# Table of contents

1 Foreword .................................................................................................................................................... 5  
  1.1 Notes on the documentation ................................................................................................................ 5  
  1.2 Safety instructions ................................................................................................................................... 6  
2 Concept ........................................................................................................................................................... 7  
3 Workflow .......................................................................................................................................................... 9  
  3.1 Creating a new project ............................................................................................................................ 9  
  3.2 Using an existing project ......................................................................................................................... 9  
  3.3 Working with set functionality ............................................................................................................... 10  
4 Reference, user interface ............................................................................................................................. 14  
  4.1 Settings tab .......................................................................................................................................... 14  
  4.2 History tab ........................................................................................................................................... 15  
  4.3 Diff tab .................................................................................................................................................. 16  
  4.4 Status tab ............................................................................................................................................ 17  
  4.5 Conflict tab ......................................................................................................................................... 18  
  4.6 Manual tab ........................................................................................................................................... 20  
5 Reference, project settings ........................................................................................................................... 21  
6 Reference, server settings ............................................................................................................................ 22  
7 FAQ ............................................................................................................................................................. 23  
8 3rd Party Licenses ......................................................................................................................................... 24
1 Foreword

1.1 Notes on the documentation

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with applicable national standards.

It is essential that the documentation and the following notes and explanations are followed when installing and commissioning the components.

It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH.

Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents:


with corresponding applications or registrations in various other countries.

EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.
1.2 Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of symbols

In this documentation the following symbols are used with an accompanying safety instruction or note. The safety instructions must be read carefully and followed without fail!

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚠️ DANGER</td>
<td>Serious risk of injury! Failure to follow the safety instructions associated with this symbol directly endangers the life and health of persons.</td>
</tr>
<tr>
<td>⚠️ WARNING</td>
<td>Risk of injury! Failure to follow the safety instructions associated with this symbol endangers the life and health of persons.</td>
</tr>
<tr>
<td>⚠️ CAUTION</td>
<td>Personal injuries! Failure to follow the safety instructions associated with this symbol can lead to injuries to persons.</td>
</tr>
<tr>
<td>📌 NOTE</td>
<td>Damage to the environment or devices Failure to follow the instructions associated with this symbol can lead to damage to the environment or equipment.</td>
</tr>
</tbody>
</table>

Tip or pointer

This symbol indicates information that contributes to better understanding.
More and more machine and system functions are implemented in software. To allow for this, nowadays teams of several programmers are involved in creating the control code and subsequently the commissioning of these machines and systems. This poses several challenges, especially during commissioning:

- In many cases, access to source control systems is not guaranteed
- The latest (active) software version on the machine may differ from the local version
- If changes that are "downloaded" to the target system turn out to be incorrect, it is not possible to undo the "download".

TwinCAT Multiuser was developed to meet these challenges. This is a "local" Source Control System on the target system, the handling of which has been fully integrated into the existing workflow. This means that no in-depth knowledge of the exact functionality of Source Control Systems is required to use TwinCAT Multiuser.

TwinCAT Multiuser is available from version TC3.1.4024. In this version, TwinCAT Multiuser mainly facilitates collaboration of several programmers, with each programmer working on one PLC project. If several PLC projects are integrated in a TwinCAT project, the multiuser functionality can be enabled separately for each project. In this case, separate repositories are automatically created for each of these projects, both locally and on the target system. The following diagram illustrates the concept.

Fig. 1:

The target system is selected as the Source Control Server during commissioning, since a connection to the target system is required in any case in order to load a control program onto the target system. This avoids the need for additional infrastructure.

Another objective of multiuser integration is to avoid the need for specific Source Control knowledge. The Source Control functionality is integrated in the standard workflow. For example, the current project status is automatically transferred to the target system during a download or online change of a project, without the need to trigger the transfer separately. The history is also created automatically. All changes (including user name, time stamp and change) are recorded. If required, a query for a comment can be set for each change transfer.
To ensure that the history created in this way continues to be available after commissioning, a Source Control System was integrated, which permits several repositories and also transfers the full history into all repositories. After successful commissioning, it is thus possible to transfer the active project status of the machine/system to a "development repository", including traceability of all steps performed during commissioning. The latter is done with the on-board resources of the Source Control System.

The Source Control System on which the multiuser functionality is based is Git.

**Differentiation from the Source Control Integration of TwinCAT:**

The general interfacing of TwinCAT with Source Control Systems remains unaffected by the multiuser functionality and can take place independently of it. By using the standard interface of the Microsoft Visual Studio Shell integrated in TwinCAT, a large number of Source Control Systems are available for this purpose. In addition, all aspects of a TwinCAT project can be handled by this integration.

The TwinCAT multiuser functionality is based on the Git standard. As already described, the aim of this functionality is to enable a team of several programmers to collaborate on a control program during the commissioning phase, without the need for a special infrastructure or special knowledge of Source Control Systems on the part of the programmers.

Although it is therefore possible to use different Source Control Systems for both functionalities, the full capabilities can only be achieved if both functionalities are based on Git. This refers to the transfer of the history during commissioning to the "main" Source Control System.
3 Workflow

3.1 Creating a new project

To activate this function for a project that is not yet managed through the multiuser functionality, carry out the following steps.

1. Activate the Use Multiuser option in the project settings of the PLC project that is to be managed through the multiuser functionality (see Reference, project settings [21]).
2. Open the Multiuser Explorer window under View -> More Windows -> Multiuser Explorer.
3. Select a PLC project in the combo box of the Multiuser Explorer.
   ☰ Several tabs should now appear in the Multiuser Explorer.
4. Select the Settings tab.
5. In the ADS Route field, select a target system for the multiuser repository.
6. In the Multiuser Repository field, select a name for the multiuser repository.
7. If required, select the option Ask for update message on each usage (see also Settings tab [14]).
8. Click the Init local and remote button.
   ☰ The multiuser repositories in the project and on the target system are initialized.

3.2 Using an existing project

If an existing project is to be connected to the multiuser server for which a repository has already been set up on the target system, proceed in the same way as for setting up the functionality. The only difference is that in the combo box for the name of the target system you do not create a new repository but select an existing one.

The steps in detail:

1. Open the project.
2. In the project settings of the PLC project that is to be managed through the multiuser functionality, check whether the option Use Multiuser (see Reference, project settings [21]) is enabled. If not, enable it.
4. Select the PLC project in the combo box of the Multiuser Explorer.
   ☰ Several tabs should now appear in the Multiuser Explorer.
5. Select the Settings tab.
6. In the ADS Route field, select the target system that contains the multiuser repository.
7. In the Multiuser Repository combo box, select the name of the multiuser repository to which the project is to be linked.
8. If required, select the option Ask for update message on each usage (see also Settings tab [14]).
9. Click the Init local and remote button.
   ☰ The multiuser repository in the project is now initialized.

If you do not have an existing project, you can create a TwinCAT project. Then add an empty PLC project and link it to the multiuser repository on the target system as described above.

The Git Source Control System uses absolute paths in the repositories. This means that copying a project in which multiuser functionality is already installed (including the .TcGit folder) will result in the multiuser functionality not working correctly!
## 3.3 Working with set functionality

If the multiuser functionality was set up as described under Creating a new project or Using an existing project and linked to the PLC project, this functionality is automatically integrated into the existing workflow.

### Transferring the data to the target repository

The changes generated while working on the PLC project are automatically transferred to the target repository (if multiuser functionality is activated) during a download or online change.

Checks are carried out to ascertain whether there is a conflict with the data in the target repository. This is the case if other programmers working on the project have already made changes to the same project parts (e.g. POUs, settings etc.) and transferred them to the target repository. If there is a conflict, it is displayed in the following dialog.

![Update Remote](image)

In this selection dialog you then have the option to
- resolve the conflicts using the TwinCAT Project Compare Tool,
- force an update of the target system including the target repository, or
- cancel the login.

**Force Update** can be used if the target system is required to run with the current project without taking into account the changes made by the other programmers involved in the project.

The normal workflow for collaborating on a project involves resolving conflicts. In this step, you can merge the changes you have made to the project with those of the other programmers working on the project.

1. To do this, click **Solve Conflict**.
   - In the dialog that follows you will be informed that resolving conflicts may cause changes to the files, in which case the project has to be recompiled after a merge operation.
2. Confirm this message with **OK**.

![Microsoft Visual Studio](image)

- The **Conflict** tab now opens. This allows you to discard all your local changes and apply the version from the target repository, or to merge the two versions.
3. To merge, click **Merge Local and Remote**.

4. For each conflict displayed, click **Start Merge**.
   - The TwinCAT Project Compare Tool opens, in which you can merge the changes.
5. Accept the result with **Accept**.
Once all changes have been merged, you are notified that there are no further conflicts.

6. Now log in again.

The current and merged version of the project is transferred to the target system and stored in the target repository.
Only one at a time

Although several programmers can be logged into the target at the same time to view the project status, only one programmer can make changes at any one time. All other programmers have to log out briefly. The logged-in programmers are shown in the following dialog.

Once the changes have been made, the other programmers can log in again and the workflow shown above is triggered to merge the changes with those of the other programmers. If the other programmers have not yet made any further changes to the project, they are notified that a newer version is available on the remote system.
4  Reference, user interface

The Multiuser Explorer is the central administration tool for the multiuser functionality. This window remains empty as long as no project has been associated with multiuser functionality.

As soon as a project has been selected, the tabs listed below become visible.

4.1  Settings tab

Once a PLC project has been selected, the Settings tab appears as follows.
| **ADS route** | Combo box for selecting an existing ADS route |
| **Multiuser repository** | Combo box for selecting an existing MU repository or creating a new one |
| **Remote Multiuser URL** | URL for repository |
| **User name** | User name to be used for the entries in the history. The default setting is the placeholder "%LoggedInUser%". This is automatically replaced by the locally logged in user as user name in the comment. |
| **Ask for update message on each usage** | Option to query change messages |
| **Update message** | Structure of the update message. The placeholders "%Action %, %FilesChanged%" and "%UserMessage%" are automatically replaced by the corresponding information during an online change or download. If a placeholder is removed, the corresponding information is removed from the automatically generated comment. Placeholder: "%Action%": indicates whether an online change or download was performed. "%FilesChanges%": indicates the number and reason for the changed files (e.g. 2 added). "%UserMessage%": is replaced by the user-specific comment if the option Ask for update message on each usage is enabled. |
| **Init local and remote** | Initializes the local and target repositories |

### 4.2 History tab

The History tab has a Show History button. This opens a tool window which shows the history of the current project.

Example: History of the documentation example. In this example, changes were made both on the target system and on the local system.
4.3 Diff tab

The Diff tab shows all files where changes have been made that have not yet been transferred to the target repository.

As an example, the following image shows changes made to the PLC project file of the project Untitled1.
### 4.4 Status tab

Displays the status of the multiuser functionality and provides the option to disable this setting. This will delete the local repository.

#### Environment

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ok</td>
<td>Displays the state of the multiuser server:</td>
</tr>
<tr>
<td></td>
<td><strong>OK</strong>: local and target repository found</td>
</tr>
<tr>
<td></td>
<td><strong>LocalNotExists</strong>: no local repository was found</td>
</tr>
<tr>
<td></td>
<td><strong>RemoteNotExists</strong>: no target repository was found</td>
</tr>
</tbody>
</table>

#### Files

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ok</td>
<td>Shows the status of the files.</td>
</tr>
<tr>
<td></td>
<td><strong>OK</strong>: local and target repository found, no newer project status on target repository</td>
</tr>
<tr>
<td></td>
<td><strong>LocalNotExists</strong>: no local repository was found</td>
</tr>
<tr>
<td></td>
<td><strong>RemoteNotExists</strong>: no target repository was found</td>
</tr>
<tr>
<td>LokalsbehindRemote</td>
<td>A newer version exists on the target repository</td>
</tr>
</tbody>
</table>

#### Delete Multiuser files / Deactivate Multiuser

Deletes the local repository and disables multiuser functionality

---

The Files status entry also displays **OK** if the local files status is newer than that in the repository.

You can see which files have been changed in the [Diff tab](#).
4.5 Conflict tab

The Conflict tab shows whether there are any conflicts between the local status and the status in the target repository, or if a merge has been triggered. The following cases are possible:

Local and target repositories have the same status:

![Image of the Conflict tab showing no conflicts.]

The local system contains changes that have not yet been transferred:

![Image of the Conflict tab showing the option to discard local changes.]

Changes were made on both the local and the remote system:
Click the **Merge Local and Remote** button to show a list of the changes. In the following view you will see all objects in which changes have been made on both sides (for example the POU Main in the following figure).

To merge the changes, click the **Start Merge** button. The TwinCAT Project Compare Tool opens, in which you can merge the objects. After a successful merge, please confirm the changes with **Accept**, both in the TwinCAT Project Compare tool and in the Multiuser Explorer.

The target repository contains changes that have not yet been implemented locally:
Click the **Get Latest** button to fetch the current status of the target repository. TwinCAT now informs you that files have changed in the background and asks whether you want to reload them. Confirm this question with **OK**.

### 4.6 Manual tab

The **Manual** tab is used to write the current project status to the target system or to fetch it from there.

<table>
<thead>
<tr>
<th>Pull (override local)</th>
<th>Fetches the current project status from the target system and overwrites the local project. No merging of the project statuses takes place.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Push (override remote)</td>
<td>Overwrites the status on the target system. No merging of the project statuses takes place.</td>
</tr>
</tbody>
</table>
5 Reference, project settings

To use Multiuser in a project, this function has to be enabled.

| Use Multiuser | Enables the multiuser functionality of the PLC project |
6 Reference, server settings

The TwinCAT Multiuser functionality is based on the Git Source Control System. This is automatically installed with the "Minimal Git for Windows" version.

Changing the storage location of the Git repositories

If necessary, you can change the default storage location of the Git repositories on the target system. The default location is "C:\ProgramData\Beckhoff\MultiuserRepository".

To adjust the path, change the entry in the file "<TwinCAT Folder>/Functions/Multiuser/directorypath.config" so that it points to the desired path.
7 FAQ

The Multiuser functionality is behaving in a way that is not described. What can I do?

The status of the Multiuser functionality can be viewed in the MultiuserExplorer on the Status tab. This is the first place to look if there is any unexpected behavior when using the Multiuser functionality (see Status tab [p. 17]).

Also check whether the TwinCAT3 AdsGitServer Windows service has started on the target system and restart it if necessary.

How can I restart with the Multiuser functionality?

Local Git repositories can be deleted on the Status tab in the MultiuserExplorer with the DeleteMultiuser files/Deactive Multiuser button in order to restart with the Multiuser functionality (see Status tab [p. 17]).

I get the message: "RepositoryExistsException". How can I resolve this situation?

On account of an invalid operation in the past, a Git repository has already been created for the current project.

Delete this invalid repository on the Status tab in the MultiuserExplorer using the DeleteMultiuser files/Deactive Multiuser button.

I get the message "ADS error 0x745: Timeout elapsed". How can I resolve this situation?

Open the Status tab in the MultiuserExplorer. The message "RemoteNotExists" indicates that the remote repository has not been created or has been deleted.

Alternatively, look on the target system in the Windows Explorer as well under C:\ProgramData\Beckhoff \MultiuserRepository.

If you get this message several times, restart the TwinCAT3 AdsGitServer Windows service on the target system.

I get the message "ADS error 0x1: an internal error has occurred". How can I resolve this situation?

Please check whether the Multiuser functionality has been properly installed. Among other things, the files LibGit2Sharp.dll, TcAdsGitPackage.dll and TcAdsGitServer.exe as well as the folders cmd and mingw32 must be present in the C:\TwinCAT\Functions\Multiuser folder.
8 3rd Party Licenses

The TwinCAT Multiuser functionality uses Git as source control system, which is released under GNU General Public License version 2.0.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

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

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you". Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program 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.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public 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.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms. To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.