

The GNU Enterprise Application Server

Application Programmer's Interface
Edition 0.0.11, 2004-02-24

Reinhard Müller

Copyright © 2002 Free Software Foundation

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts.

A copy of the license is included in the section entitled “GNU Free Documentation License”.

Table of Contents

1	Data Types	1
2	API Functions	2
2.1	Session Management	2
2.2	Handling Lists Of Objects	2
2.3	Handling Specific Objects	3
3	System Classes	5
3.1	gnue_module	5
3.2	gnue_class	5
3.3	gnue_property	5
3.4	gnue_procedure	6
3.5	gnue_useraccess	6
Appendix A GNU Free Documentation License		
	7
A.1	ADDENDUM: How to use this License for your documents	
	12
Appendix B Data Type and Function Index		
	13

1 Data Types

In the API definition, we will make use of the following data type placeholders, which will have to be translated into appropriate data types for the various implemented RPC mechanisms:

void Data Type
This is used as the result type for functions that actually don't return a result.

boolean Data Type
This is a boolean data type that can only hold TRUE or FALSE values.

integer Data Type
This is an integer data type which must be able to contain signed 32 bit values.
This restriction limits the maximum list size to more than 2 billion objects.

string Data Type
This is a data type that must be able to hold variable length strings without length limitations.

stringlist Data Type
This is an one-dimensional array of elements of the type **string**. Implementation may restrict the number of elements to 32767.

stringtable Data Type
This is a two-dimensional array of elements of the type **string**. Implementation may restrict the number of columns as well as the number of rows to 32767.

session_id Data Type
This is a 32 bit integer value identifying a session. There is no special meaning one could read from the value.

list_id Data Type
This is a 32 bit integer value identifying a list. There is no special meaning one could read from the value.

2 API Functions

All API functions can raise exceptions on failure. All functions are atomic, in the sense that in case of an exception the function has no effect at all. No error in a function call can cause a function to be "half-done".

The exact syntax of the API functions is dependant on the selected RPC interface and the language sitting on top of it. However, we are describing the functions in a C-like syntax here, using the data type placeholders we defined above.

2.1 Session Management

- session_id open** (*auth_parameters*) Function
 Opens a connection to Appserver using the given parameters for authentication. The number and type of the parameters still have to be decided. The return value is a handle to the session that has to be passed to all subsequent function calls. An Exception is raised if the server cannot be contacted or authentication fails.
- void close** (*session_id session*, *boolean commit*) Function
 Closes the connection identified by *session*. If *commit* is TRUE, an uncommitted transaction of this session is implicitly committed; if *commit* is FALSE, it is implicitly rolled back. After this call, *session* will no longer be a valid session identifier. An Exception is raised if *session* is not a valid session identifier.
- void commit** (*session_id session*) Function
 Commits the current transaction of *session*, making all changes permanent. An Exception is raised if *session* is not a valid session identifier.
- void rollback** (*session_id session*) Function
 Discards all changes done in *session* since the last **commit** or **rollback**. An Exception is raised if *session* is not a valid session identifier.

2.2 Handling Lists Of Objects

These functions provide a means for getting data for a list of objects fulfilling certain conditions.

- list_id request** (*session_id session*, *string classname*, *string conditions*, *stringlist sortorder*, *stringlist properties*) Function
 Requests a list of objects of class *classname* matching the *conditions*. Appserver prepares to send the values of the properties listed in *properties* on subsequent calls to **fetch**, where the order of the objects is determined by the properties listed in *sort_order*. The properties in *sort_order* may, but need not appear in *properties*. *classname*, *conditions*, *sort_order*, and *properties* must contain fully referenced identifiers for classes or properties.
 This function only returns a list identifier. No actual data is passed over the network when calling this function.
 An Exception is raised if *session* is not a valid session identifier, the requested class does not exist, the current user has not access to the requested class, or any of the given *properties* does not exist.
- integer count** (*session_id session*, *list_id list*) Function
 Returns the number of objects contained in *list*, where *list* is the return value of a previous call to **request**.
 An Exception is raised if *session* is not a valid session identifier or *list* is not a valid list identifier.

stringtable fetch (*session_id session, list_id list, integer start, integer count, boolean close*) Function

Returns a 2-dimensional array of data with *count* rows, where column 0 always holds the *object_id* of the object, and the remaining columns contain the values for the properties defined in the previous call to **request**. Negative values for *start* indicate position from the end of the list. Negative values for *count* are invalid. Count may not be greater than 32767.

If *close* is TRUE, then the list is "closed", all server-side memory allocated with the management of this list is freed and the *id list* may no longer be used in calls to **count** or **fetch**. However, *object_ids* and other data already received remains valid.

An Exception is raised if *session* is not a valid session identifier or *list* is not a valid list identifier.

2.3 Handling Specific Objects

These functions provide a means for reading, writing and deleting an object or a set of objects, as well as for calling a procedure for an object or a set of objects. However, the *object_ids* for the objects to operate upon have to be determined before these functions can be used, for example by using the list handling functions described above.

stringtable load (*session_id session, string classname, stringlist object_ids, stringlist properties*) Function

Returns a 2-dimensional array of data with a row for every entry in the *object_ids* list and a column for every entry in *properties*. Unlike **fetch**, this function does *not* automatically return the *object_ids* in column 0.

If this function is called with one of the *object_ids* being an empty string, the corresponding row in the result contains a list of the data types of the *properties*. This is a temporary hack and will be removed again in a future version!

An Exception is raised if *session* is not a valid session identifier, the requested class does not exist, the current user has not access to the requested class, any of the given *object_ids* does not exist, or any of the given *properties* does not exist.

stringlist store (*session_id session, string classname, stringlist object_ids, stringlist properties, stringtable values*) Function

Stores the data in *values* in the objects identified by *object_ids*. Every row in *values* matches an entry in *object_ids*, while every column matches an entry in *properties*. Empty object ids indicate that new objects with that data should be created. Validation is performed before the actual storing is done. If validation of a single object fails, none of the objects are stored, but an exception is raised. This function returns a list of all object ids of the stored objects. This is important for the caller to know under which object ids the new objects have been stored and can be accessed from now on. Note that after calling **store**, **commit** has to be called to make the changes persistent, while a call to **rollback** can undo the changes.

An Exception is raised if *session* is not a valid session identifier, the requested class does not exist, the current user has not access to the requested class, any of the given *object_ids* does not exist, any of the given *properties* does not exist, or any of the *values* does not fit the corresponding property.

void delete (*session_id session, string classname, stringlist object_ids*) Function

Deletes the objects of class *classname* identified through *object_ids*.

An Exception is raised if *session* is not a valid session identifier, the requested class does not exist, the current user has not access to the requested class, or any of the given *object_ids* does not exist.

stringlist call (*session_id session*, *string classname*, *stringlist object_ids*, *string procedurename*, *stringlist parameters*) Function

Calls the procedure *procedurename* for every object identified through the *object_ids* and passes the same *parameters* to every call. The number of entries in *parameters* must match the parameter count of the procedure. This function returns a list of strings that contains the results of the procedure calls for each object.

An Exception is raised if *session* is not a valid session identifier, the requested class does not exist, the current user has not access to the requested class, any of the given *object_ids* does not exist, or the given procedure is not defined.

3 System Classes

The following classes are always defined and can be accessed to query and/or influence class definitions and (in future) other information about the state of the Application Server.

3.1 gnue_module

gnue_name string(35) Property of gnue_module
The name of the module.

gnue_comment string(70) Property of gnue_module
Arbitrary text explaining the purpose of the module.

3.2 gnue_class

gnue_module gnue_module Property of gnue_class
The module that originally defines the class.

gnue_name string(35) Property of gnue_class
The name of the class without the module name. You can find out the module name by referencing the **gnue_module** property.

gnue_comment string(70) Property of gnue_class
Arbitrary text explaining the purpose of the class.

3.3 gnue_property

gnue_class gnue_class Property of gnue_property
The class the property belongs to.

gnue_module gnue_module Property of gnue_property
The module that defines the property.

gnue_name string(35) Property of gnue_property
The name of the property without the module name. You can find out the module name by referencing the **gnue_module** property.

gnue_type string(35) Property of gnue_property
The type of the property. This can be one of the predefined types "string", "number", "boolean", "date", "time", or "datetime", or the name of a class, in which case the property is a reference property to that class.

gnue_length number(6) Property of gnue_property
The length of the property. Only relevant if **gnue_type** is "string" or "number".

gnue_scale number(4) Property of gnue_property
Only relevant if **gnue_type** is "number", in which case it defines the number of fractional digits, while **gnue_length** defines the total number of digits.

gnue_comment string(70) Property of gnue_property
Arbitrary text explaining the purpose of the property.

3.4 gnue_procedure

gnue_class gnue_class	Property of gnue_procedure
The class the property belongs to.	
gnue_module gnue_module	Property of gnue_procedure
The module that defines the property.	
gnue_name string(35)	Property of gnue_procedure
The name of the property without the module name. You can find out the module name by referencing the gnue_module property.	
gnue_language string(10)	Property of gnue_procedure
Language of the procedure. Currently only "python" is valid.	
gnue_code string	Property of gnue_procedure
The source code of the procedure.	
gnue_comment string(70)	Property of gnue_property
Arbitrary text explaining the purpose of the procedure.	

3.5 gnue_useraccess

Note: This class is only preliminary and will most probably disappear in future and be replaced by a more flexible access control system.

gnue_username string(20)	Property of gnue_useraccess
gnue_password string(20)	Property of gnue_useraccess
gnue_classlist string(100)	Property of gnue_useraccess
A list of the fully qualified names of all classes the user should have access to, seperated by blanks.	

Appendix A GNU Free Documentation License

Version 1.1, March 2000

Copyright © 2000 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgments" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgments and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover

Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled “History” in the various original documents, forming one section entitled “History”; likewise combine any sections entitled “Acknowledgments”, and any sections entitled “Dedications”. You must delete all sections entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an “aggregate”, and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document’s Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that

you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.1 ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) *year* *your name*.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being *list their titles*, with the Front-Cover Texts being *list*, and with the Back-Cover Texts being *list*. A copy of the license is included in the section entitled ‘‘GNU Free Documentation License’’.

If you have no Invariant Sections, write ‘‘with no Invariant Sections’’ instead of saying which ones are invariant. If you have no Front-Cover Texts, write ‘‘no Front-Cover Texts’’ instead of ‘‘Front-Cover Texts being *list*’’; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Appendix B Data Type and Function Index

B

boolean..... 1

C

call..... 4
close..... 2
commit..... 2
count..... 2

D

delete..... 3

F

fetch..... 3

I

integer..... 1

L

list_id..... 1
load..... 3

O

open..... 2

R

request..... 2
rollback..... 2

S

session_id..... 1
store..... 3
string..... 1
stringlist..... 1
stringtable..... 1

V

void..... 1