

AnyBody Release Notes

(Version 4.0.0)

Contents

[System Requirements and Recommendations](#)

[Version: 4.0.0](#)

[User interface \(UI\)](#)

[AnyScript and Modeling](#)

[Bug fixes and other minor changes](#)

[Documentation, Demos, and Model Repository](#)

[Known Issues \(link to separate document\)](#)

[\(To the top...\)](#)

System Requirements and Recommendations

- The system runs only on standard Windows PC computers with Windows XP or Windows Vista operating systems.
- The system may work on earlier Windows operating systems, but this is not supported. In particular, Windows98 systems have been known to cause problems with graphics.
- The system will in principle run on very little memory, but if you have less than 128 Mb you will only be able to run very small models. Larger models typically require 256 Mb or more, and a full-body model does not run well on less than 512 Mb.
- The system should work on any processor capable of running the recommended operating systems, but the faster the better. Large models are computationally demanding.
- A graphics adapter with OpenGL support is preferable. In particular, graphics performance may be slow on shared-memory graphics adapters. At least 128Mb of dedicated memory on the graphics adapter is recommended for running large models such as full-body models from the model repository.
- 0.5Gb of free disk space primarily needed when working with full-body models.

[\(To the top...\)](#)

Version: 4.0.0

[\(To the top...\)](#)

User interface (UI)

- **Model View:** Model View has been equipped with a new render engine that is based on scene graph technology with a much closer relationship to the model. The highlights in the new Model View are:
 - Interaction with the model appearance in the Model View allowing simple manipulations without acting through drawing objects in the AnyScript code.
 - Available manipulations are object selection, show, hide, and transparency setting of objects
 - Model appearance manipulations are available in the Model View as well as on objects in the Model Tree Views.
 - Navigation from Model View to the model, i.e., the Model Tree Views, is now possible. This, together with model appearance manipulations, is very powerful extensions for browsing the model.
 - Not all features for the old Model View render engine have been restored in the new. Therefore, it is in version 4.0 possible to switch to the old render engine resulting in a Model View much similar to previous AnyBody versions. This is done in the properties of each Model View.
 - AnyChart drawing objects is not yet handled properly by the new render engine, but this will be changed in a version 4 update. AnyChart is being redesigned with new facilities.
 - The scene graph technology of the new render engine is a basic improvement that will allow powerful new extensions to Model View in future versions. Version 4.0.0 is just the first step towards a more powerful Model View. User feedback will be sincerely appreciated.
- **More links in the UI**
 - AnyScript class names are now linked to a new html-based version of the AnyScript Reference manual.
 - Object links, i.e., links between text with object names and the model (Model Tree View), have been enabled, greatly improving the model browsing facilities of AnyBody. Object links appear in error messages, warning, etc., and text based class operations windows, such as the Object Descriptions window, and more.
- **AnyScript Editor**
 - Minor updates of the editor facilities in particular for browsing the model.
- **Console application**
 - Class Operations can now, to some extent, be executed from the Console application.
 - Class Operations with user interaction are not fully support yet.

- **Operation control**
 - The AnyBody operation control has been redesigned to provide a more safe interaction between independent operations, such as model operations (AnyOperation) and user operation e.g. Class Operations, and connected operations, such as nested operations.
 - The operation control window is updated with better progress counters for the operations, in particular for sub-operations of nested operation structures.
 - Basic operation output has been redesigned and standardized according to the new operations control counter system.
 - The new operation structure in new design of the mechanical studies, see below, required these updates.

[\(To the top...\)](#)

AnyScript and Modeling

- **“AnyScript++”**: AnyScript has been equipped with facilities for making user-defined classes, or as they are called “Class Templates”. This new set of features has been called AnyScript++ since it is a step similar to the programming language C++, which was created as an extension to standard C with the facilities of using classes (Object-oriented programming). AnyScript++ is a smaller extension to previous versions of AnyScript than C++ was compared to C, but nevertheless this is the first step allowing AnyBody users to define their own classes for AnyScript. It is important to notice the following:
 - User-defined Class Templates can be defined inside the AnyScript files. A Class Template functions after proper declaration almost like normal built-in AnyScript classes, i.e., objects can be declared using the Class Template name and syntax similar declaring normal objects in the AnyScript code.
 - Class Templates are restricted to being created by objects of other existing classes. This implies that Class Templates cannot define new algorithmic functionality such as operations (member functions/methods). It can only contain other objects of existing classes.
 - Access control of the members in the Class Template allows some level of encapsulation data in the Class Template based objects. Members of the Class Template be open for initialization, either obligatory or optionally, like any normal AnyScript object. Otherwise, initialization is denied.
 - Template arguments allow conditional code in the Class Templates. Template arguments are parameters provided to the objects in the object declaration line. They are handled during the preprocessing stage of loading and therefore they can be used to control which part of the code that shall be active. They can be considered as parameters created by #define –statements outside the Class Template and supplied to it during the object declaration.

- **Mechanical Studies:** The mechanical studies, hereunder AnyMechStudy, AnyBodyStudy, AnyBodyCalibrationStudy have been redesigned to accommodate for new features in this and future versions. Notice the following:
 - Conceptual idea: The new concept is that all mechanical studies are basically alike and it is their predefined set of operations that make the primary difference the mechanical study classes. A new base-class, AnyMechStudyBase, has been introduced to support this concept. Significant parameters for a particular operation is now found as attributes to the operation object itself and therefore multiple objects of the same operation class can exist in the same study thereby allowing similar analyses but with the possibility for different properties. AnyMechStudyBase-derived classes have a predefined sets of operations but more operations can be added by the user if practical for a particular purpose.
 - Old functionality from previous versions of AnyBody is maintained in deprecated classes and members in order to supply some degree of backwards compatibility; but please notice that deprecated functionality will be removed in future versions. It is recommended to use this opportunity to remove old functionality from your models. All old functionality should also be available, either unchanged or in improved forms, within the new mechanical studies and operations.
 - AnyKinStudy is a new mechanical study class aimed exclusively at kinematic analysis. It is also introduced as a base-class for other studies that also needs kinematic analysis capabilities.
 - AnyBodyCalibrationStudy has been updated. It is now derived from AnyKinStudy, and therefore it has lost its superfluous capability for doing inverse dynamic analysis, which was inherited from its former base-class AnyBodyStudy.
 - AnyMechStudy has been updated. Old functionality is available in the AnyMechStudy1, but notice that AnyMechStudy1 is deprecated and will be removed.
 - AnyBodyStudy has been significantly updated and redesigned (see further comments below). Old functionality is available in the AnyBodyStudy1, but notice that AnyBodyStudy1 is deprecated and will be removed.
 - Mechanical operation classes used in the redesigned mechanical studies are all new operations classes. They are all ordered in a special branch of the AnyOperation class tree. Most mechanical operations are now equipped with a pre- and a post-processes. These processes are basically operation sequence classes, similar AnyOperationSequence, which can be filled by the user. Some mechanical studies, for instance AnyKinStudy and AnyBodyStudy, have predefined content in the pre-process. For instance, the kinematic analysis and inverse dynamic analysis of AnyMechStudy and AnyBodyStudy have predefined pre-processes containing the initial condition operation of the same study. Functionally, this makes the system behave almost like previous versions, except that the possibilities for using different settings for different operations and for

their pre- and post-processes are significantly improved. This will be highly valuable due to the many new features that are and will be added to these operations.

- **Muscle Recruitment/Inverse Dynamics in AnyBodyStudy**
 - The inverse dynamics solver in the new AnyBodyStudy class has been equipped with new optimization solvers that provide a much more robust and versatile foundation for the muscle recruitment analysis. Previous versions of AnyBody has been known to have weaknesses for very large musculoskeletal models. These problems are minimized with the new solver engine.
 - With the new solver engine, a number of new and modified muscle recruitment criteria have been introduced.
 - The Muscle Recruitment Criteria options are:
 - **Min Max Strict:** This is basically the Min Max criterion with the “iterative” option known from previous AnyBody versions. A new algorithm has been designed that makes the solution more robust even for full body models as the ones in the public model repository. The new algorithm has also eliminated various tweaking needs that were occasionally used in the larger models before.
 - **Quadratic:** The sum of muscle activities squared; a well-known criterion in the biomechanical research community.
 - **Polynomial:** The sum of muscle activities raised to some user defined power.
 - **Linear:** The sum of muscle activities. This criterion is known to be non-physiological and it is added primarily because (1) it is an obvious case for any new-comer to the field, (2) it is therefore useful for educational purposes, and (3) it is straight forward to obtain and an obvious candidate in combination with other criteria. It is not recommended to use this criterion independently for any kind of real musculoskeletal analysis.
 - **Special criteria allow auxiliary terms to be added to the primary term of the muscle recruitment criteria function.**
 - “Min Max Auxiliary” is a non-iterative min-max solver that uses a linear and a quadratic term to resolve the inherent indeterminacy that lies in non-iterative min-max.
 - Quadratic with an auxiliary linear term is a combination of the quadratic and linear criteria.
 - Auxiliary terms, when appropriate, can be weighted according to the primary terms by user-defined weight factors.
 - **Upper bounds have been introduced (optionally) for all criteria.** For the strict Min Max criterion upper bounds do not influence the muscle load distribution, but for other criteria upper bounds may be essential to the muscle load distribution, at least for higher loading cases.

- The criteria functions are typically based on muscle activity defined as muscle force divided by current muscle strength. The new muscle recruitment setup opens for some other definitions for the basic measure in the criteria functions, but these options are still limited. These settings can be done for primary and auxiliary terms independently.
- **New input files:** Three new data input file formats are supported. They were introduced as beta functionality in version 3.1, but there are now out of their beta testing phase.
 - AnyInputC3D: A C3D importer that can read in C3D data file and create a kinematic model of the markers, which can be used for driving a body model. The model driving the markers by the c3d data can be exported as AnyScript code for further modification if needed. Basic data filtering functionality is included in the class.
 - AnyInputBVH: A BVH file importer that can read in BVH data file and create a kinematic model of the hierarchical kinematic data. Basic data filtering functionality is included.
 - AnyInputDB: A database connection module that allows connection and import of data from standard databases such as MS Access, MS SQL Server, MySql, OracleXE. Others databases may follow in future versions, depending on demand.

[\(To the top...\)](#)

Bug fixes and other minor changes

- AnyBody.exe and AnyBodyCon.exe are no longer independent exe files that can be moved freely around on your system. They do now depend on external library DLL files and they must reside in the installation directory where required library files are located properly.

[\(To the top...\)](#)

Documentation, Demos, and Model Repository

- Version 1.0 of the AnyBody Managed Model Repository (AMMR) has been released. This support and uses facilities of AnyBody version 4.0. It is recommended to use this new repository as basis for your modeling.
- The demo models installed together with AnyBody has been updated with models and modifications from the AMMR.
- The AnyScript reference manual has been updated significantly. In addition, to the PDF version of the manual, a new setup in CHM format (HTML based Windows help file) has been included. This version of the reference manual is used as target for the links from the AnyBody GUI application.

- Sections about AnyScript++ have been added to the introduction chapter of the AnyScript Reference Manual: More elaborate documentation will come with future versions.
- The tutorials have also been updated:
 - Now they are enclosed in installation package in CHM format.
 - Tutorials about mechanical studies have been updated significantly. A new tutorial about musculoskeletal modeling and muscle recruitment has been added. It is called Inverse Dynamics of Muscle System.
 - The Getting Started and the Block Building Tutorials have been updated according to the new repository structure and new facilities for structuring models in the software. AnyScript++ (Class Templates) is, however, not yet used in the public repositories.
 - Not all tutorials have been completely updated to the new version

[\(To the top...\)](#)
