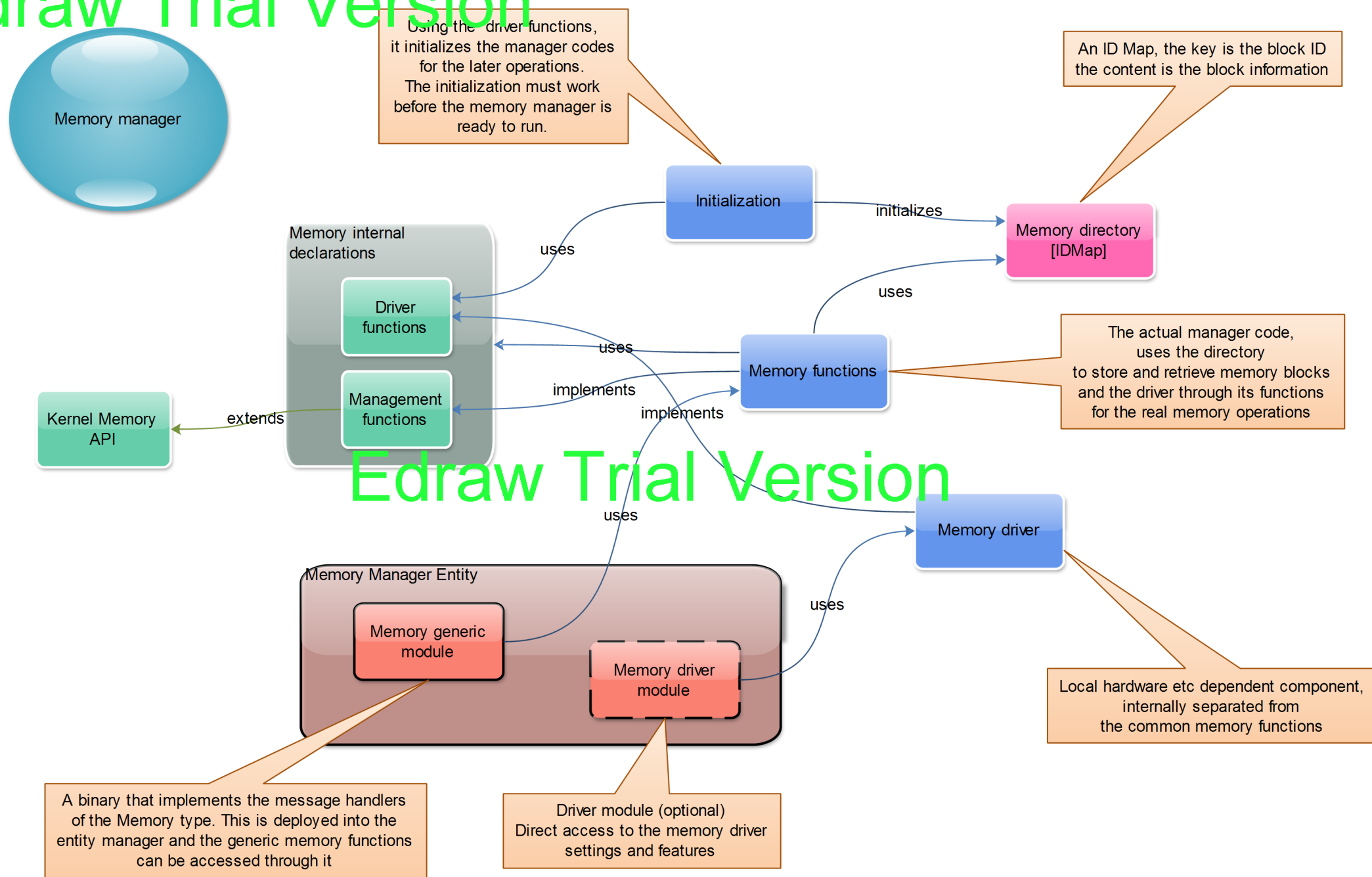
Edraw Trial Version



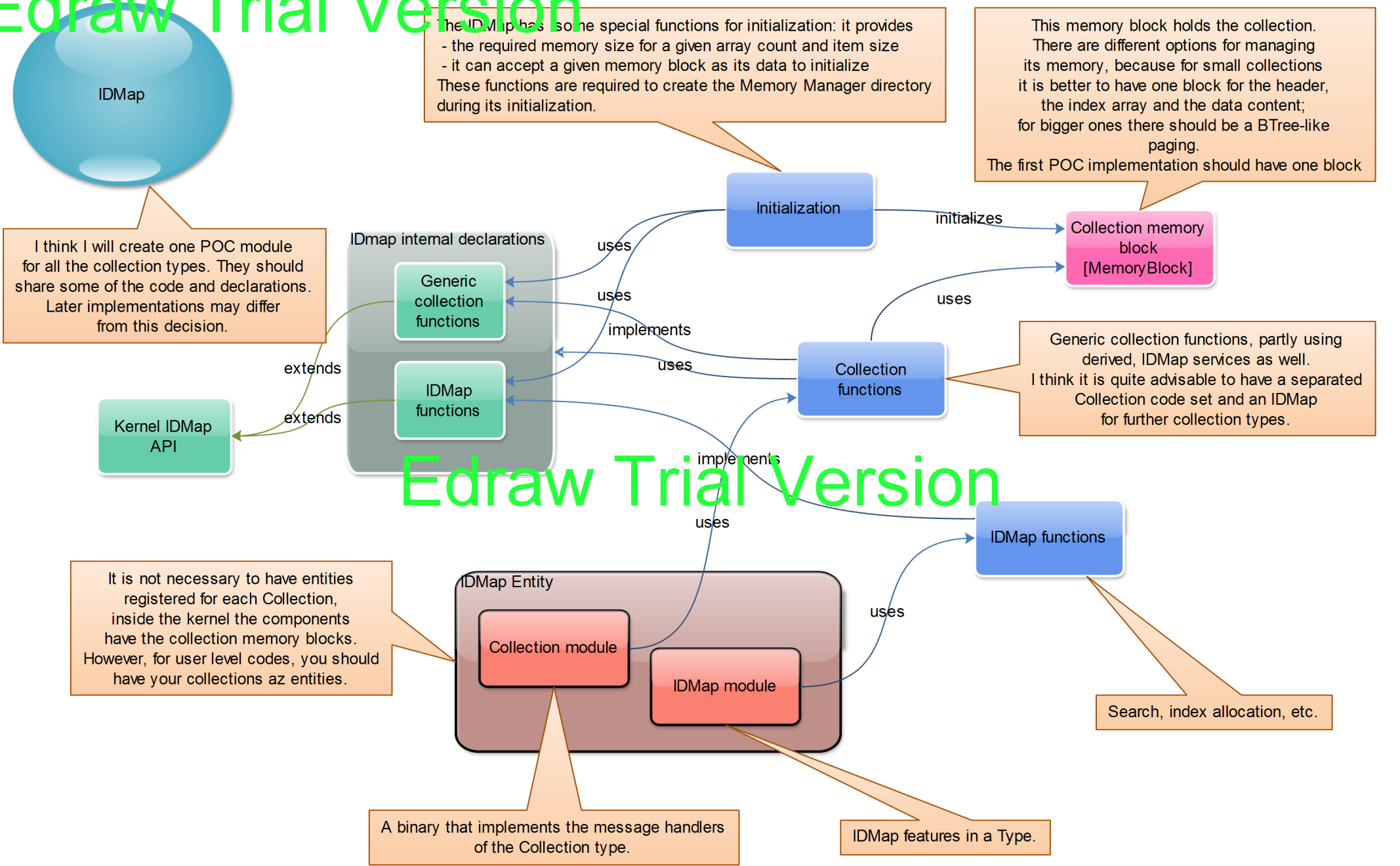
Edraw Trial Version

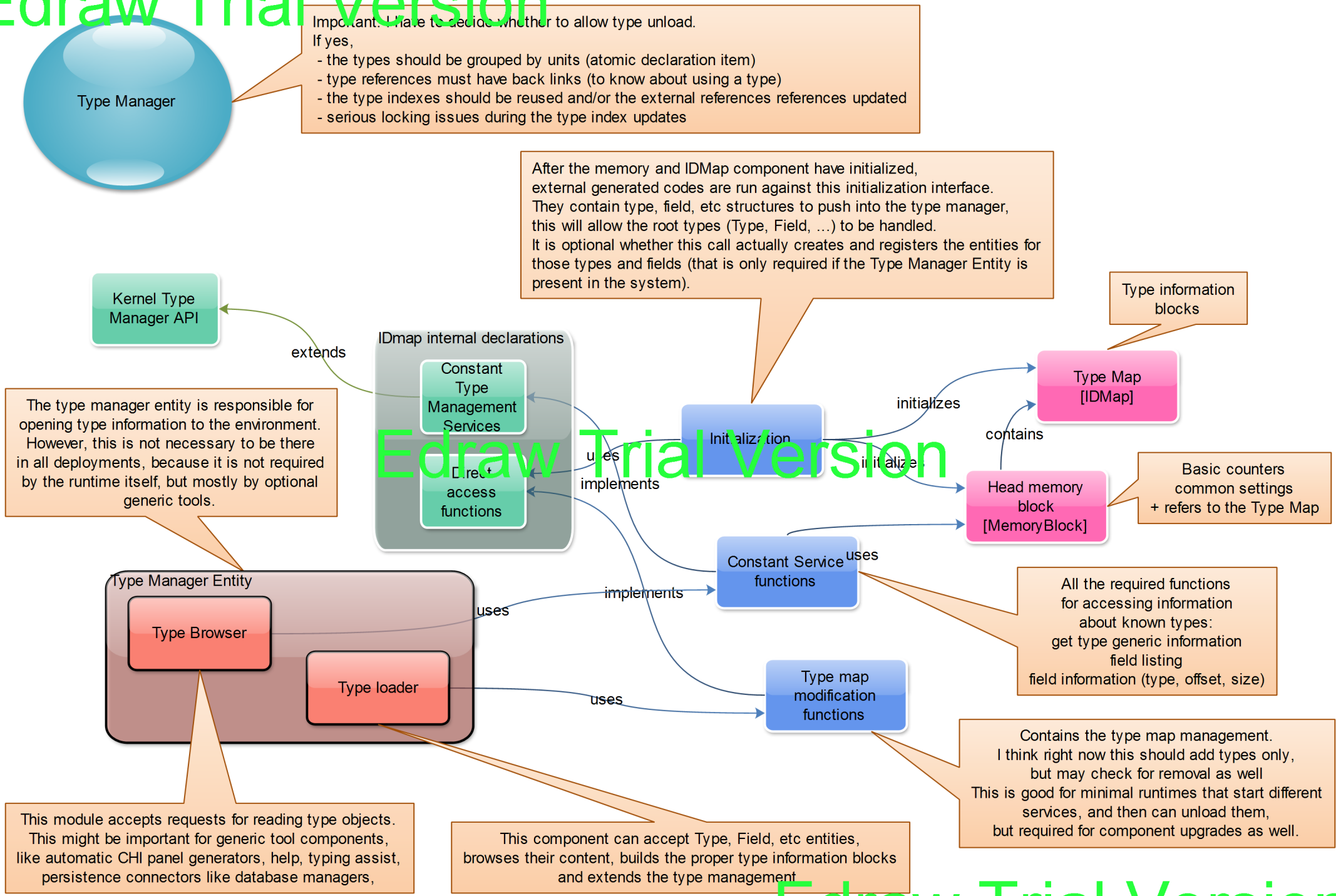
Edraw Trial Version

Edraw Trial Version



Edraw Trial Version





Edraw Trial Version



Edraw Trial Version

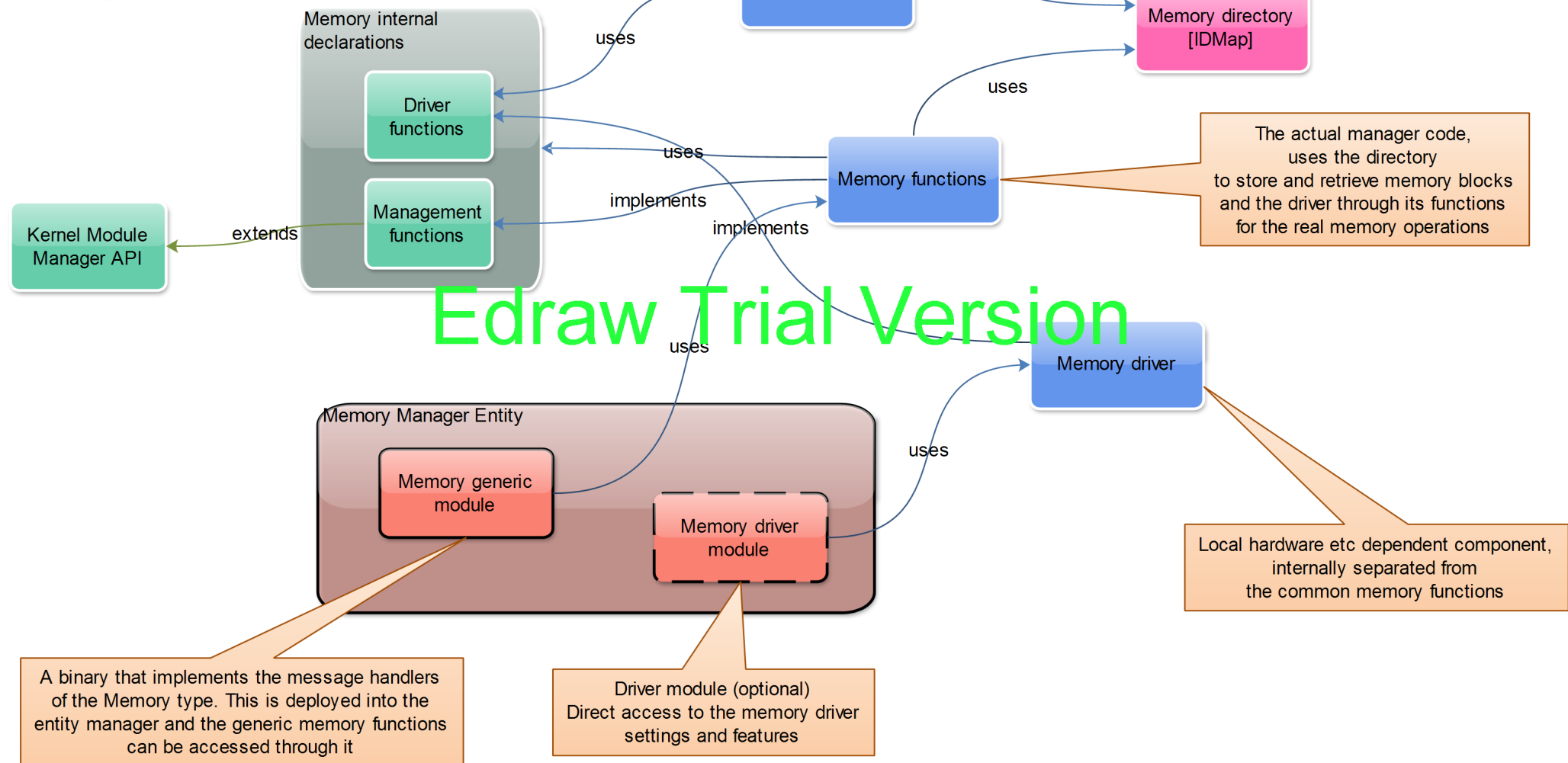
Edraw Trial Version

Edraw Trial Version

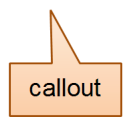
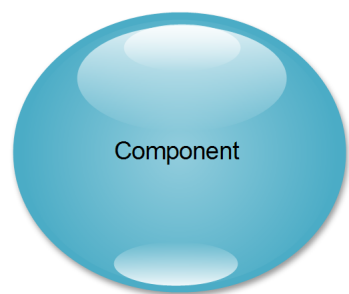
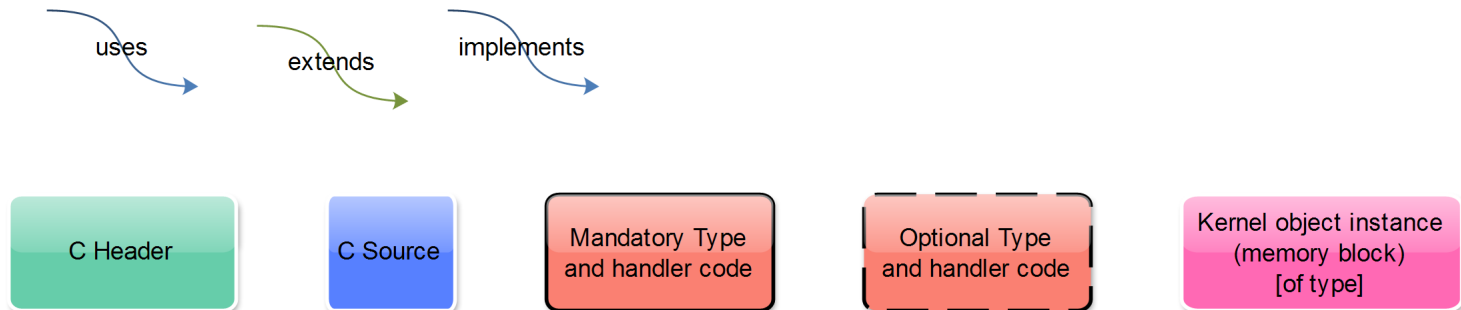


Using the driver functions, it initializes the manager codes for the later operations. The initialization must work before the memory manager is ready to run.

An ID Map, the key is the block ID the content is the block information



Edraw Trial Version



Edraw Trial Version

Edraw Trial Version