FELyX – The Finite Element library eXperiment

Tutorial

May 2005

EVEN – Evolutionary Engineering AG
Leonhardstr. 25, LEC B14
CH-8092 Zurich
Switzerland

Phone : +41 (0)44 633 3532
Fax : +41 (0)44 632 1702
Mail: info@even-ag.ch
WWW: www.even-ag.ch

Contents

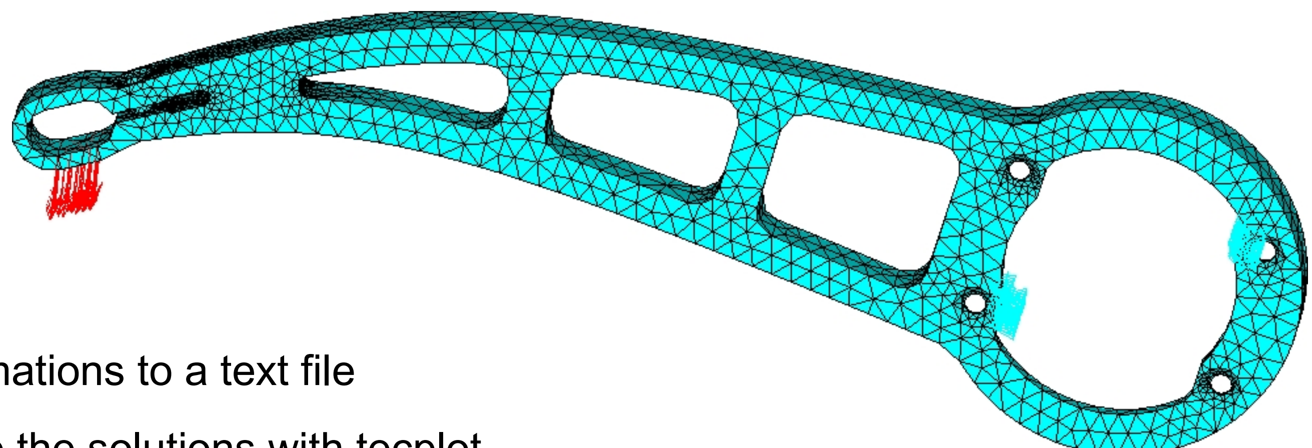
- Session 1:
  - Get to know the entire development environment
  - Using StructObject to solve a structural FE-Problem, the first application using FELyX
  - Write standard output text files
- Session 2:
  - The structure of FELyX / Using FELyX to solve different FE-disciplines
  - Derive StructObject / Build a custom application
- Session 3:
  - Enhance the capabilities by reading own input files
  - Read own input-files using the boost::spirit library
- Session 4 / 5
  - Element formulation
  - Discuss defined problems

Session 1

- The structural finite element problem
  - What we will problem to be solved
- Installation
  - What products are needed
  - Which ones are recommended
  - How to install them
- Using StructObject in a Program
  - Building a first executable which solves a FE-Problem

- The Problem to solve:
  - A cantilever beam under the shown loadings
  - The model is fully defined within ANSYS and written to a database using

```
Preprocessor >> Archive Model >> Write
```



- What is to be done:
  - Write the nodal deformations to a text file
  - Write a file to visualize the solutions with tecplot

Session 1

- Installation
  - The following tools and libraries are required
    - CVS
    - GNU build tools
    - Boost libraries version 1.32
  - The easiest to get these things (using SUSE linux 9.3):
    - Install SUSE linux 9.3
    - Include the development tools of SUSE 9.3
    - Include boost 1.32
    - Install linCVS from lincvs.org
- Get the FELyX sources
  - Under http://sourceforge.net/projects/felyx/ is all the information regarding the project
  - Type the following on the commandline or use lincvs to get the sources

```
cvs -z3 -d:pserver:anonymous@cvs.sourceforge.net:/cvsroot/felyx co -P felyx
```

Session 1

- KDevelop – the programming environment
- First application
  - Start KDevelop and build a hello_world project
- Starting with FELyX
  - FELyX is organized as a KDevelop project
  - Build the project
  - Build the documentation
  - Build a new target within the tutorial folder and add a file to the target
  - Add the used libraries

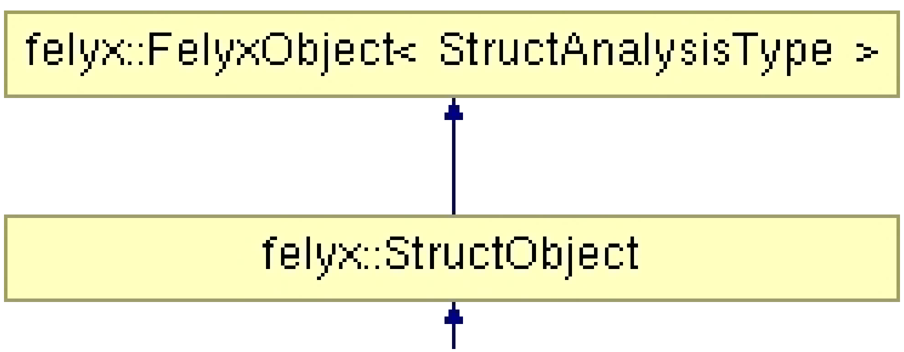
- Solving the problem of the cantilever – beam using StructObject
  - The Object StructObject is the base object for all structural calculation, it is derived from FelyxObject
  - The needed functionality is
    - Load the ANSYS database file
    - Solve the problem
    - Write some output to the screen / a file

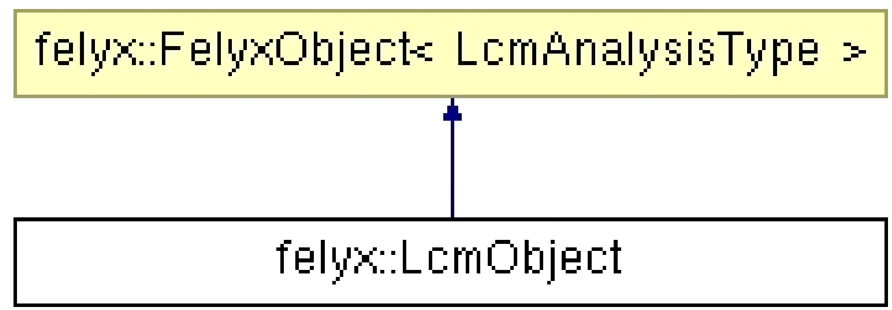
Contents

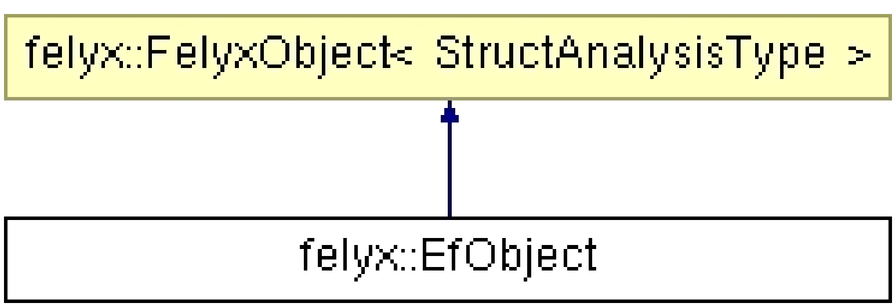
- Session 1:
  - Get to know the entire development environment
  - Using StructObject to solve a structural FE-Problem, the first application using FELyX
  - Write standard output text files
- **Session 2:**
  - **The structure of FELyX / Using FELyX to solve different FE-disciplines**
  - **Derive StructObject / Build a custom application**
- Session 3:
  - Enhance the capabilities by reading own input files
  - Read own input-files using the boost::spirit library
- Session 4 / 5
  - Element formulation
  - Discuss defined problems

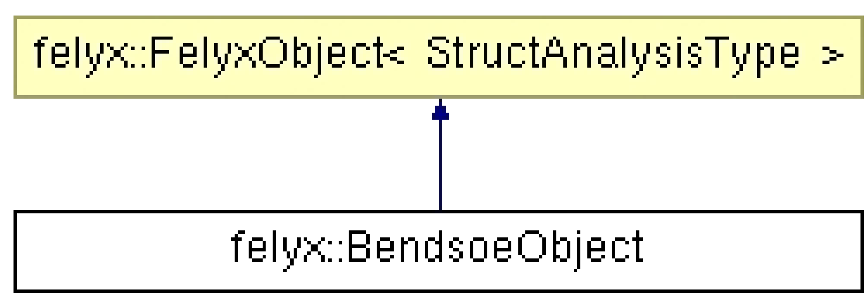
Session 2: Handling different physical disciplines in FELyX

- Implemented as derived FELyX objects
- Parameterized by a AnalysisType class

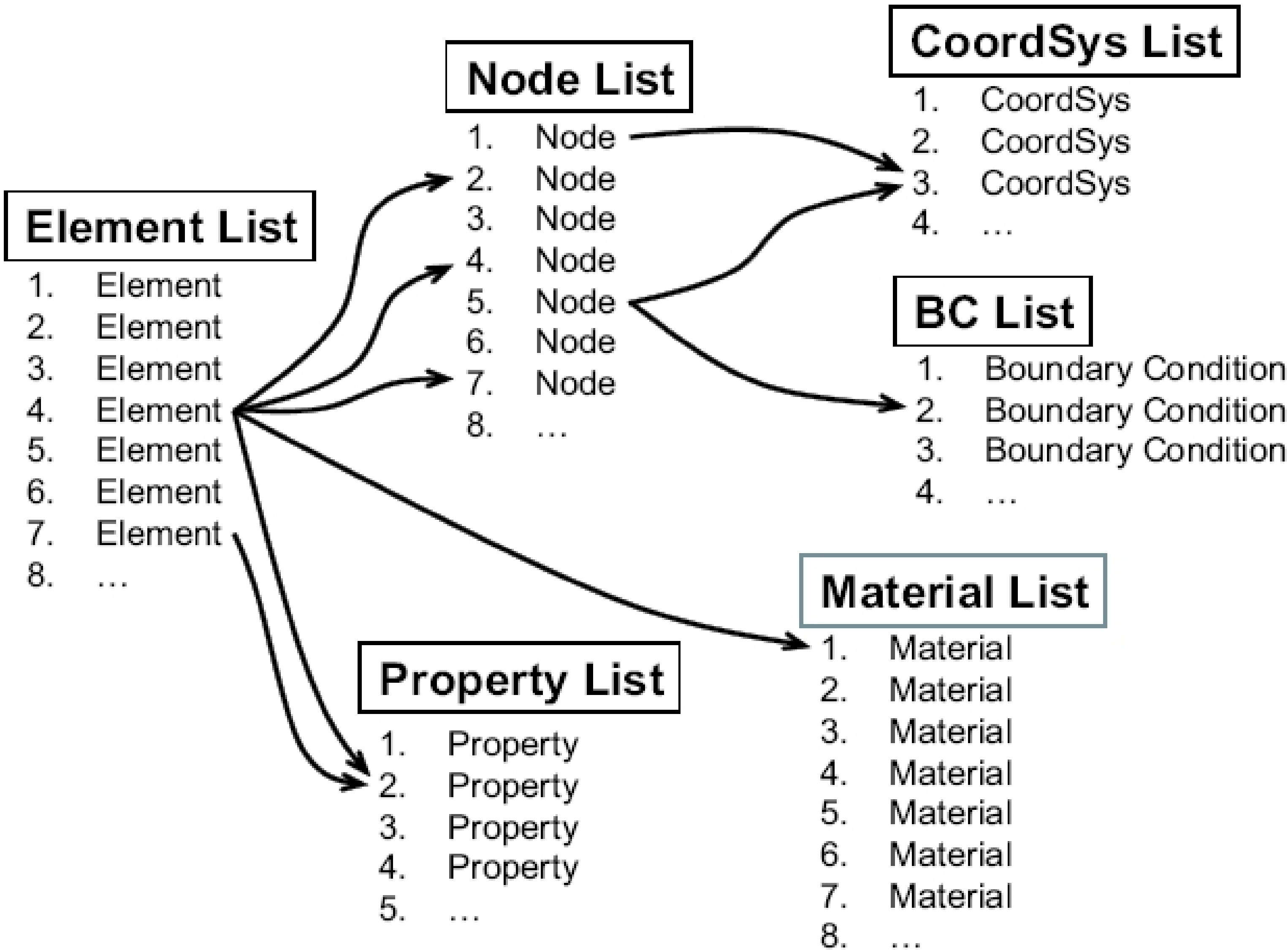






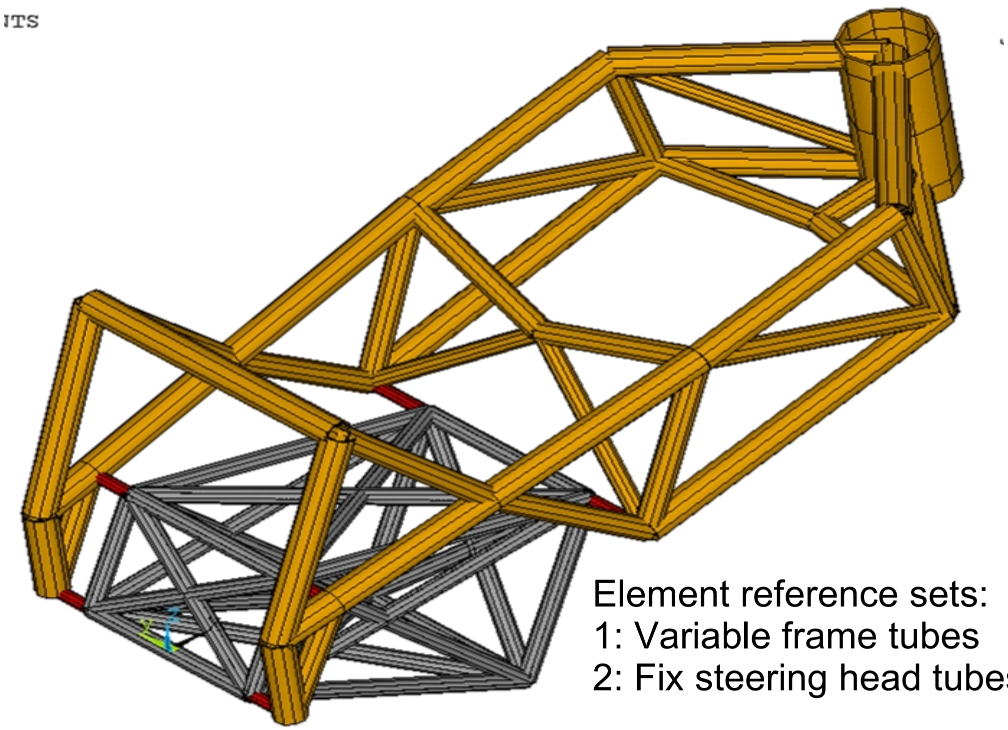


Session 2: Organization Finite Element entities in FELyX



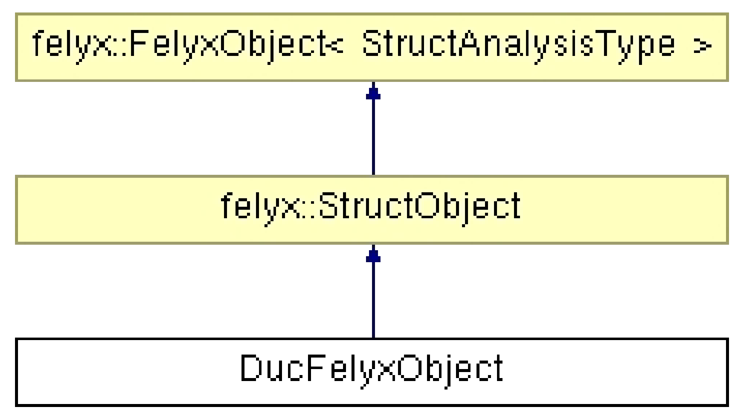
Session 2:
Implementation of a derived StructObject for the analysis of a ducati frame

- Tasks:
  - Implement a custom FELyX object to analyze the ducati frame
  - Eval mass of frame
  - Vary tube dimensions of frame (PropertySets 0-14)
  - Evaluate maximum stress in frame for a braking loadcase
  - Evaluate torsion stiffness of frame

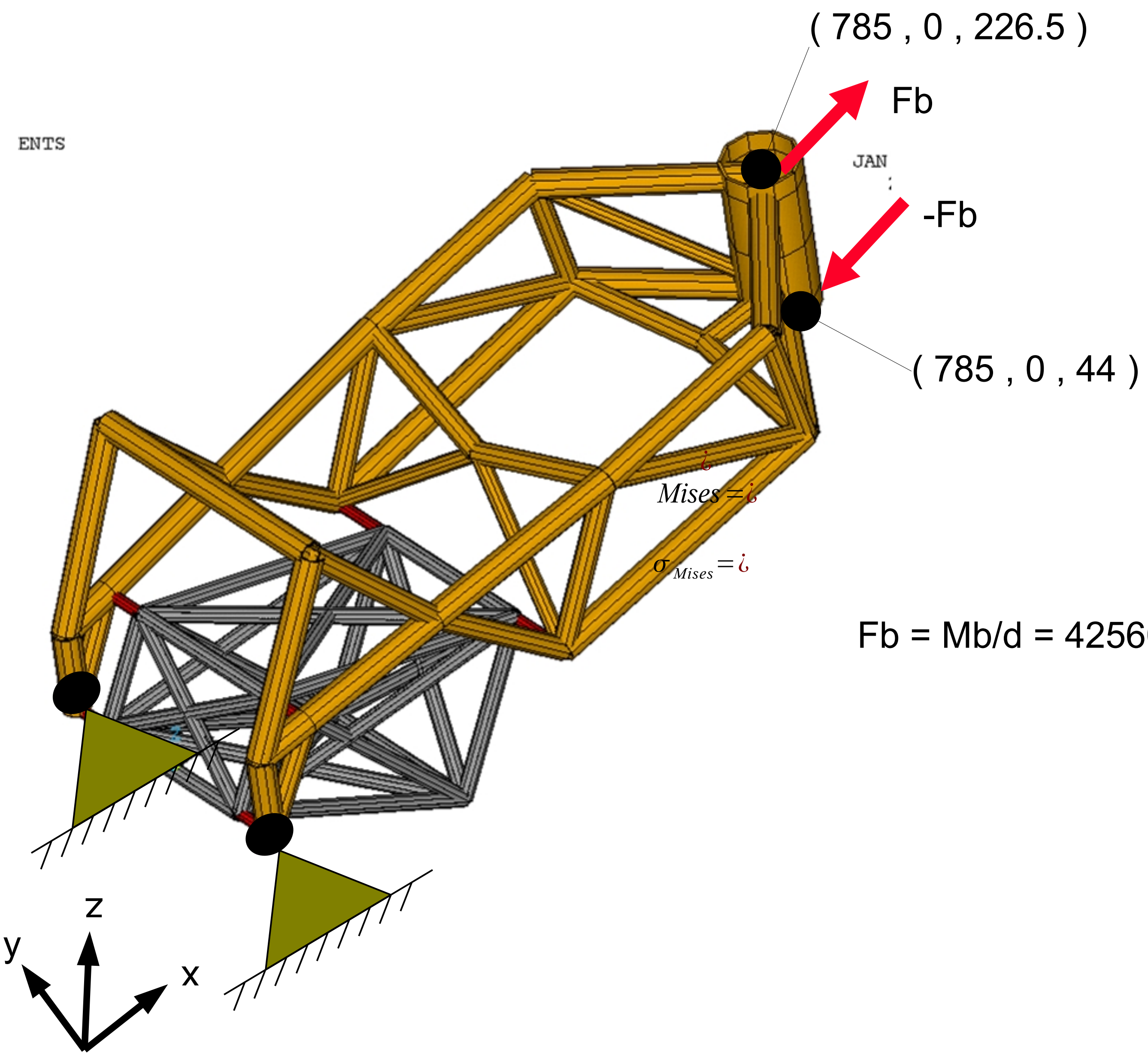


Element reference sets:
1: Variable frame tubes
2: Fix steering head tubes

Key values of original frame:
Weight = 7.4 kg
Torsion stiffness = 1310 Nm / degree
Max. stress for braking = 450 N/mm2

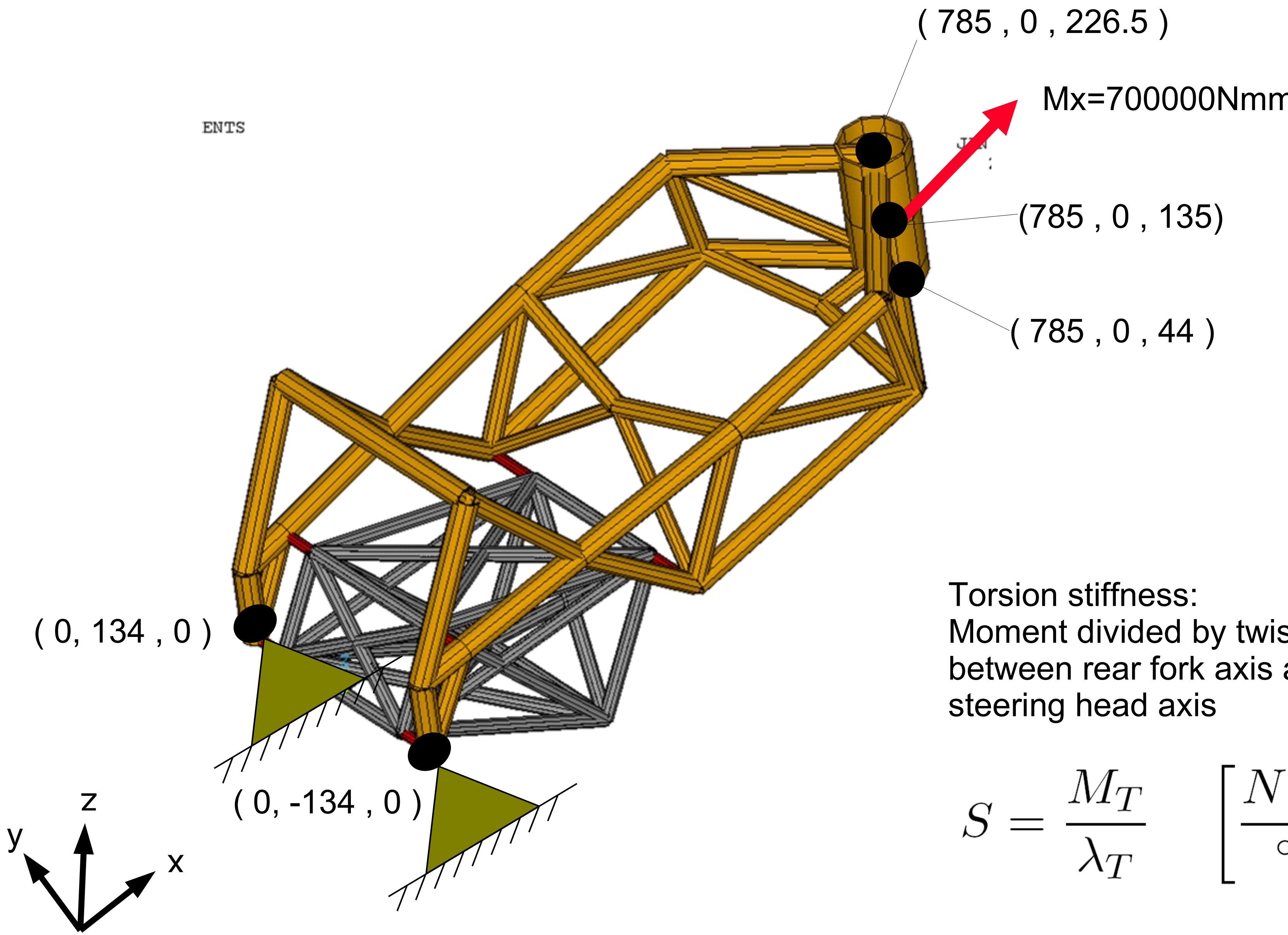


Session 2:
Definition of braking loadcase



Fb = Mb/d = 4256000 Nmm / 182.5 mm

Session 2:
Definition of torsion loadcase / stiffness



Torsion stiffness:
Moment divided by twist angle between rear fork axis and steering head axis

$$S = \frac{M_T}{\lambda_T} \qquad \left[\frac{Nm}{\circ}\right]$$