Yggdrasil documentation

Temporal storage of simple objects

Overview of Yggdrasil

Terje Kvernes & David Ranvig

July 11, 2008

Contents

1 What is Yggdrasil?

Yggdrasil aims to a "dynamic relational temporal object database". In essence, Yggdrasil aims to add two abstractions to the traditional view of a relational database: implicit temporal storage and a simple object model to represent the data stored. In addition to this Yggdrasil allows the relations of the entities stored within to be altered, and new entities and their relations to be added while the system is running. The relations are described by the administrator of the system and as soon as any relation is described to Yggdrasil, it is added to the overall structure of the installation.

In Yggdrasil lingo you can think of an entity as a "class" from the OO world. The "object" is an instance of this entity, each object existing in several temporal versions within the system.

1.1 Objects

An object is an instance of an entity within the system. This instance is the primary data unit within Yggdrasil.

An object within Yggdrasil isn't a singular instance of grouped data. As any change to this object is kept, every version of the object throughout its existence is stored.

The limitations are currently bound to the objects being fairly simple, they are not allowed to store anything more complex than anything that can be mapped into a text field, and their only relations to other objects are the ones defined by the setup of Yggdrasil.

1.2 Relations

All relations within Yggdrasil exist between entities.

1.3 Temporal dimension

A principal idea of Yggdrasil is that data is only inserted, never deleted. This also means that there is a clear distinction between "available" and "current" data contained within the system. As long as no time frame, or slice, is requested, all requests work on the "current" dataset.

Deletion only happens if and only if the object structure ("entity" in Yggdrasil lingo) they belong to is deleted, and that deleted structure is "purged". Deletion of an entity is therefore, in some circumstances, reversible.

Changes in the structure will retain the information if possible, the system will inform you at any time if any action you take will permanently delete any data.

1.4 Dynamic storage

New entities and new relations between entities can be issued on the fly while the system is running live.