

DDE Client:

Dynamic Data Exchange for PLT Scheme

R. Matovinovic
robert.matovinovic@web.de

October 31, 2006

Abstract

This is the documentation for ddeclient a package providing DDE connectivity to PLT-Scheme.

Contents

1	Introduction	2
2	Realization	2
3	Features	2
4	Requirements	3
5	Files and Directories	3
6	Getting Started	3
7	Structure of a DDE process	4
8	Procedures	4
8.1	List of procedure arguments	4
8.2	List of procedures	5
9	Supported Clipboard Formats	6
10	Thanks	7
11	Glossary	8
12	License	9

Copyright ©2006 Robert Matovinovic
Permission is granted to copy, distribute and/or modify this document and related files under the terms of the GNU Lesser General Public License, Version 2.1 published by the Free Software Foundation. A copy of the license is included in this document section “License”.

1 Introduction

It enables DDE connectivity from within PLT-Scheme to other applications to control them and/or exchange data. Although DDE is a very old and not state of the art protocol there is still software around which can only communicate with other software via DDE. The use of such software was the reason for developing this package.

This documentation does not introduce DDE but the use of the provided procedures. For using DDE with a certain program look at this programs documentation. Because commands and examples how to communicate with it have to be provided there. The example program distributed with this package can be a good starting point to check out the commands and to see if you get what you want from the other application. There is a certain DDE terminology which might be unfamiliar to you. In that case have a look at the Glossary. For any questions, comments, suggestions and your user experience please contact me via email.

2 Realization

Ddeclient.ss relies on a C/C++ dynamic link library DDECLient.dll, which provides the DDE connectivity and exports the necessary C-functions and data. It's written for use with other programs as well, so if you are interested ask me for additional material like docs, header and examples. Ddeclient.ss imports the procedures of DDECLient.dll and provides easier to use procedures for scheme, that user's don't have to worry about freeing memory for example. A disadvantage over this approach is that data returned to the client is converted to scheme values by copying thus using additional memory.

3 Features

This package is beta release software.

- Enables DDE connections between one client up to 20 services/server applications.
- Request, execute and poke transactions can be performed synchronously and asynchronously. Advise transactions are not implemented since I don't have any idea how they work and no examples to test.
- Errors occurring during a conversation are reported and the the program is terminated. They are displayed in the interactions windows, if DrScheme is used.

4 Requirements

The package was developed and tested under Windows XP SP2 and PLT Scheme 350.4. Anything higher should work anything lower since the introduction of the Foreign Function Interface should also work, but as always there is no guarantee in both directions.

5 Files and Directories

This package contains the following main files and directories:

- **/docs** Directory of documentation files
- **/compiled** Directory for compiled module files
- **/dll** Directory for source and additional files of DDEClient.dll
- **/dll/DDEClient.dll** C/C++ dynamic link library for DDE connectivity
- **ddeclient.ss** Interface module between DDEClient.dll and PLT-Scheme
- **ddeclientsample.ss** Sample program
- **info.ss** Package specific file for installation

6 Getting Started

The package contains a sample file `ddeclientsample.ss`. It's a commented module file which shows examples of the DDE procedures provided by `ddeclient.ss`. It uses Microsoft Word as service application, since most Windows users will have it on their machines.

Load `ddeclientsample.ss` in DrScheme with,

```
(require (planet "ddeclientsample.ss"
                ("mato" "ddeclient.plt" 1 0)))
```

Open Microsoft Word, make sure you have write access to your `C:\` drive and there is no file named `test.doc`. If you don't have write access to `C:\` search for the file name `C:\test.doc` in `ddeclientsample.ss` and replace it with what ever suits you. The easiest way to open `ddeclientsample.ss` is to run the require expression above without having opened Word. This causes an error in the interaction window and you can click on its page icon to open the file. Read through the file. It provides a procedure `(run)`. Type `(run)` in the interaction window and look at the output. After that experiment with it, copy blocks and change them in your own programs.

To load `ddeclient.ss` into your own programs require it as follows:

```
(require (planet "ddeclient.ss" ("mato" "ddeclient.plt" 1 0)))
```

7 Structure of a DDE process

To implement a DDE conversation you need to set up a number of steps which are described below. As structure of DDE process in this document is regarded the set of commands which is needed for a successful DDE conversation. The procedures provides by ddeclient.ss have to be used as follows.

dde-init

Initializes DDEML.dll, which provides the DDE features of the operating system

dde-connect

Establishes DDE connection to a service

One or more of the following procedures do the actual DDE transaction:

dde-cmd

universal command for execute, poke and request transactions

dde-request

request for data

dde-request-string

request for data as string

dde-execute

execute a command in the service application, i.e. menu command to open a certain file

dde-poke

send data to the service application

dde-disconnect

Closes down the DDE connection to a service of a server, frees memory

dde-uninit

Uninitializes DDEML.dll

8 Procedures

8.1 List of procedure arguments

Most of the procedure arguments are the same ones for different procedures. They are described here together.

- **conv**
[number] conversation number, returned by dde-connect, index number to refer to a certain dde conversation.

- **type**
[string] type of DDE transaction can be "execute", "poke" or "request".
- **item**
[string] the command you want to send to the service. I.e. to see what topics are available you can set it to "topics" in a request transaction, after connecting the service with the topic "system", but not all programs support this topic. The available commands are dependent on the service and topic. Look up the documentation of the service for information about them. Note: A major drawback of using DDEML.dll is that **item** must not exceed 255 characters. If it is longer there will be an error.
- **data**
[string] data you send by a poke command to the service.
- **format**
[string] format of DDE data. It determines in which format the data between the applications is sent. See 9 below.
- **timeout**
[number] indicates the waiting time in milliseconds until the transaction waits to be finished before returning to the program. If the number is greater 0 and the service doesn't return in time the transaction is terminated. 0 indicates an asynchronous transaction, which means the transaction is waiting for the service's answer as long as it takes.
- **access**
[string] flag in which format DDE data is copied to DDEClient.dll variables. It is meant for convenience since a lot of conversations only use ascii strings. It can be "byte" or "string". Has to be "byte" in case strings other than ascii format should be read. In that case the returned byte-string has to be converted accordingly, i.e. by `bytes->string/utf-8`.

8.2 List of procedures

- **(dde-init)**
Initialize DDEML.dll the basic dll for DDE. DDEML.dll is installed with the operating system, so normally there is nothing to do to access it. To be called once in a program before any other DDE procedure.
- **(dde-uninit)**
End usage of DDEML.dll. To be called after the last dde procedure of a program.
- **(dde-connect service topic) -> number**
Establish a connection to a service for a conversation. Returns a number which distinguishes the conversation from others. Service is a string containing the name of a DDE server application, i.e. "winword", usually the service name is the name of the programs exe-file. Topic is a string of the

topic the service is connected for. There are different themes over which you can communicate with the service. Often a topic is a file name which is currently open in the service application. Most servers support also a topic "system".

- `(dde-disconnect conv)`
Cancel a conversation and stop the connection to a service; meaning of argument see 8.1.
- `(dde-cmd conv type item data format timeout access)` -> **multiple**
General procedure for execute, poke and request transactions, the communication with the service; meaning of arguments see 8.1. The return value depends on `type` and `access`. Only `type = "request"` will have a return value. `access = "byte"` returns data as byte-string. `access = "string"` returns data as string.
- `(dde-request conv item format timeout)` -> **byte-string**
Abbreviation of a `dde-cmd` request transaction, which returns DDE data as byte-string; meaning of arguments see 8.1. The byte-string leave the flexibility how the data are further analyzed to the programmer.
- `(dde-request-string conv item timeout)` -> **string**
Abbreviation of a `dde-cmd` request transaction, which returns DDE data as string; meaning of arguments see 8.1.
- `(dde-execute conv item timeout)`
Abbreviation of a `dde-cmd` execute transaction, format is `CF_TEXT`; meaning of arguments see 8.1.
- `(dde-poke conv item data timeout)`
Abbreviation of a `dde-cmd` poke transaction, format is `CF_TEXT`; meaning of arguments see 8.1.
- `(dc-dde-version)` -> **string**
returns a string with information about the version of DDEClient.dll.

9 Supported Clipboard Formats

The following formats can be used to determine how the DDE data will be formatted between service and client. Microsoft uses clipboard formats to format data which is exchanged via DDE. Such it is possible not only to transfer text but also other data, for example pictures. If you use anything else than the `CF_TEXT` format your `access` argument must be set to "byte" and you have to handle the resulting byte string on your own to get the desired data, which may be a lot of work. The formats themselves are described in the Windows SDK. There are a lot more formats but only the following are available through the DLL. Write the format strings in capitals exactly as they are given here.

- **CF_TEXT**
Null-terminated, plain ANSI text in a global memory block.
- **CF_BITMAP**
A bitmap compatible with Windows 2.x.
- **CF_METAFILEPICT**
A Windows metafile with some additional information about how the metafile should be displayed.
- **CF_SYLK**
An ASCII text format used by some older Microsoft products.
- **CF_DIF**
Software Art's data interchange format (DIF). Also an ASCII text format.
- **CF_TIFF**
Tag image file format (TIFF) data in a global memory block.
- **CF_OEMTEXT**
Similar to **CF_TEXT** but using the OEM character set.
- **CF_DIB**
A global memory block containing a Windows device-independent bitmap (DIB) as a BITMAPINFO structure followed by the bitmap bits.
- **CF_PALETTE**
A color-palette handle. (Used in conjunction with **CF_DIB**.)
- **CF_PENDATA**
Data is for the pen extensions to Windows.
- **CF_RIFF**
Resource interchange file format (RIFF) data as a global memory block.
- **CF_WAVE**
A specific case of RIFF in which the contained data is a waveform (sampled sound).

10 Thanks

Eli Barzilay and colleagues thanks for the foreign function interface. Without it the project wouldn't be possible.

I owe a lot concerning the C/C++, DDE part of this project to the following people, companies through their books or presence of their work on the internet:

www.angelfire.com/biz/rhaminisys/ddeinfo.html for their info on DDE and their example archive `xlddec.zip`

Jürgen Wolf and Galileo Computing for his very instructive book on C programming “C von A bis Z”, at www.galileo-press.de/openbook/c_von_a_bis_z/

Steven Randy Davis for his book “C++ for Dummies”

www.functionx.com who ever are behind it, for their perfect tutorial on creating dlls at www.functionx.com/visualc/libraries/win32dll.htm, I wouldn't have started without it.

Faweb, Takebishi for their DDE example at http://www.faweb.net/us/ioserver/sample_vc.html

Graeme S. Roy for his library mpatrol to check this dll for memory leaks at www.cbmamiga.demon.co.uk/mpatrol/

Robert Schmitt for his article and binaries to get mpatrol started so quickly at www.codeguru.com/cpp/w-p/win32/tutorials/article.php/c12231/.

Microsoft Knowledgebase for some articles like KB279721, but the DDE documentation is mostly quiet unreadable in my opinion.

11 Glossary

Client Application that asks for the conversation, here the scheme program.

Service In earlier times called server, the application which is to be controlled or serves the client with data. It is normally the name of the application's exe-file, example: "winword".

Transaction The interaction between client and server, which is of a certain type like "request", "execute", "poke".

Conversation The connection and what happens between client and service. Since a client can have more than one conversation at the same time with different services each conversation is indexed by a distinct number.

Synchronous transaction In a synchronous transaction the client sends a transaction to the service and waits a certain time for the service to finish. It returns the control to program only if the service finishes the transaction or if the timeout reached.

Asynchronous transaction In an asynchronous transaction the client does not wait for the service to answer, but returns the control to the program after sending the transaction. To get the data the program has explicitly to ask if the service has returned data. This kind of transaction is preferable if the duration of a transaction is not known. The dde-... commands handle asynchronous transactions in a way that they seem to be synchronous transactions without a timeout. This is very convenient for a lot of applications.

12 License

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 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.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages—typically libraries—of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, 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 and use pieces of it in new free programs; and that you are informed that you can do these things.

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

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the

Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a “work based on the library” and a “work that uses the library”. The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND
MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called “this License”). Each licensee is addressed as “you”.

A “library” means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The “Library”, below, refers to any such software library or work which has been distributed under these terms. A “work based on the Library” means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term “modification”.)

“Source code” for a work means the preferred form of the work for making modifications to it. For a library, 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 library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete 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 distribute a copy of this License along with the Library.

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 Library or any portion of it, thus forming a work based on the Library, 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) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, 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 Library, 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 Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you 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.

If distribution of 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 satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a “work that uses the Library”. Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a “work that uses the Library” with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a “work that uses the library”. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a “work that uses the Library” uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a “work that uses the Library” with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer’s own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
- b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user’s computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.
- c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
- d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
- e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the “work that uses the Library” must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be 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.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate

distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library 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.

9. 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 Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library 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 with this License.

11. 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 Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library 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 Library.

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.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library 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.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser 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 Library 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 Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, 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

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY “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 LIBRARY IS WITH YOU. SHOULD THE

LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. 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 LIBRARY 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 LIBRARY (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 LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), 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 Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. 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 library’s name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library 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 Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; 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.

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

Yoyodyne, Inc., hereby disclaims all copyright interest in the library
'Frob' (a library for tweaking knobs) written by James Random
Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Index

asynchronous, 8

client, 8

conversation, 8

dc-dde-version, 6

DDE Client, 1

dde-cmd, 6

dde-connect, 5

dde-disconnect, 6

dde-execute, 6

dde-init, 5

dde-poke, 6

dde-request, 6

dde-request-string, 6

dde-uninit, 5

DDEClient, 1

service, 8

synchronous, 8

transaction, 8