Overview documentation, trivariatemodel

Vibeke Skytt

June 17, 2010

1 Trivariatemodel

The module trivaratemodel represents a set of volumes. As a module it depends on several other GoTools modules:

- gotools-core for geometric representations of curves and surfaces
- trivariate for geometric representation of volume
- compositemodel for topology for surface sets and solids

There are also further dependencies to other modules particularily through the module compositemodel.

2 Volume topology

The volume topology in GoTools is an extension to the boundary represented topology implemented in the module compositemodel. However, in contrast to the boundary represented topology, the volume topology is not manifold. Thus, we must expect more than two faces to meet in an edge.

The top entity in the topology structure is the volume model (VolumeModel) which consists of a set of volumes. Each volume has a topological entity implemented in ftVolume and a geometrical representation implemented as ParamVolume. Information about the geometric representation of a volume can be found in the module trivariate.

As for Body in compositemodel, ftVolume is surronded by one or more shells represented as SurfaceModels. A shell is a collection of faces (ftSurface)

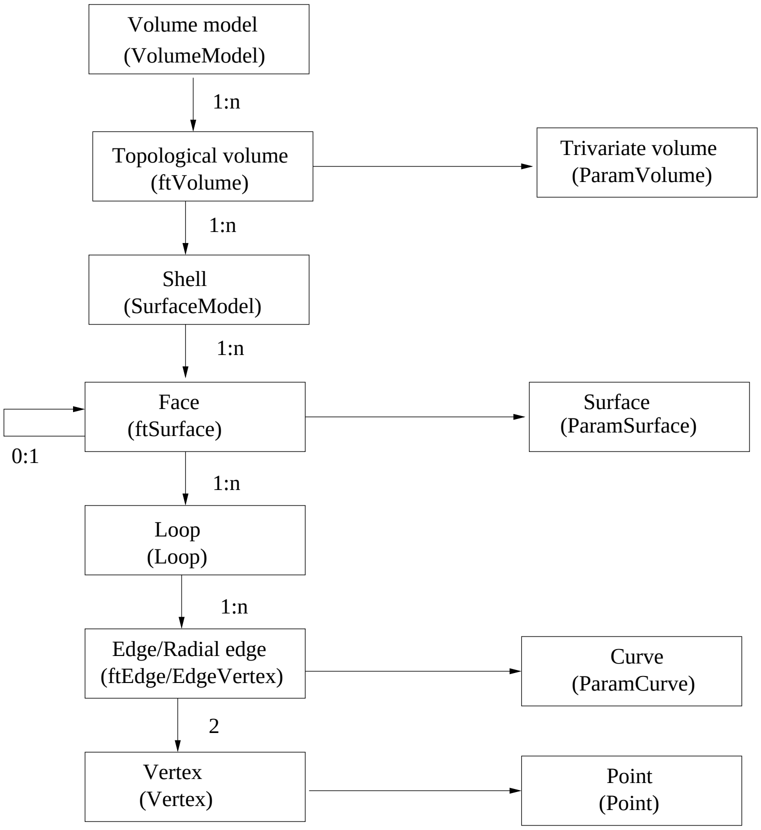

Figure 1: Topology structures for a volume model. The GoTools class names are given in brackets

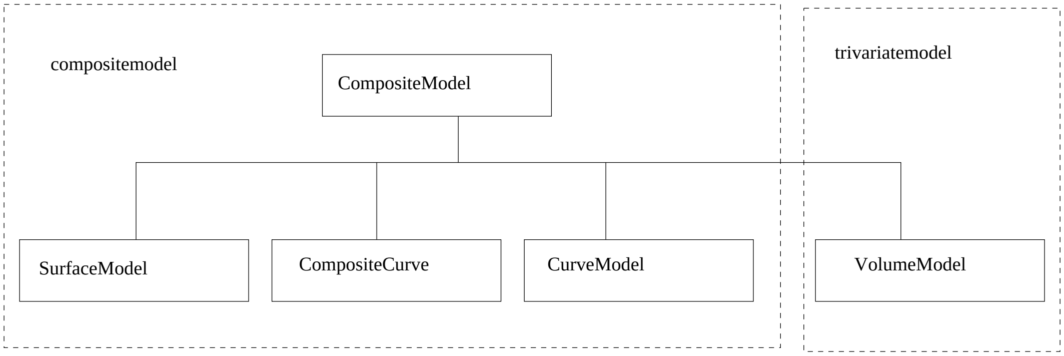

Figure 2: The inheritance tree for a volume model

which have a geometric representation as a ParamSurface. ParamSurface is described in the module gotools-core.

A face is limited by a number of loops that are sequences of edges (ftEdge). In this context, the face is used to represent adjacency between volumes, see section 5. An edge is limited by two vertices.

3 Volume Model

VolumeModel is a composite model as illustrated in figure 2. See also the description in the documentation of composite model. As such it inherites a function interface, but some methods are not implemented. Thus, the class is incomplete, but not in the sense of a topological entity.

VolumeModel has the following functionality:

- Fetch one entity in the set, either as a topological or geometric volume.
- Evaluate the volume model given information about which entity to evaluate
- Compute bounding box
- Add a new volume to the volume model
- Check if all geometrical volumes are NURBS
- Fetch all vertices in the model

- Fetch all radial edges in the model, see 6.
- Fetch non-radial edges, i.e. edges that does not belong to at least two volumes.
- Fetch all inner faces in the model
- Fetch all boundary faces in the model
- Check if the model has a corner-to-corner configuration
- Ensure that the model has a corner-to-corner configuration
- Ensure that adjacent spline volumes share common spline spaces at common boundaries
- Fetch the boundary of the volume model described as a number of SurfaceModels
- Divide the volume model into connected volume models

The constructor of the volume model requires a number of tolerances, namely gap, neighbour, kink and bend. These tolerances are the same as the ones required for a composite model, and the tolerances are explained in the documentation of the compositemodel module.

4 Topological Volume Entity

The topological volume entity is implemented in the ftVolume class and it inherits Body from the compositemodel module. The name ftVolume is choosen to be in line with ftSurface and ftEdge. Those entities have got their names for historical reasons. The prefix *ft* has no deeper meaning.

An ftVolume is a Body and has thus access to its shells, i.e. boundaries, and can check whether two ftVolumes are adjacent. An ftVolume can also

- Fetch the corresponding geometry volume
- Fetch the bounding box of this volume
- Fetch all adjacent ftVolumes

- Fetch all radial edges belonging to this volume or being common between this volume an another volume

- Fetch edges belonging only to the current volume

- Compute adjacency information between the current volume and another volume

- Return information about outer boundaries

- Get local information about correspondence of coefficients of two adjacent spline volumes

- Return the relation between the current volume and a given vertex

5 Extensions to Face

The initial boundary representation implementation of a face, was no longer sufficient when the entity should serve as part of the boundary of a volume belonging to a volume model. Some extensions turned out the be required. The face (ftSurface) has knowledge about the Body it belongs to, if any. It has also a pointer to an adjecent face. This pointer is used in the context of a volume model. During the assembly of a volume model, information about coincident boundary faces of the ftVolumes are computed, and the pointer representing adjacency is set accordingly.

An ftSurface provides access to its twin and to the at most two bodies sharing the common face represented by this ftSurface.

When the ftSurface represents the boundary surface of an ftVolume, the corresponding parametric surface will be of type SurfaceOnVolume. This class is implemented in the module trivariate. A SurfaceOnVolume has knowledge about the spatial representation of the surface, the ParamVolume on which it belongs and the position of the surface in the parameter domain of the volume. Currently, a SurfaceOnVolume is constructed only as a boundary surface of a SplineVolume. It contains enough information to identify which volume boundary it represents and the orientation of the surface compared to this boundary.

6 Radial Edge

In a non-manifold model, more than two faces can meet in an edge, and thus the half edge representation implemented in ftEdge is not sufficient to hold the model. The radial edge (EdgeVertex) is an extension to the topology structures of compositemodel. An EdgeVertex contains information of all pairs of half edges (ftEdge) meeting in an edge, and each ftEdge belonging to an EdgeVertex has access to this EdgeVertex.

An EdgeVertex instance is defined when adjacency between two ftSurfaces is found. Thus, at edges in an ftVolume where no adjacent ftVolume exists, no EdgeVertex instance will be defined. Then the edge is represented with a pair of ftEdge instances.

The EdgeVertex class is placed in the module compositemodel, but is used in connection with volume models. In addition to functionality related to topology build, the class has some access functionality:

- Fetch edges meeting in the radial edge, either uniquely defined, i.e. one instance for a pair or twins, or absolutely all edges.
- Fetch a specified edge
- Check if a ftEdge belongs to a radial edge
- Check if a ftEdge belongs to a radial edge and has no information about a corresponding ftEdge. That means that the corresponding surface does not belong to a volume.
- Fetch adjacent faces
- Fetch adjacent bodies