

Rigid motion applied to a Family for a set of Zones

The proposal is a modification of the *Family_t* node. The modification re-use existing CGNS structures, but it implies some constraints to *Zone_t* node and sub-nodes. For time dependent data, this extension is also related with extension ‘**Modification of BaseIterativeData and ZoneIterativeData for reference frame and family motion**’.

First, we propose to allow the *RigidGridMotion_t* node under the *Family_t* node. The application of the motion is distributed on each zone having a *FamilyName_t* FamilyName value corresponding to the *Family_t* node name.

```
Family_t :=
{
  List( Descriptor_t Descriptor1 ... DescriptorN ) ;           (o)
  FamilyBC_t FamilyBC ;                                       (o)
  List( GeometryReference_t
        GeometryReference1 ... GeometryReferenceN ) ;       (o)
  RotatingCoordinates_t RotatingCoordinates ;                (o)

  List( RigidGridMotion_t RigidGridMotion1 ... RigidGridMotionN ) ; (o)

  List( UserDefinedData_t UserDefinedData1 ... UserDefinedDataN ) ; (o)
  int Ordinal ;                                             (o)
} ;
```

Side-effect:

We require that if motion is specified in *Family_t*, that each *Zone_t* which is a member of the family should have a *FamilyName_t* attribute named *FamilyRigidMotionName* which points back to the *FamilyRigidMotion* node. When an application has to determine whether there is a *FamilyRigidMotion* or not, this application only needs to check for the existence of a *FamilyRigidMotionName* attribute using a per-Zone loop, instead of parsing all families and making the match with zones.

```
Zone_t :=
{
  ...other SIDS Zone_t attributes...

  FamilyName_t FamilyName                                     (o)
  FamilyName_t FamilyRigidMotionName;                     (o)

  ...other SIDS Zone_t attributes...
} ;
```

Remarks:

1- It is the responsibility of the user application to find if the Zone has a Family with time iterative data or not. In the case of a time-dependant rigid motion declared at the *Family_t* level, no *RigidGridMotionPointers* would be found at the *Zone_t* level, only the *Family_t* attribute which points to the Family motion.

2- The use of a Family rigid motion in at least one Family of a *CGNSBase_t* implies no Motion node in all *Zone_t* of this *CGNSBase_t* (i.e. no rigid or arbitrary motion). The RigidMotion for the *Zone_t* is instead defined by the *FamilyRigidMotion_t*.