How to migrate from earlier Multilizer versions to Multilizer 2009

Multilizer 2009 has a complete set of migration paths for users that have localized their Delphi/C++Builder software with any earlier version of Multilizer.

Multilizer 2009 Products

The following Multilizer products support localization of any Delphi/C++Builder software:

- Multilizer 2009 app can be any of the following products:
    - Multilizer 2009 Enterprise or
    - Multilizer 2009 Pro for Developers or
    - Multilizer 2009 Lite for Developers
- Multilizer Localization components ship with any of the products mentioned above.
- M7P2MLD is a separate (tool) product that converts a Multilizer 2009 project to a Multilizer Binary Dictionary file (.mld).
- Delphi/C++Builder Source scanner plug-in is a separate plug-in product that enables Multilizer 2009 to be used exactly as in Multilizer 5. I.e., this scanner reads strings to translate from application source code and writes out the .mld file.
- MLD Editor is a separate tool that allows users to edit the MLD[1] dictionary directly. This tool is needed e.g. when end-user edits translations.

---

[1] MLD dictionary is a file that contains the translations of the software. The MLD file is used by Multilizer localization components.

1. Upgrade from Multilizer 2007

Simply open the Multilizer 2007 projects and continue working in Multilizer 2009.

2. Upgrade from earlier Multilizer versions

This applies to Multilizer version 1 through 6.

Before upgrading you should know, whether Multilizer components are used or not.

- All localization projects of Multilizer 1-4 are component localization projects.
- Multilizer 5 offered both component localization projects and binary localization projects.
- Multilizer 6 supported only binary localization.

2.1. Upgrade of binary localization projects

Support for binary localization was added in Multilizer 5. In binary localization Multilizer reads translations from the executable file and writes out localized executables files or resource files.

Multilizer components are not needed in binary localization. This was one of the reasons they were dropped from Multilizer 6. [2]

Migration

Multilizer 2009 supports binary localization out-of-the-box. Users that want to view and edit VCL forms visually need either Multilizer Pro for Developers or Multilizer Enterprise.

Required products:

- Multilizer 2009 Enterprise
  OR
- Multilizer 2009 Pro for Developers
  OR
- or Multilizer 2009 Lite for Developers

To-dos:

Migration of Multilizer 5 and Multilizer 6 binary localization projects is done in two steps:

1. Create a Multilizer 2009 localization project and choose the executable to localize.
2. Use Import Wizard to import translations from the old project to the new one

---

[2] Multilizer localization components for Delphi and C++Builder were re-introduced in Multilizer 2007. Multilizer 2009 introduces support for the latest Delphi/C++Builder versions 2000 and 2010.

2.2. Upgrade of component localization projects

Multilizer 2009 includes Multilizer Localization Components. In order to fully support component localization, there is a Delphi/C++Builder Source scanner plug-in. This scanner plug-in works as Multilizer 5 or older; it reads the strings from a Delphi/C++Builder project and writes out a dictionary file (.mld).

Migration

The following migration path is referred as "no changes" scenario in Appendix A. See Appendix A for other ways of upgrading,

Required products:

- Multilizer 2009 application (**Multilizer 2009 Enterprise** or **Multilizer 2009 Pro for Developers**) AND
- **Delphi/C++Builder Source scanner** plug-in.

To get the support for latest Delphi/C++Builder versions the user may need to install the Multilizer Localization Components. The installers are included in Multilizer application setups.

To-dos:

Migration from older Multilizer component localization projects is done in 2–3 steps:

1. Create a Multilizer 2009 localization project and choose the Delphi/C++Builder project to localize.
2. Use Import Wizard to import translations from the old project to the new one
3. (Optional) Install the latest version of Multilizer Localization Components in Delphi/C++Builder IDE.

Appendix A: Upgrade Path Summary

The table below summarizes the different upgrade scenarios for users of Multilizer components in Multilizer 5.x or older.

- **Upgrade scenario** column summarizes the upgrade in one sentence.
- **Product needed** column tells which products user needs to upgrade to.
- **Pros** and **Cons** columns compare the pros and cons with Multilizer 5 in respect to localized software.
- **Upgrade tasks** tells what the user needs to do, in order to get the software localized[3].

Upgrade scenario	Product needed	Pros	Cons	Upgrade tasks
Switch to VCL Localization Architecture	1) Multilizer 2009 app	1) Localization architecture follows the "VCL standard" 2) No penalty in run-time performance	1) Several files can be generated (one file for each language and exe localized) 2) The language change on run-time does not work that well.	1) Remove all Multilizer components. 2) Convert all Translate() functions to use resourcestring instead. 3) Convert all hard-coded strings to resourcestring.
Switch to IvResDLLDictionary	1) Multilizer 2009 app 2) Multilizer Localization Components	1) Working language change on run.time	1) Several files can be generated (one file for each language and exe localized) 2) Moderate performance penalty in starting app or changing language.	1) Convert all Translate() functions to usage of resourcestring. 2) Convert all hard-coded strings to resourcestring.
Continue using MLD Dictionary (IvBinaryDictionary)	1) Multilizer 2009 app 2) Multilizer Localization Components 3) M7P2MLD app	–	–	1) Convert all Translate() functions to usage of resourcestring. 2) Convert all hard-coded strings to resourcestring.
"No changes" [4]	1) Multilizer 2009 app[5] 2) Multilizer Localization Components 3) Delphi/C++Builder Source scanner	–	–	–

---

[3] In all scenarios the user needs to create a Multilizer 2009 localization project and import existing translations from the old project.
[4] This scenario requires MLDScanner plugin. Final decision on implementation to be done within a month.
[5] Either Multilizer 2009 Enterprise or Multilizer 2009 Pro for Developers