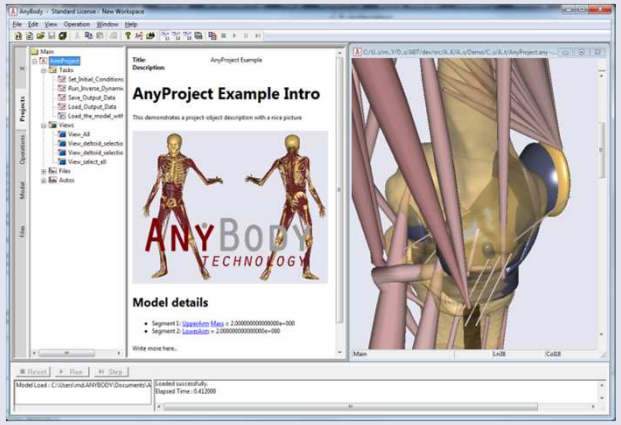
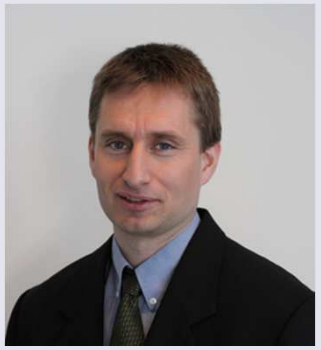
Preview of
The AnyBody Modeling System
version 5.1



Presenter
Michael Damsgaard
Head of Development, Ph.D.



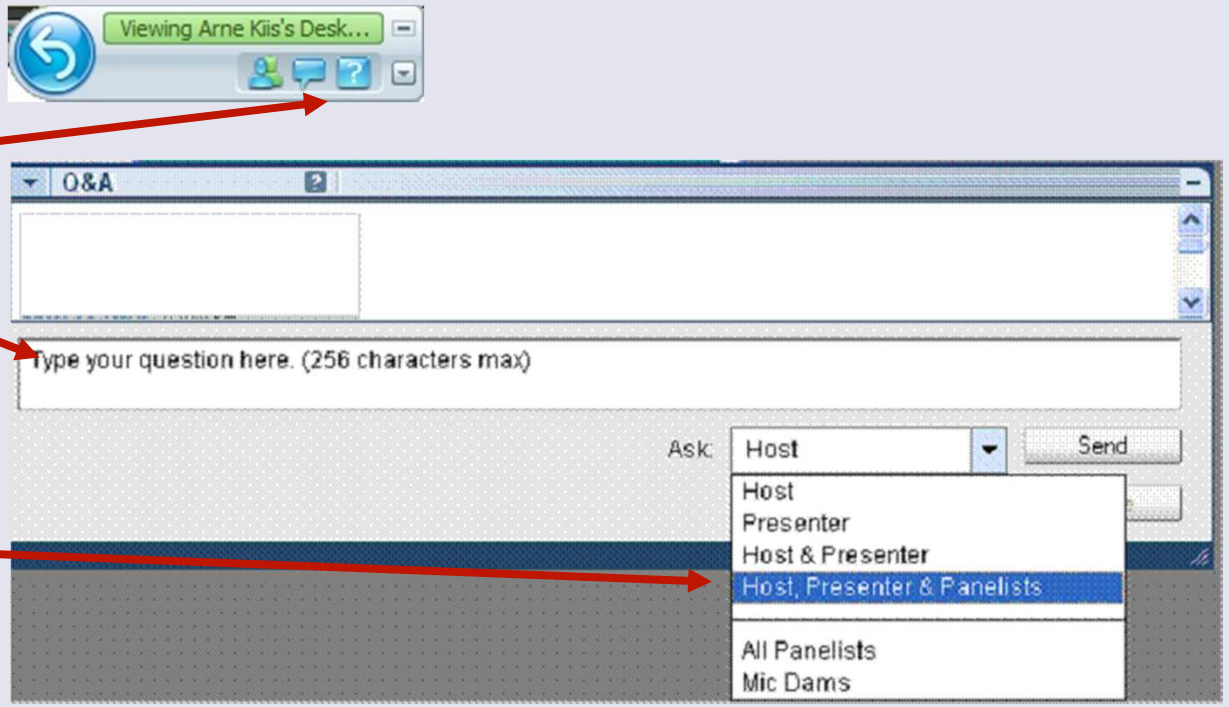
Host:
Arne Kiis
Sales Manager

The web cast will start in a few minutes….

ANYBODY TECHNOLOGY

Q&A Panel

- Launch the Q&A panel from the menu bar.
- Type in your question.
- Send your question to ”Host, Presenter & Panelists”



Notice the answer displays next to the question in the Q&A box. You may have to scroll up to see it.

ANYBODY TECHNOLOGY

Introduction

- The AnyBody Modeling System, v.5.1 (August, 2011)
- Previous versions focus has been on the entire body:
  - Comprehensive full-body modeling
  - Advanced facilities for driving the model kinematically
  - Muscle recruitment and many other simulation facilities
  - Overall body performance, ergonomics, sports, etc.
- Version 5.x and in particular version 5.1, we focus more on details
  - Advanced joint modeling
  - Patient-specific modeling
  - Improved interfaces to related softwares such as FEA and medical image processing tools
  - Orthopedics: Implant design, injury and trauma, surgical planning
  - UI:
    - Wrapping of the complicated models
    - Processing of many trials

ANYBODY TECHNOLOGY

Contents

- Introduction
- Features for advanced joint modeling
  - Surface contact model
  - Enhanced force-dependent kinematics solver
- New and updated interfaces
  - Hook for external code (C++ or Python)
  - Improved FEA interface for Ansys and Abaqus
  - Interface to Mimics, Materialise
- GUI features
  - User-defined documentation in HTML-based views in AMS GUI
  - Drawing Widgets
- Models

ANYBODY TECHNOLOGY

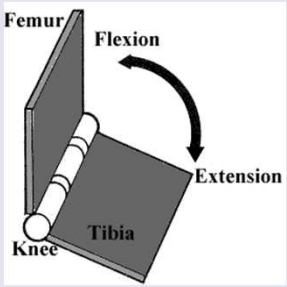
Features for Advanced Joint Modeling

- Surface contact model
- Enhanced Force-Dependent Kinematics Solver

ANYBODY TECHNOLOGY

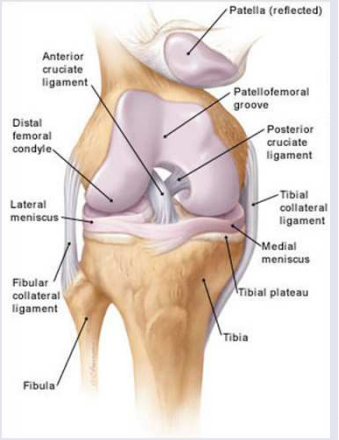
Motivation

Simple/idealized joint modeling:



- Simple (conforming) joints
  - Simple (or simplified) kinematics
  - Bilateral reactions
  - Infinite stiffness (Ligaments are not important or ignored)
- Purpose:
  - Simple and fast modeling with data available
  - Overall/general biomechanics
  - Ergonomics assessments

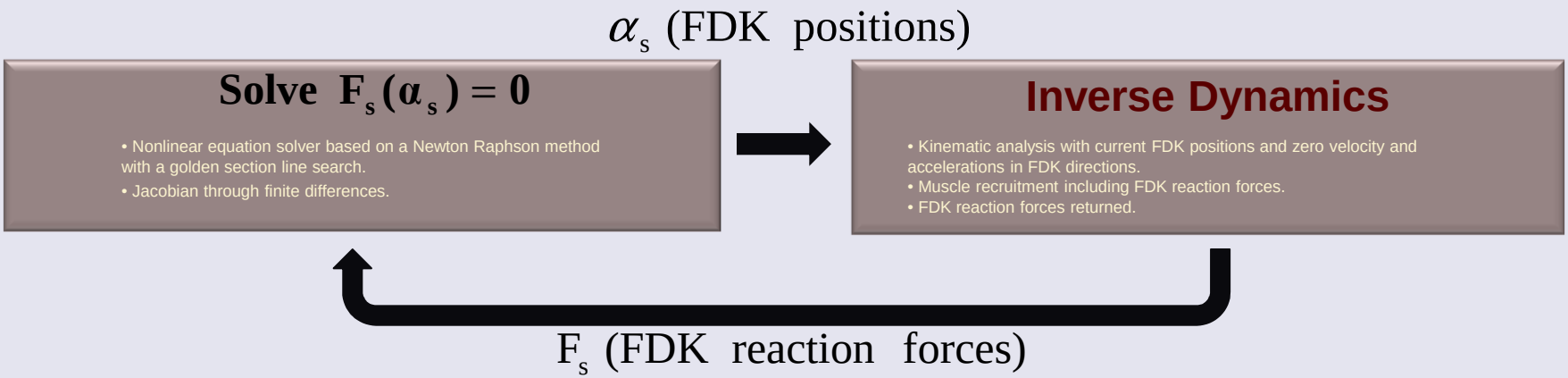
Advanced joint modeling:



- Non-conforming joints
  - Contact (unilateral reactions)
  - Clearance (unilatertal kinematics)
  - Sliding
  - Elasticity (ligament flexibility allow significant motion)
- Purpose:
  - Correct modeling of complicated joints -> detailed biomechanics
  - Injury, implant and surgery modeling

ANYBODY TECHNOLOGY

Force-Dependent Kinematics

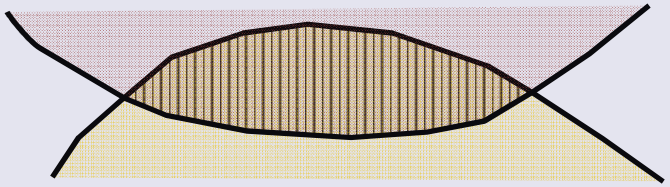


- Objective: *A simpler and more efficient way solution than forwards dynamics using reasonable assumptions*
- Assumptions:
  - FDK degrees of freedom experience small motion
  - Dynamics of these are neglected, i.e., velocity and acceleration is set to zero.
  - Positions are governed by elastic structures (ligaments, contact surfaces, …)
  - Friction is neglected
- Enhancements in v.5.1
  - Improved equation solver improving robustness, in particular with contact problems
- Examples – Work in Progress!
  - The 2010 Grand Challenge Knee Implant Model
  - Spine Model with facet joints (SpineFX project)

ANYBODY TECHNOLOGY

Surface Contact Model

- Rigid surface contact model
  - STL-based
    - Input is STL files only
    - Mesh has to adequately fine
  - Penetration violation integrated over the contact area
  - Different penetration models is currently being tested



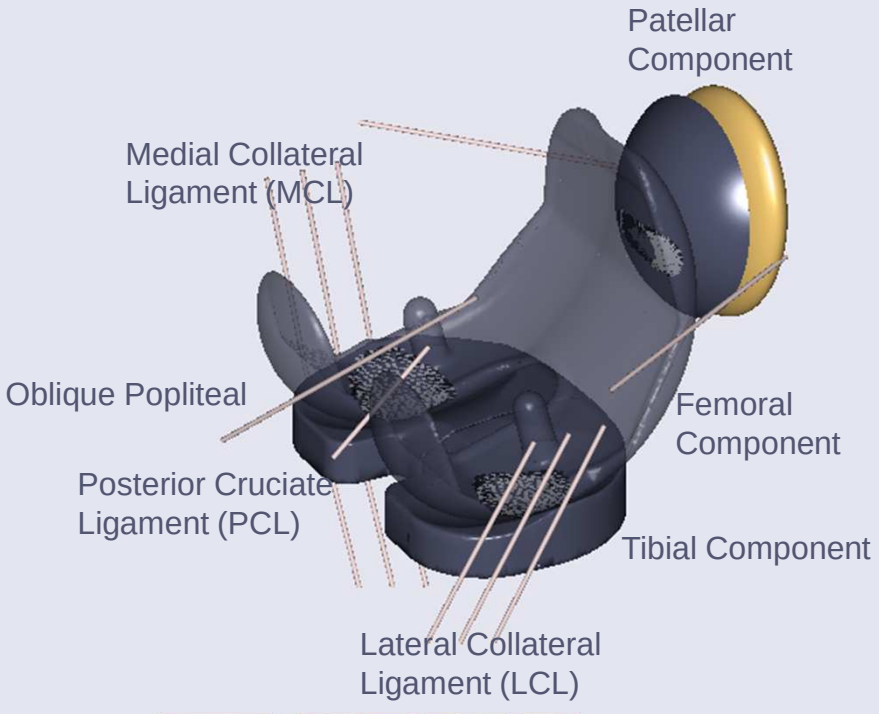
- Objective:
  - To get the kinematics right
    - Center of pressure
    - Muscle moment arms
  - Not to get the right contact pressure or surface stress

ANYBODY TECHNOLOGY

Example: Knee Implant Model

- Data sets
  - 2010 Grand Challenge (Fregly et. al.)
  - The TLEM model from AnyBody model repository

- Developed by Michael Skipper Andersen, Aalborg University
- References:
  - Webcast, April 5, 2011
  - ISB2011: Michael S. Andersen & John Rasmussen, Abstract #343



Artificial elastic structures for FDK solver:
- Artificial patella ligaments
- Artificial tibial-femoral stiffness

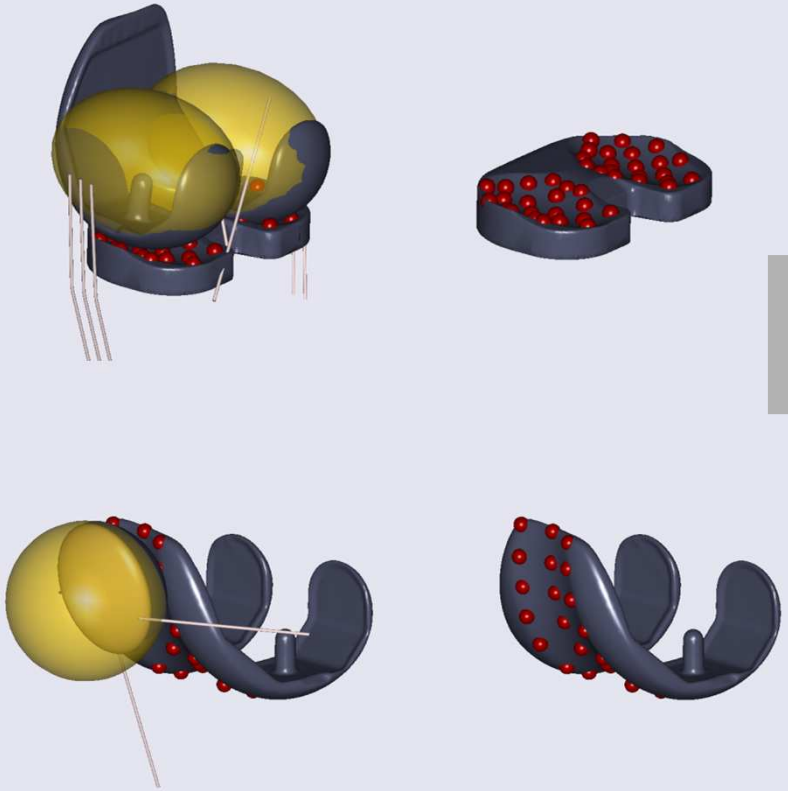
Anterior Cruciate Ligament (ACL) is removed

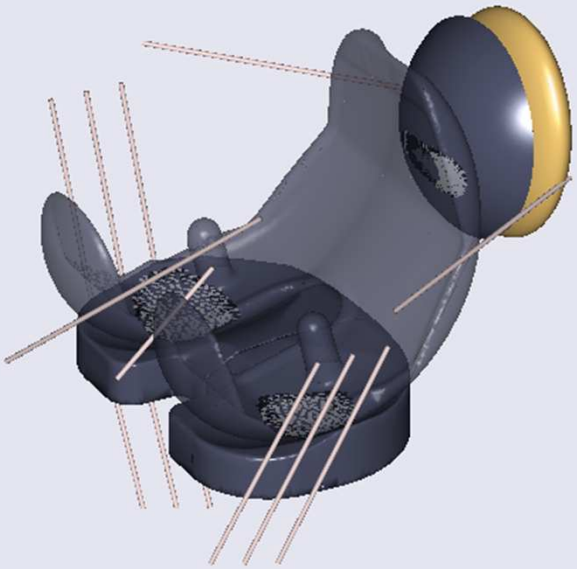
ANYBODY TECHNOLOGY

Simplified Modeling Workflow

AnyBody, v.5.0

AnyBody, v.5.1

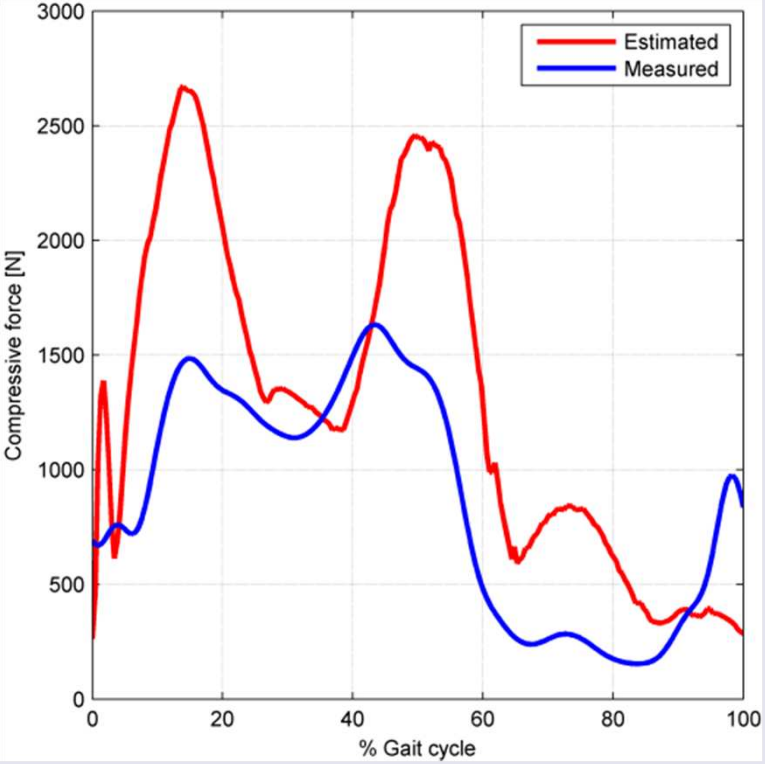




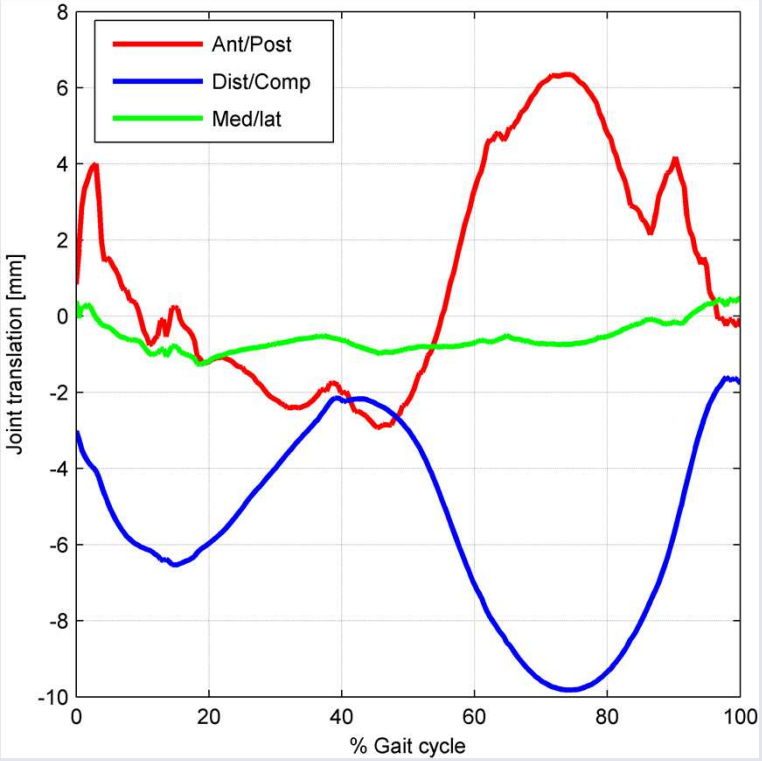
AnyBody TECHNOLOGY

Results – work in progress

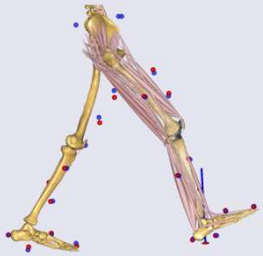
Total compressive force



Joint translations



Case: Normal gait

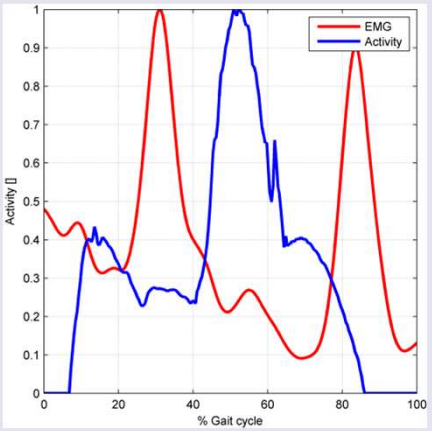


ANYBODY TECHNOLOGY

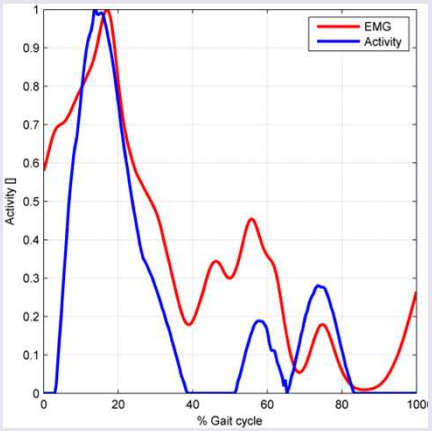
Results – work in progress

EMG results:

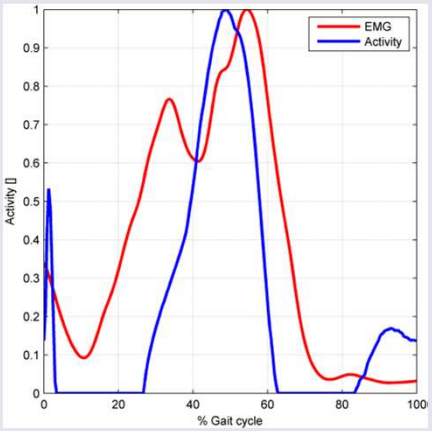
Rectus Femoris



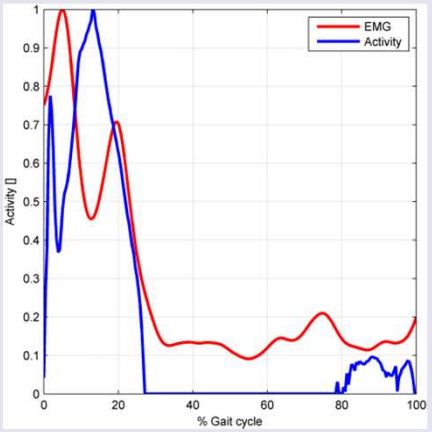
Vastus lateralis



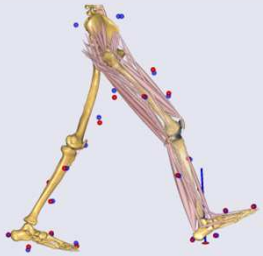
Gastrocnemius



Gluteus Maximus



Case: Normal gait



ANYBODY TECHNOLOGY

“AnyFunction Hook”

for C++ and Python

ANYBODY TECHNOLOGY

What is AnyFunction?

- Features:
  - Functions in AnyScript are also objects, global or in-model
  - Return a single value
  - Evaluated by the AnyScript expression engine
  - Arguments can be any AnyScript data type
  - Multiple list of arguments (Polymorphism)
    - AnyFunction objects consist of “mono-function" objects

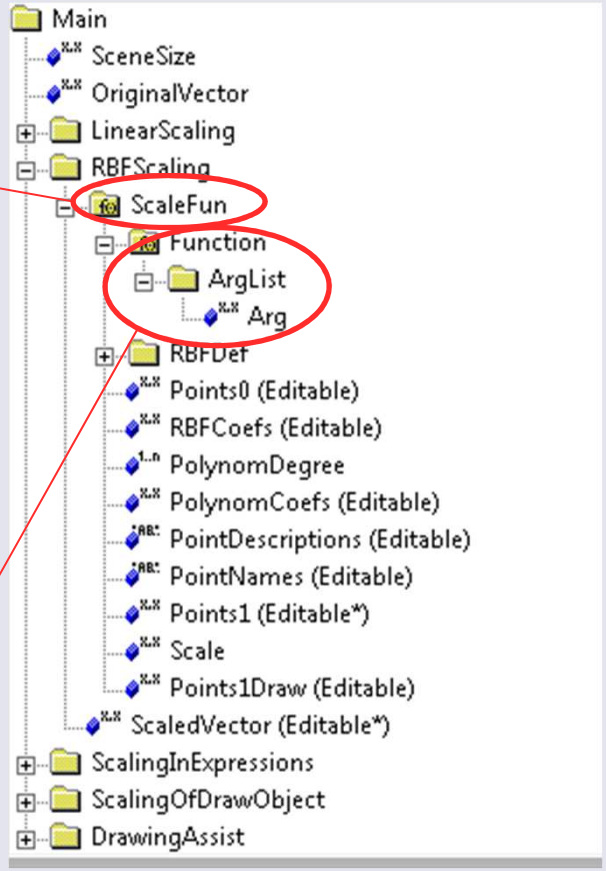
- Examples
  - Standard functions: Sine, cosine, …
  - In-model functions:
    - Interpolation functions used for driving motion or forces
    - 3D transformation functions used for scaling the models
    - Filter functions

ANYBODY TECHNOLOGY

In-model AnyFunction

Example:

```
AnyFunTransform3DRBF ScaleFun = {
  RBFDef.Type = RBF_Gaussian;

  ...

};


AnyVec3 ScaledVector = ScaleFun({x,y,z});
```



Mono function object
& argument list

ANYBODY TECHNOLOGY

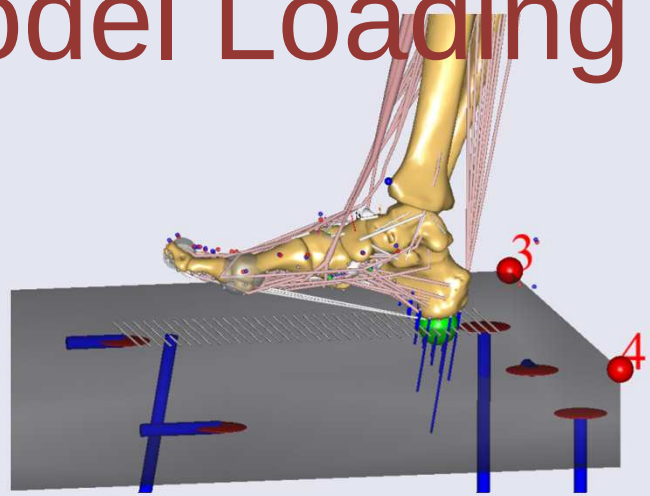
AnyFunction Hook

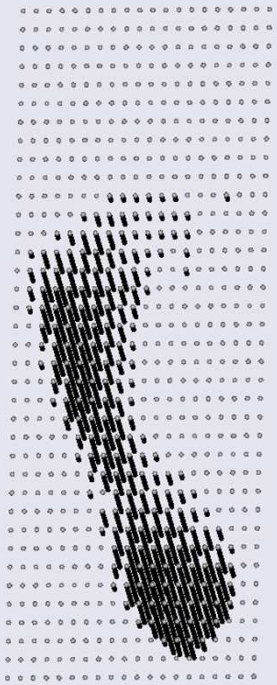
- AnyFunEx is extern-function class
  - In-model function object
  - Function-body is programmed
- Language Support
  - C++ (Compiled into DLL)
  - Python
- Data types
  - Supported: Floating point number, integers, strings
  - Not supported yet: “All other AnyValues”

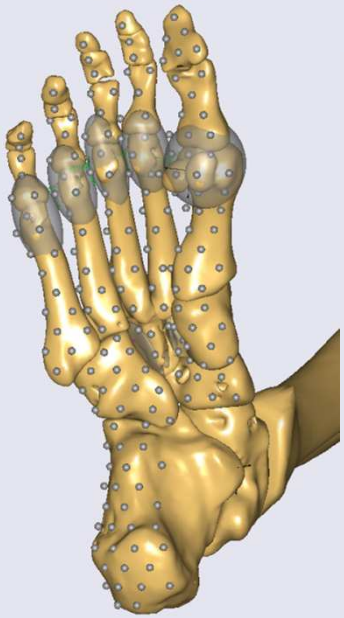
ANYBODY TECHNOLOGY

Example: Foot Model Loading

- Foot model:
  - 26 segments
  - Nodes for load application on each segment
  - References
    - AFootPrint (www.afootprint.eu)
    - ISB2011: Carbes et. al. *A new multisegmental foot model and marker protocol for accurate simulation of the foot biomechanics during walking,* Abstract #183
- Measured loads:
  - Force plate data
  - Pressure grid





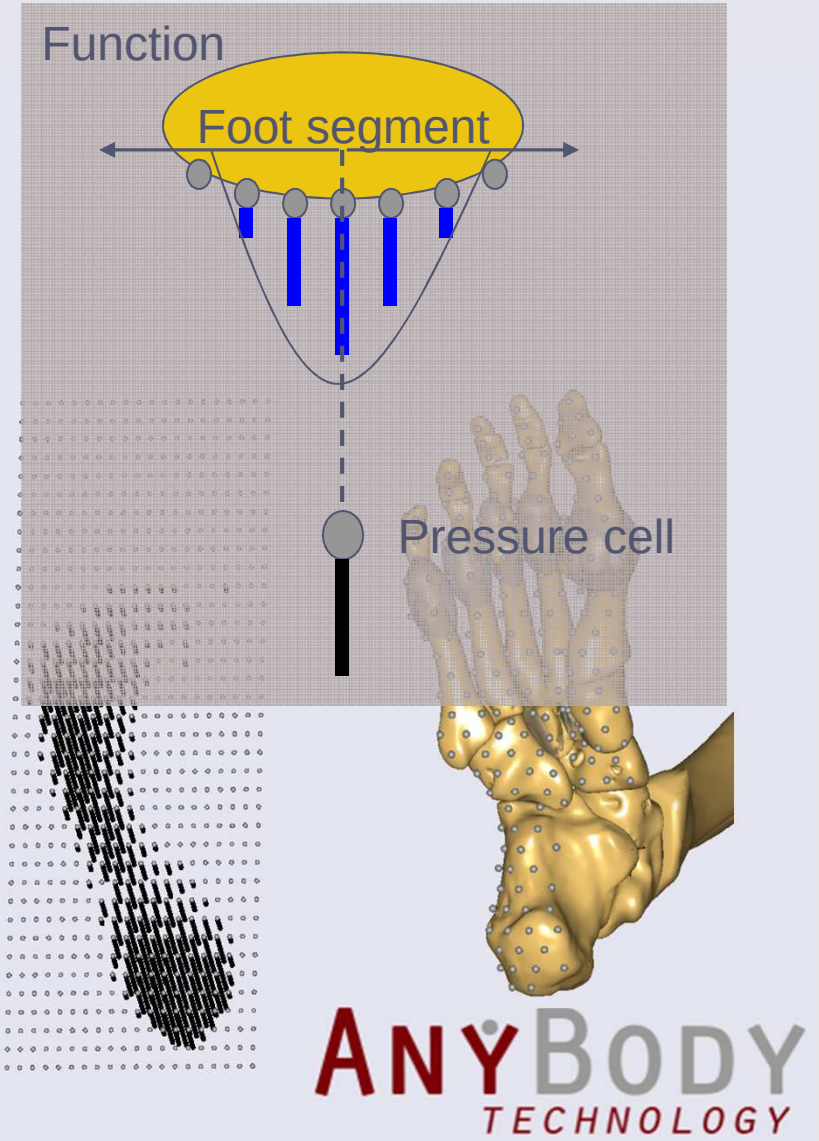


ANYBODY TECHNOLOGY

Example: Foot Model Loading

- The amount of pressure of each cell is distributed to all the force-nodes of the model according to a bell-shape function of the horizontal distance.

- The final output of the calculation is a coefficient representing the percentage of the force plate 3D resultant force to be applied to each node of the foot.



Example: Foot Model Loading

The function implementation

- Function declaration (function object)
  - Function name
  - Return: A vector of coefficients for each node on the foot
- Function implementation (mono function object)
  - Function arguments
    - a matrix with the position in space of all the foot force-nodes
    - a variable with the current time value of the study
  - Function body (here Python)
- Function call (evaluation in expression)

AnyScript Function declaration

```
AnyFunEx PressureFun =
{
  AnyVector Return = .FootPressureNodeCoeffVecInitial;
  AnyFunExMonoPy PressureFunction =
  {
    ModuleFileName = "Final2";
    ArgList =
    {
      AnyMatrix Mat = ...FootPressureNodeMatInitial;
      AnyFloat Time = 0.0;
    };
  };
};
```

Function in "Final2.py":

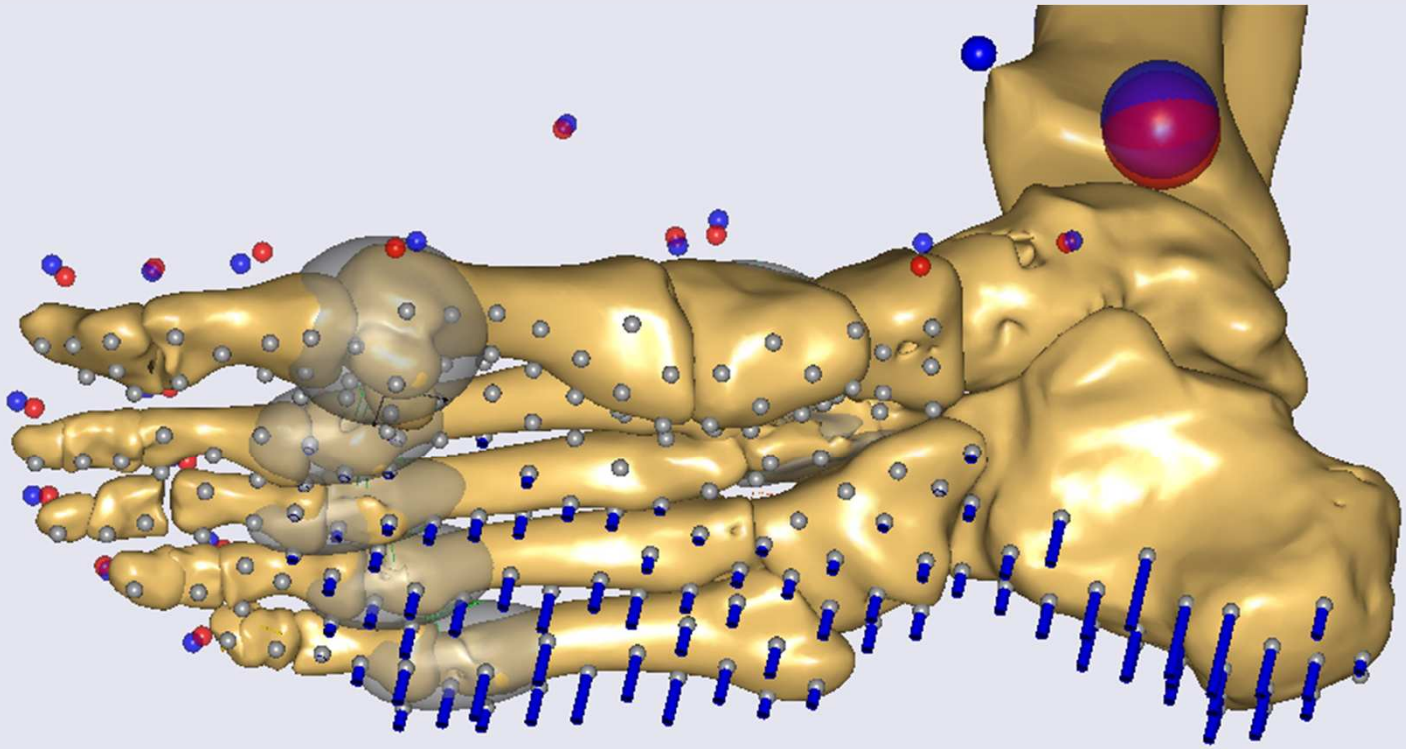
```
def PressureFunction(foot, timestep):

  ...

  return FinalCoeffVecTuple
```

```
AnyVector FootPressureNodeCoeffVec =  PressureFun(FootPressureNodeMat, Main.Studies.InverseDynamicStudy.t);
```

AnyScript Expression Evaluation

ANYBODY TECHNOLOGY

Example: Foot Model Loading



The final result is a serie of 3D forces applied all over the plantar surface of the foot and to all the 26 segments according to the time variation of the recorded plantar pressure.

ANYBODY TECHNOLOGY

“AnyFunction Hook” Perspectives

- Areas of possible usage
  - All sorts of post- and pre-processing of data
  - Force functions that are too advanced for simple AnyScript statements
    - Measurements
    - Extern contact models, e.g. FE-based models
    - User input
- Kinematic drivers is currently not an area of usage.
  - The kinematic engine requires analytical derivatives

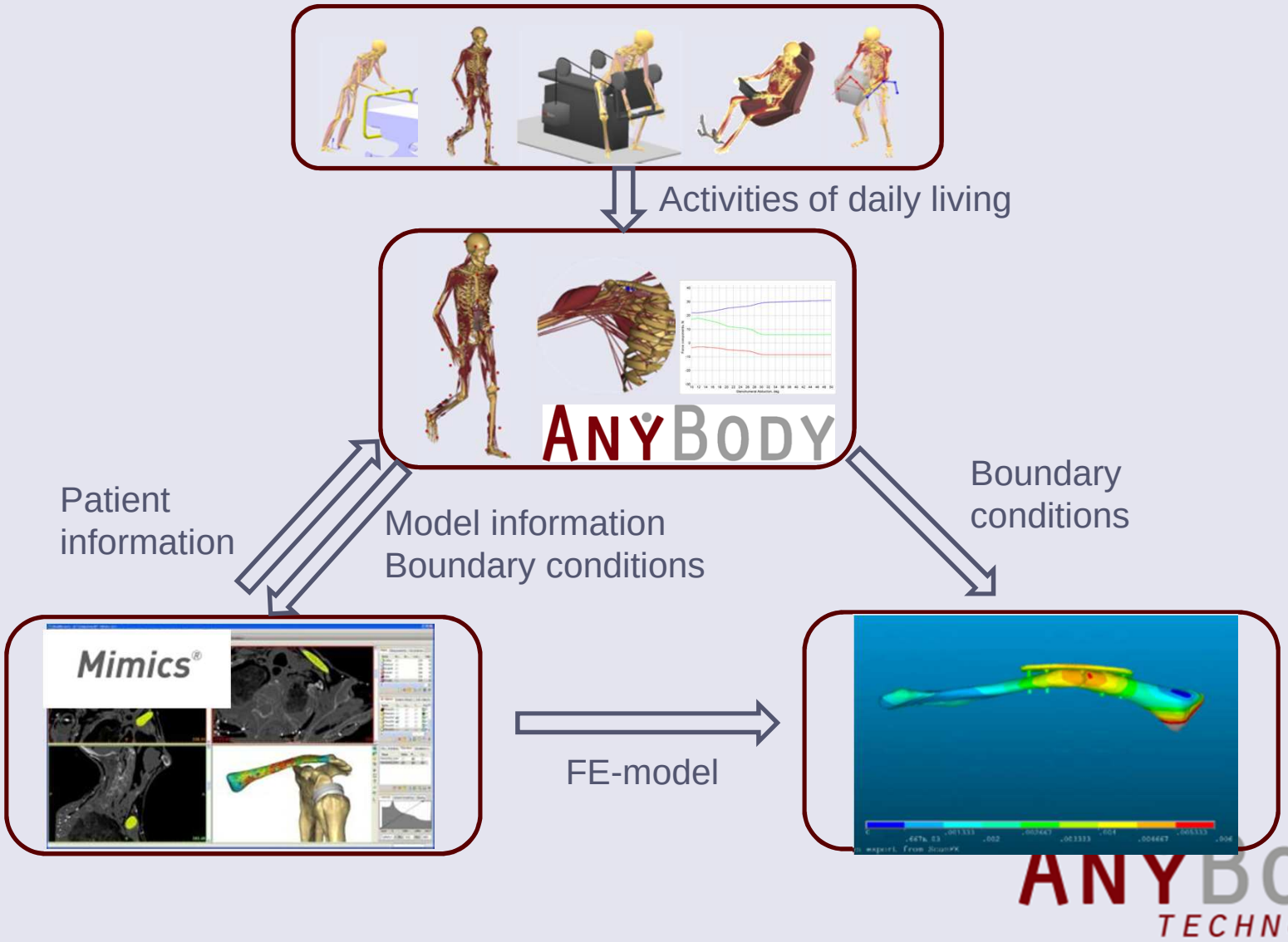
ANYBODY TECHNOLOGY

Software Interfaces

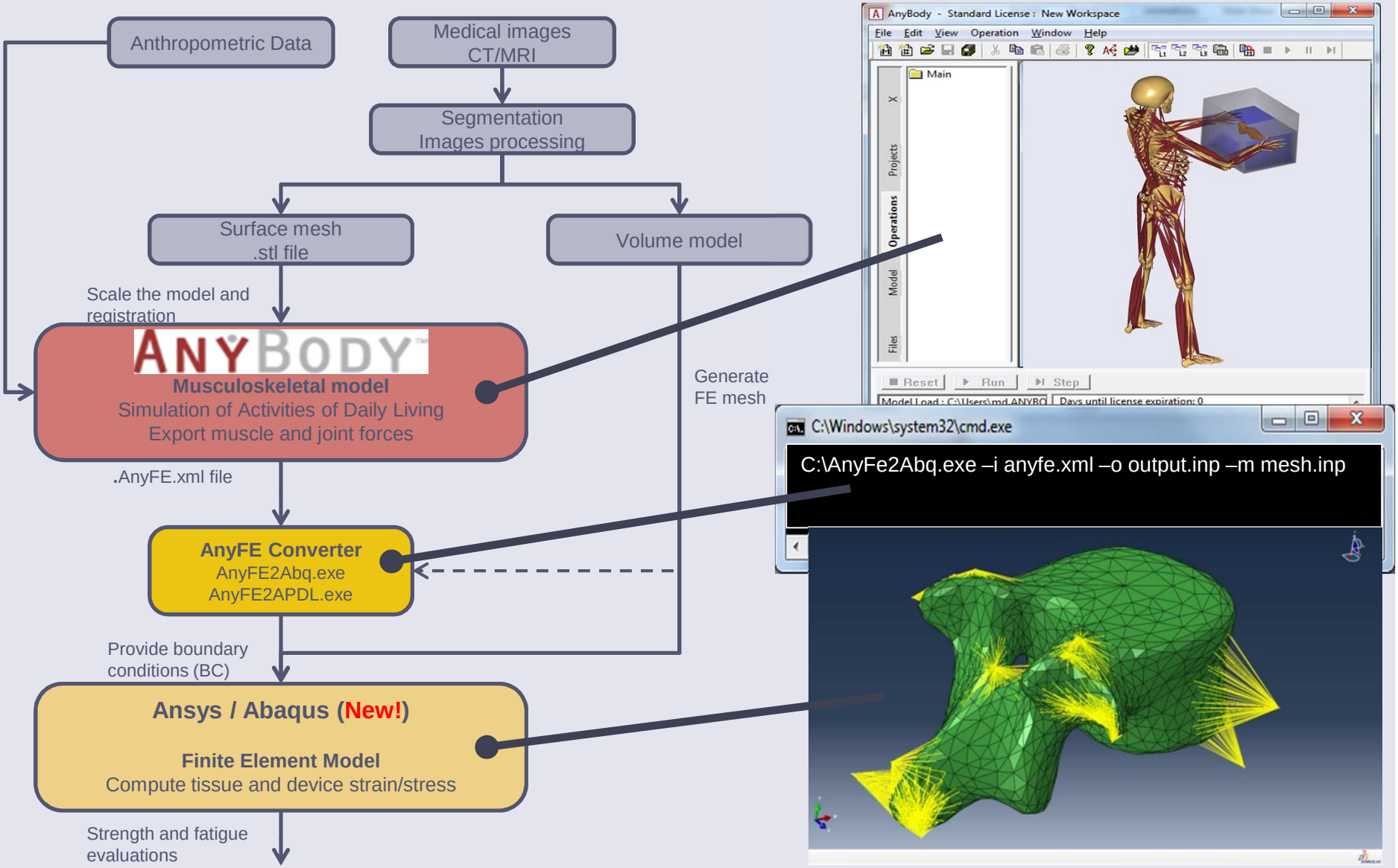
- FEA interface for Ansys and Abaqus
- Mimics interface

ANYBODY TECHNOLOGY

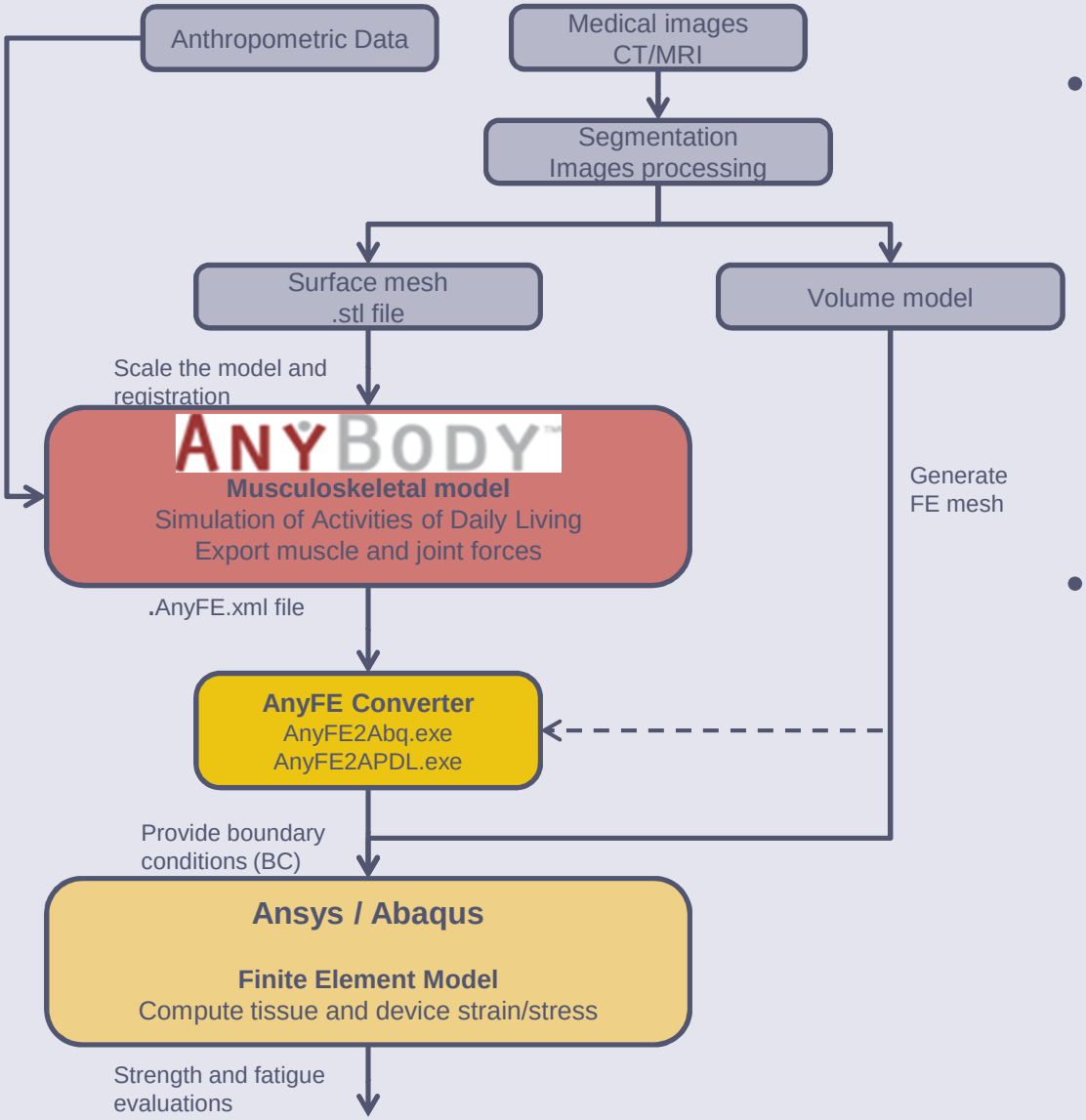
Functional patient-specific modeling



FEA interfaces



FEA interfaces



- AnyFE Converters
  - Ansys
    - Produce APDL code
    - Uses APDL-based template file. The template refers to the mesh file.
  - Abaqus (New!)
    - Produce Abaqus inp-file
    - Based on given mesh file
- News in AMS v.5.1
  - Reference system handling for registration
  - Object naming in output
  - Tutorial update (Chap. 8)

ANYBODY TECHNOLOGY

AnyBody-Mimics Interface

Generation 1:

- Supported by AMS 5.1
- Supported by Mimics 14.1

Activities of daily living

Step 1: Load model

Patient information:

- Patient anthropometry (manual / xml(1)?)
- Landmarks(2) (xml(1) file)
- Segmented bone (STL, manual export?)
- Registration (manual?)

Step 1

Step 4

Step 5: Apply landmarks & simulate forces

Step 2

Boundary conditions:

AnyFE output + Converter to FE

Step 6

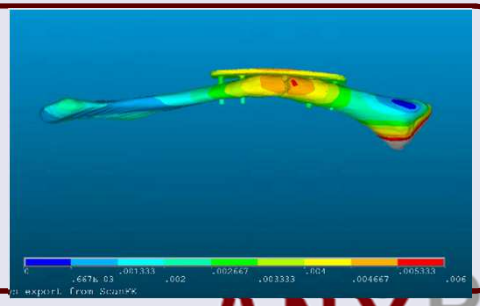
Step 3: Mark bony landmarks

Registration?

FE-model:

- Mesh
- Mesh + BC ?

Step 6



(1) XML file using the common format. (2) Bony landmarks, muscle points, or even skin points.

ANYBODY TECHNOLOGY

AMS GUI enhancements

- User-defined content in AMS views
- Drawing Widgets

ANYBODY TECHNOLOGY

User-defined content in Views

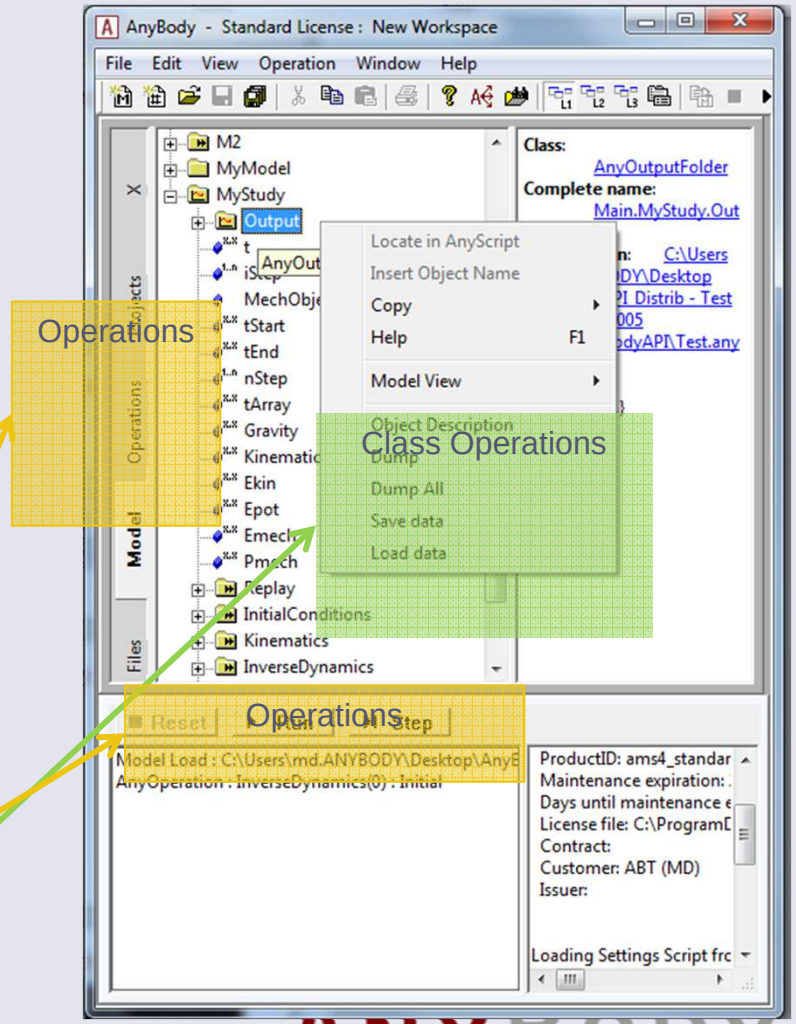
- What's new
  - AnyBody's view with hyperlinks are now HTML-based
- User-defined descriptions
  - Project descriptions (AnyProject)
  - Inline model documentation (Documentation Comments)

- Enhancing your possibilities to …
  - document your model
  - wrap the model functions for other users of the model

User-defined content in Views

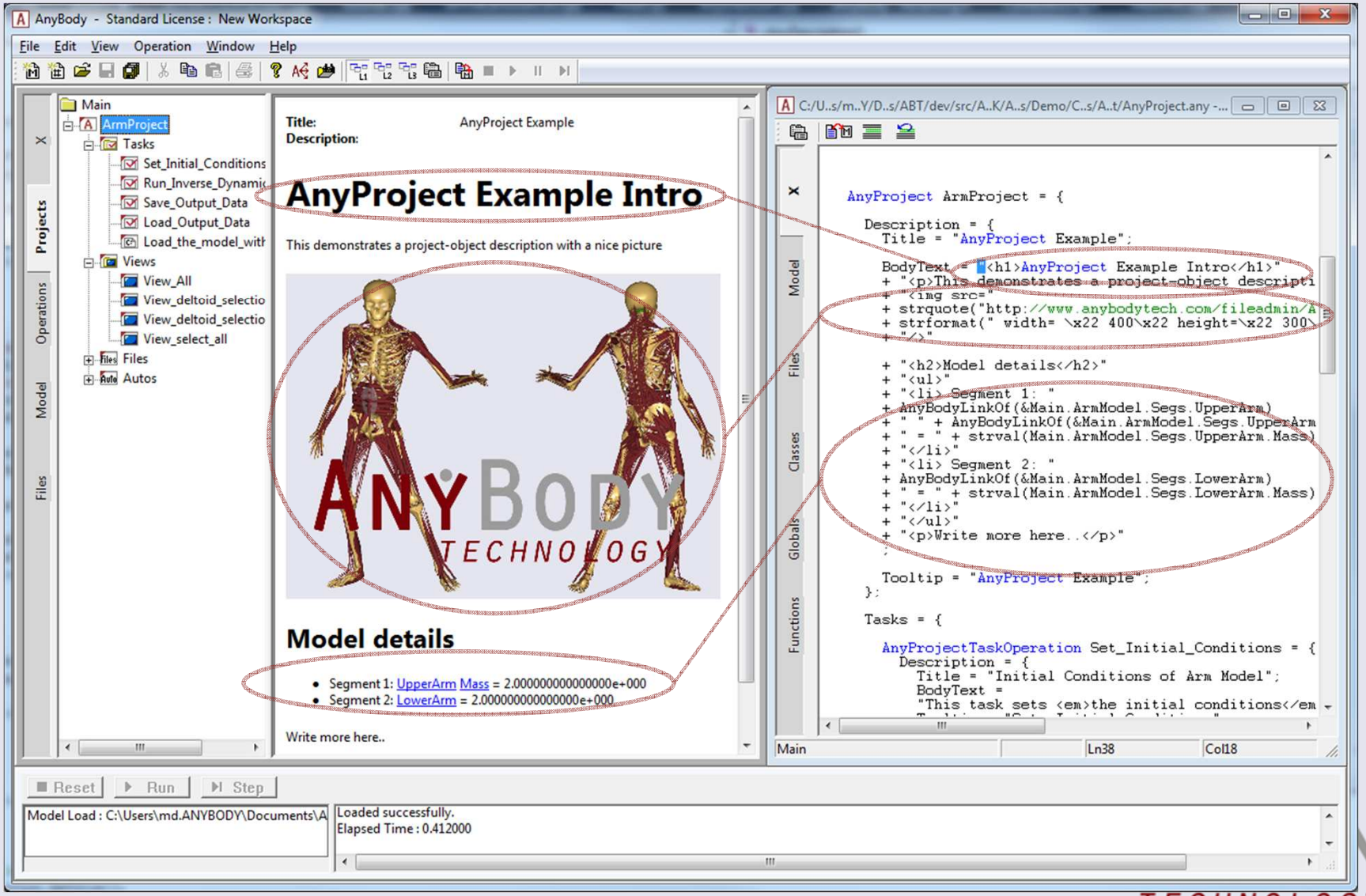
Functions

- HTML:
  - Formatted text
  - Images
  - External links
- Internal links
  - Object links to Model Tree
  - Code links to editor
  - Operation execution
    - AnyOperation objects
    - Class Operations

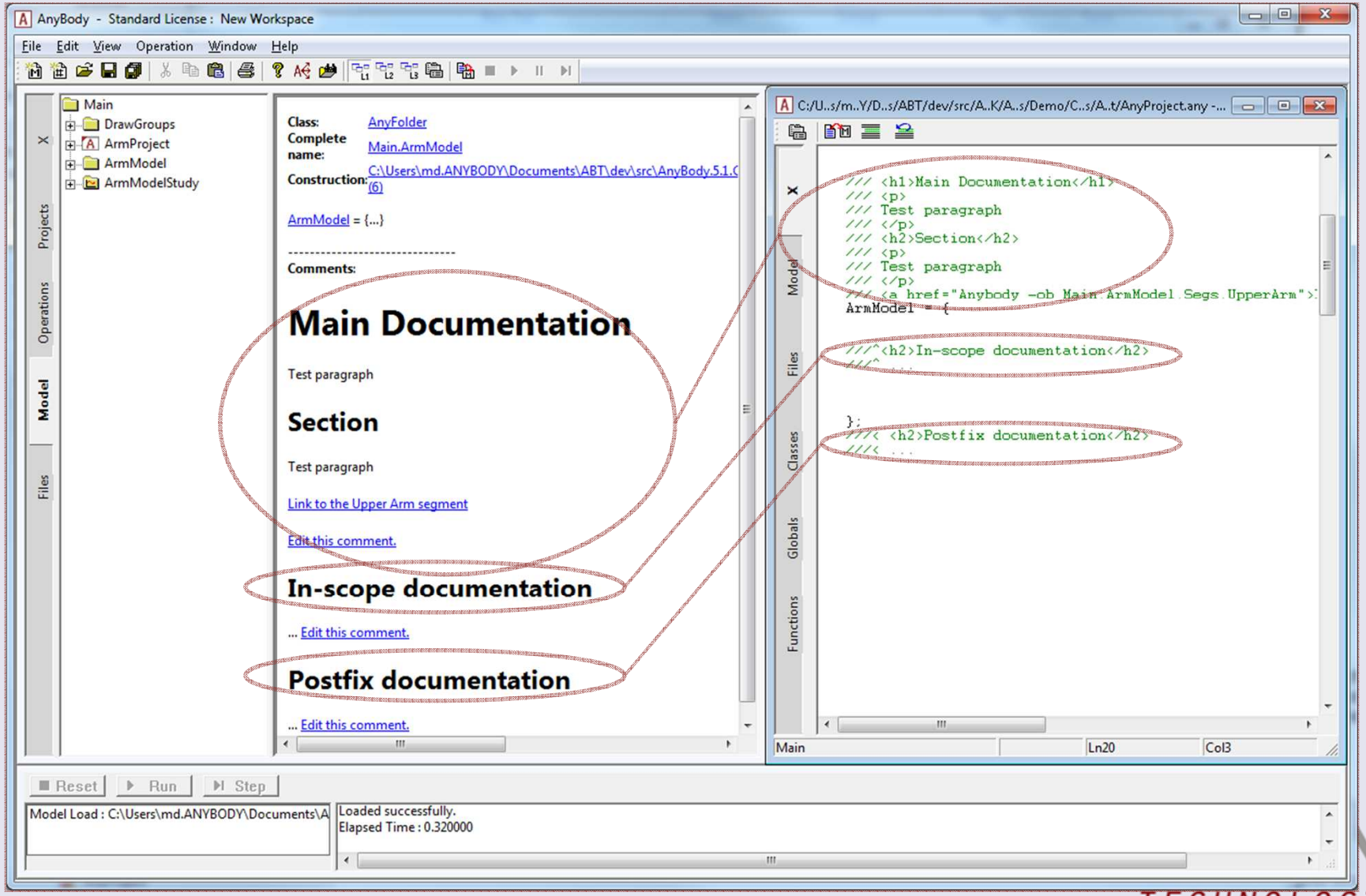


ANYBODY TECHNOLOGY

Projects (AnyProject)



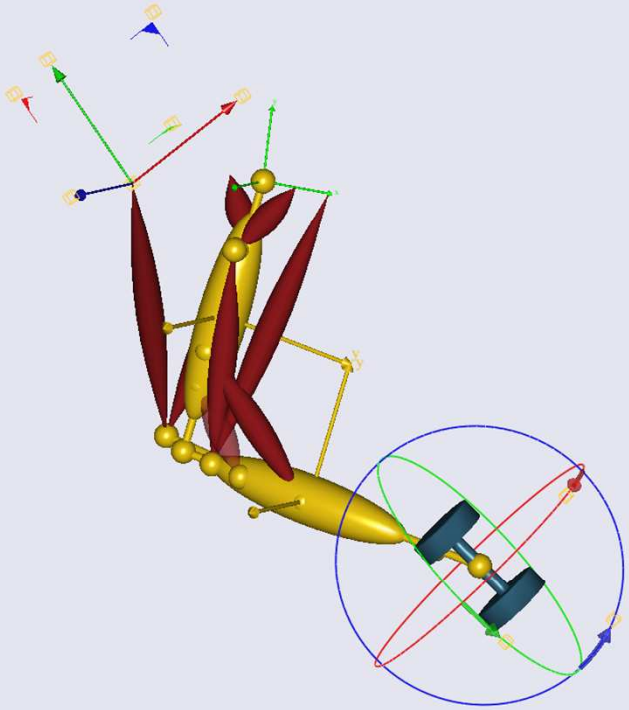
Documentation Comments



TECHNOLOGY

Drawing Widgets

- Functions
  - Objects in the model
    - Classes AnyDrawWidget*
  - Interaction via Model View
  - Sub-sequent operation
    - OnDrag
    - OnMouseRelease
    - More to come
- Widget movement types:
  - Translation
    - Vector
  - Rotation
    - Rotation Matrix
    - Single angle / Euler Angles (New!)
- News: Joint angle interactions



ANYBODY TECHNOLOGY

Models

- AMMR, v.1.4
- Other models

ANYBODY TECHNOLOGY

Models

- AnyBody Managed Model Repository (AMMR), v.1.4
  - C3DProject – updated for easier usage
    - AnyProject wrapped C3D data interface application
  - Various updated applications
- AMS demos and tutorials updated

- Knee Implant Model (Grand Challenge 2010)
  - A separate model, based on AMMR body models

Summary

- Introduction
- Features for advanced joint modeling
  - Surface contact model
  - Enhanced force-dependent kinematics solver
- New and updated interfaces
  - Hook for external code (C++ or Python)
  - Improved FEA interface for Ansys and Abaqus
  - Interface to Mimics, Materialise
- GUI features
  - User-defined documentation in HTML-based views in AMS GUI
  - Drawing Widgets
- Models

ANYBODY TECHNOLOGY

Resources

- Webcasts to come:
  - http://www.anybodytech.com/info.html?f=webcasts-live
  - 8 Sep: *Modeling and analysis of non-conforming joints in AnyBody I*, John Rasmussen
  - 29 Sep: *Modeling and analysis of non-conforming joints in AnyBody II*, John Rasmussen

- Previous webcasts:
  - http://www.anybodytech.com/info.html?f=webcasts-on-demand

ANYBODY TECHNOLOGY