

MEKON and HOBO are software frameworks for building ontology-driven applications, the latter being built on top of the former. The core frameworks are independent of any specific representational format, with a plug-in framework enabling the incorporation and integration of a variety of ontology-like formats and associated reasoning mechanisms.

MEKON provides an in-memory Frames Model (FM) representation for models derived from External Knowledge Sources (EKS). Automatic updating of both the form and content of evolving instantiations of the FM is provided by reasoning mechanisms associated with the EKS. Plug-ins exist for deriving FMs from OWL ontologies, and for Description Logic (DL)-based instantiation-updating.

FM instantiations can represent either concrete "instances" or abstract "queries". A plug-in framework enables the storage of the instances in an "instance store", and the execution of the queries over such a store. Instance-store plug-ins exist based on (a) in-memory OWL constructs with DL reasoning, (b) RDF triple stores (Stardog and Jena) with SPARQL, and (c) the XML database, BaseX.

HOBO enables the "binding" of a Java Object Model (OM) to a MEKON FM. This provides (a) appropriate APIs for applications that need to deal directly with domain-specific concepts, and (b) a structured means of combining domain-specific procedural processing with generic EKS-based reasoning.

Current applications of MEKON and HOBO include a knowledge-driven clinical documentation system and a web-based recruitment application. Noteworthy aspects of the frameworks' usage within these systems include (a) a fairly complex integration of OWL with a custom modelling system, (b) domain-specific customisation of the standard instantiation-updating mechanisms for scoring-based inference-filtering, and (c) domain-specific customisation of the standard querying mechanisms for, amongst other things, numerical aggregation.

The demo is via the MEKON Model Explorer GUI, which enables a model developer to browse a FM, and explore the dynamic behaviour of specific instantiations, both concrete and abstract. The demo model is based on a small OWL ontology and compatible HOBO OM, with the demonstrated behaviour being driven by both DL reasoning over the ontology, and procedural code associated with the OM.