Building Applications Examples – Part 3
CS and SourcesFiles

By Josh Fletcher, Technical Services Team Member, 4D Inc.

Technical Note 08-14

Abstract

4D 2004 features over 80 XML keys that can be used to build customized applications. The XML keys documentation gives basic information about what the keys do, but there are few examples. The goal of this series of Technical Notes is two-fold:

- Provide examples for each XML key, on Mac OS X and Windows, to aid the 4D developer in understanding what each key does.

- To make editing the XML project file, used in building applications, an easier task by providing a GUI editor.

Part 1 of this series provided a 4D database that can be used to edit project files.

Part 2 of this series provided examples for the XML keys from the General Parameters and Licenses themes.

In part 3 of this series, examples for the XML keys from the CS and SourcesFiles themes are presented.

Table of Contents

Introduction

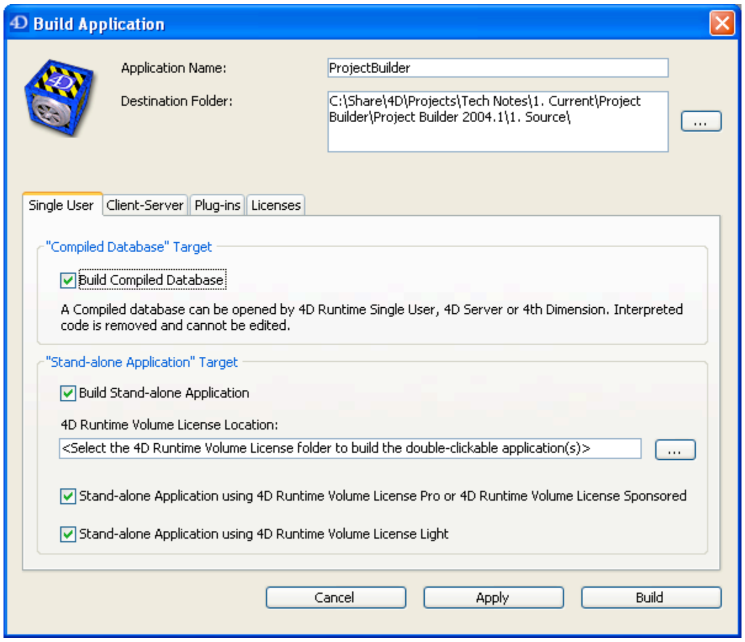
4D 2004 features the ability to control the application building process with the use of an XML project file. There are over 80 XML keys that can be used to build customized applications. The project file can contain some or all of these XML keys, as needed.

These XML keys allow many parts of a 4D application to be customized; for example the application name can be changed, or the path to the data file can be altered. The 4D 2004 XML Keys documentation can be downloaded from here:

http://www.4d.com/support/documentation.html

(The title is listed as "XML Keys BuildApplication")

The XML keys documentation gives basic information about what the keys do, but there are few examples. Similarly the "Build Application" dialog found in 4D offers access to only a subset of the XML keys available (approximately 15 of the keys can be edited with this dialog):



Application Name: ProjectBuilder

Destination Folder: C:\Share\4D\Projects\Tech Notes\1. Current\Project Builder\Project Builder 2004.1\1. Source\

"Compiled Database" Target

☑ Build Compiled Database

A Compiled database can be opened by 4D Runtime Single User, 4D Server or 4th Dimension. Interpreted code is removed and cannot be edited.

"Stand-alone Application" Target

☑ Build Stand-alone Application

4D Runtime Volume License Location:

<Select the 4D Runtime Volume License folder to build the double-clickable application(s)>

☑ Stand-alone Application using 4D Runtime Volume License Pro or 4D Runtime Volume License Sponsored

☑ Stand-alone Application using 4D Runtime Volume License Light

If access to the other keys is needed, the **BUILD APPLICATION** command must be used and the XML project file must be edited manually. Documentation for the BUILD APPLICATION command can be found here:

http://www.4d.com/4ddoc2004/CMU/CMU00871.HTM

The goal of this Technical Note series is two-fold:

- Provide examples for each XML key, on Mac OS X and Windows, to aid the 4D Developer in understanding what each key does.

- To make editing the XML project file an easier task by providing a GUI editor for them.

The XML keys are organized into themes as follows:

- **General Parameters –** These are keys that are not specific to single-user, server, nor client applications.
- **Licenses –** These keys are used to specify the location of license files necessary for building applications.
- **CS –** These keys are specific to building client-server applications.
- **Sources Files –** These keys are concerned with the files that go into building applications, e.g. paths to 4D software, icon files, etc.
- **Versioning –** These keys are used to alter the meta-information for the merged application, e.g. version and copyright information.

This document will cover the examples for the XML keys from the CS and SourcesFiles themes. These examples are intended to better explain what each XML key does.

Terminology

This section defines some of the terms used in this Technical Note, for clarification.

- **XML key –** Refers to a single XML element within the list of XML elements that 4D has defined for use with the application building features.

- **Theme –** The XML keys available are organized into 5 themes: General Parameters, Licenses, CS, SourcesFiles, and Versioning.

- **Compiled structure –** A compiled 4D database (.4dc file).

- **Merged application –** Refers to a 4D database that has been "merged" with 4D software in order to create a single application. Also known as "stand-alone" or "built".

- **Merged single-user application –** Refers to a 4D database that has been merged with the 4D Runtime Volume License software.

- **Merged server application –** Refers to a 4D database that has been merged with the 4D Server software.

- **Merged client application –** Refers to the client portion of a merged client-server database. In this case there is no 4D database but the 4D Client software may have been customized as part of the build process.

- **Install image** – This term is used to describe all of the files that make up the installation of a merged application. For example a merged 4D Client install image contains the executable for the application, various libraries and resource files, etc. All of these files are collectively referred to as the "install image".

- **Install client –** The install image of a merged client application. This is the software that would be manually installed on any given computer that will be used to connect to the merged server application.

- **Upgrade client –** The install image that is downloaded to the client machine for the purposes of automatically upgrading. This is not the software that would be manually installed on a given client computer. It is used by the automatic upgrade feature only.

Structure of the Examples

The example sections are organized by theme. Furthermore the XML keys appear in the same order as they do in the *4th Dimension XML Keys BuildApplication* documentation.

For each XML key example the following information is presented:

- A short, general description of what the key does.
- Dependencies on other keys are listed, if applicable.
- Windows example:
    - Windows-specific description (if applicable)
    - Compiled structure example (if applicable)
    - Single-user example (if applicable)
    - Client example (if applicable)
    - Server example (if applicable)
- Mac OS X example:
    - Mac-specific description (if applicable)
    - Compiled structure example (if applicable)
    - Single-user example (if applicable)
    - Client example (if applicable)
    - Server example (if applicable)

Within each example the value of the XML key being tested, and optionally the values of other related keys, is shown. Please note that the XPATH of the keys is ignored. Keys from different themes will be presented side by side, as in:

```
<BuildApplicationLight>True</BuildApplicationLight>
<RuntimeVLIncludeIt>True</RuntimeVLIncludeIt>
```

The BuildApplicationLight key is not a sibling of the RuntimeVLIncludeIt key, in terms of XPATH notation. This is done to simplify the examples but remember that the keys must be placed at the correct location within the project file for them to work.

CS

This section covers examples of the XML keys from the “CS” theme.

BuildServerApplication

This key tells 4D whether or not to build a merged server application.

This key can be set to “True” or “False”.

If set to “True”, this key depends on the keys ServerWinFolder and/or ServerMacFolder. These keys must have a valid path to the 4D Server software or the build will fail.

This key also depends on the ServerIncludeIt key. If ServerIncludeIt is not set to “True” the server application will not be built.

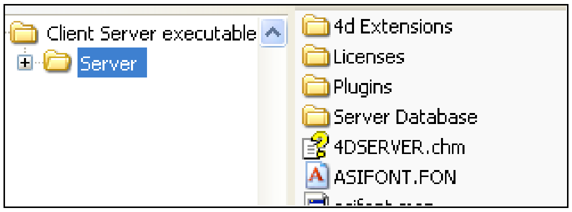
The resulting application is placed at the path “Client Server executable/Server” at the location specified by the key BuildWinDestFolder or BuildMacDestFolder or, if those keys were not used, next to the structure file.

Example

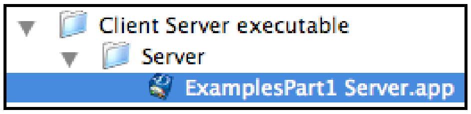
The BuildServerApplication key was set as follows:

```
<BuildServerApplication>True</BuildServerApplication>
```

Windows



Mac OS X



BuildCSUpgradeable

This key tells 4D to enable the automatic client upgrade feature and build the **upgrade client**. For more information on the automatic client upgrade feature see Technical Note 07-22, *Building Applications with 4D 2004: Automatic Client Upgrade*.

This key can be set to “True” or “False”.

The upgrade clients are placed in a folder called “Upgrade4DClient” within the merged server application. The upgrade client for Windows is called “archive.win” and for Mac OS X it is “archive.mac”.

Note that this key is useless unless the CurrentVersion key is also used.

Example

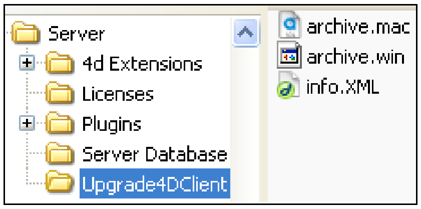
In this example the upgrade clients were created for both Mac OS X and Windows.

Windows

This key depends on the ClientWinIncludeIt and ClientWinFolderToWin keys on Windows. Optionally, if building the upgrade client for Mac OS X, it also depends on the ClientMacIncludeIt and ClientMacFolderToWin keys.

The keys were set as follows:

```
<BuildCSUpgradeable>True</BuildCSUpgradeable>
<ClientWinIncludeIt>True</ClientWinIncludeIt>
<ClientMacIncludeIt>True</ClientMacIncludeIt>
<ClientWinFolderToWin>C:\4D\4D 2004.7r3\4D Client 2004.7\</ClientWinFolderToWin>
<ClientMacFolderToWin>C:\4D\4D 2004.7r3\4D Client.app\</ClientMacFolderToWin>
```



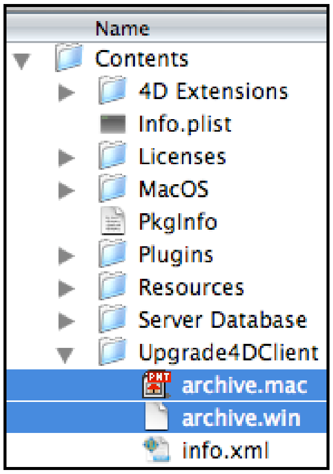
Mac OS X

This key depends on the ClientMacIncludeIt and ClientMacFolderToMac keys on Mac OS X. Optionally, if building the upgrade client for Windows, it also depends on the ClientWinIncludeIt and ClientWinFolderToMac keys.

The keys were set as follows:

```
<BuildCSUpgradeable>True</BuildCSUpgradeable>
<ClientWinIncludeIt>True</ClientWinIncludeIt>
<ClientMacIncludeIt>True</ClientMacIncludeIt>
<ClientWinFolderToMac>MacHD:Applications:4D Client:</ClientWinFolderToMac>
<ClientMacFolderToMac>MacHD:Applications:4D Client.app</ClientMacFolderToMac>
```

Tip: *In a merged server application on Mac OS X,, the "Upgrade4DClient" folder is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



```
Name
▼ Contents
    ▶ 4D Extensions
      Info.plist
    ▶ Licenses
    ▶ MacOS
      PkgInfo
    ▶ Plugins
    ▶ Resources
    ▶ Server Database
    ▼ Upgrade4DClient
          archive.mac
          archive.win
          info.xml
```

IPAddress

This key allows you to specify the eventual IP address of the merged server application.

A merged client already has the ability to automatically connect to the corresponding merged server on the default port. Furthermore, once connected, the merged client saves the server information in a file called "EnginedServer.xml" which is created in the "4D Extensions" folder. Once this file has been created, the merged client looks there for the server info.

However if it is desirable to "hard code" the IP address, or if a custom port is in use, this key helps avoid issues the first time the merged client is launched. Specifically, if there is no "EnginedSever.xml" file, the first time the merged client is launched it will simply broadcast for any matching server on the network. If there is more than one server that matches the same client, the connection will be made to the first one that responds. Similarly, if a custom port is in use, the connection attempt will simply time out and the user will be asked to specify a server. This key causes a prebuilt "EnginedServer.xml" file to be created for the merged client application, thus avoiding these issues.

If BuildCSUpgradable is set to "True", this key also affects the "info.xml" file in the merged server application.

Note: *This key has no effect on the actual IP address of the merged server. 4D handles this internally, based on the IP address of the machine.*

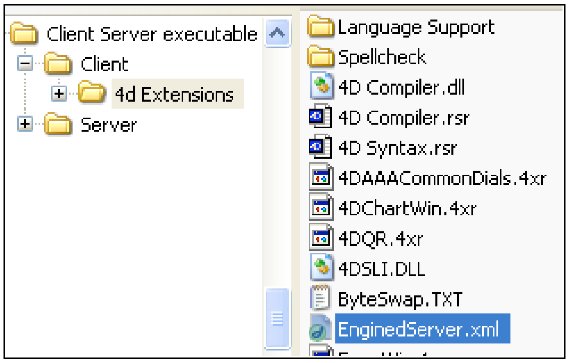
Example

The keys were set as follows:

```
<IPAddress>10.96.0.79</IPAddress>
<BuildCSUpgradeable>True</BuildCSUpgradeable>
```

Windows

Client application

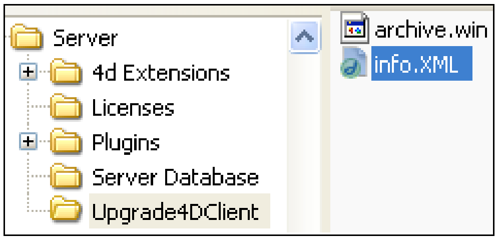


Location of EnginedServer.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<Connection>

  <IPAdress>10.96.0.79</IPAdress>

</Connection>
```

Contents of EnginedServer.xml

Server application



```
Server
  4d Extensions
  Licenses
  Plugins
  Server Database
  Upgrade4DClient
archive.win
info.XML
```

Location of info.xml

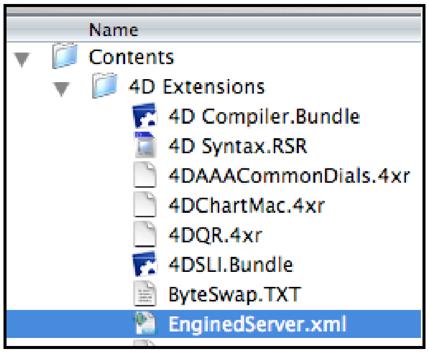
```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
<BuildName>ExamplesPart1</BuildName>
<BuildIPAdress>10.96.0.79</BuildIPAd
ress>
...
```

Contents of info.XML (partial)

Mac OS X

Client application

Tip: *In a merged client application on Mac OS X, the "EnginedSever.xml" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



```
Name
Contents
  4D Extensions
    4D Compiler.Bundle
    4D Syntax.RSR
    4DAAACommonDials.4xr
    4DChartMac.4xr
    4DQR.4xr
    4DSLI.Bundle
    ByteSwap.TXT
    EnginedServer.xml
```

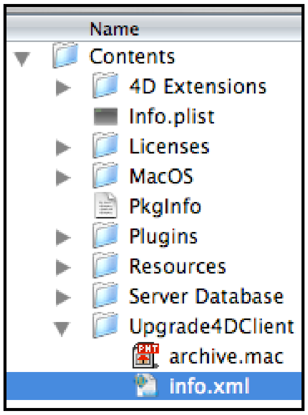
Location of EnginedServer.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<Connection>

  <IPAdress>10.96.0.79</IPAdress>

</Connection>
```

Contents of EnginedServer.xml

Server application

Tip: *In a merged server application on Mac OS X, the "info.xml" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



```
Name
Contents
  4D Extensions
  Info.plist
  Licenses
  MacOS
  PkgInfo
  Plugins
  Resources
  Server Database
  Upgrade4DClient
    archive.mac
    info.xml
```

Location of info.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
<BuildName>ExamplesPart1</BuildName>
<BuildIPAdress>10.96.0.79</BuildIPAd
ress>
...
```

Contents of info.XML (partial)

PortNumber

This key allows you to specify the eventual port number of the merged server application.

A merged client already has the ability to automatically connect to the corresponding merged server on the default port. Furthermore, once connected, the merged client saves the server information in a file called “EnginedServer.xml” which is created in the “4D Extensions” folder. Once this file has been created the merged client looks there for the server info.

However if a custom port is in use, this key helps avoid issues the first time the merged client is launched. Specifically, if there is no “EnginedSever.xml” file, the first time the merged client is launched it will simply broadcast for any matching server on the network, but on the default port of 19813. If no server is found the user will experience a -10002 error and then be shown the “Connect to 4D Server” dialog. This key causes a prebuilt “EnginedServer.xml” file to be created for the merged client application, thus avoiding these issues.

The PortNumber key must be used in conjunction with IPAddress, otherwise it has no effect.

If BuildCSUpgradable is set to “True”, this key also affects the “info.xml” file in the merged server application.

Note: *This key has no effect on the actual port number of the server. The port must be set in the Preferences of the database, or with the SET DATABASE PARAMETER command.*

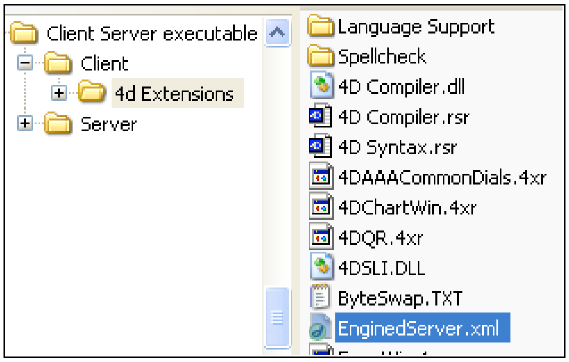
Example

The keys were set as follows:

```
<PortNumber>19815</PortNumber>
<IPAddress>10.5.0.5</IPAddress>
<BuildCSUpgradeable>True</BuildCSUpgradeable>
```

Windows

Client application

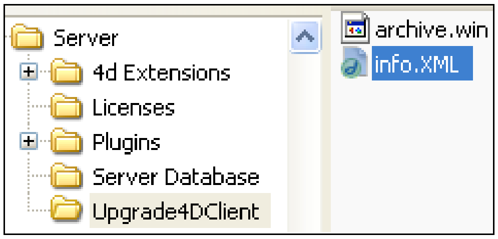


Location of EnginedServer.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<Connection>

<IPAdress>10.5.0.5:19815</IPAdress>

</Connection>
```

Contents of EnginedServer.xml

Server application



Location of info.xml

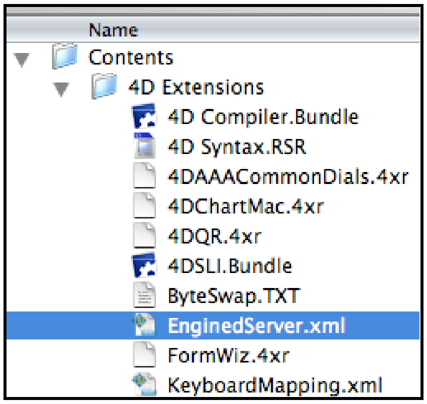
```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
<BuildName>ExamplesPart1</BuildName>
<BuildIPAdress>10.5.0.5</BuildIPAdress>
<BuildIPPort>19815</BuildIPPort>
...
```

Contents of info.XML (partial)

Mac OS X

Client application

Tip: *In a merged client application on Mac OS X, the "EnginedSever.xml" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



Name

- Contents
  - 4D Extensions
    - 4D Compiler.Bundle
    - 4D Syntax.RSR
    - 4DAAACommonDials.4xr
    - 4DChartMac.4xr
    - 4DQR.4xr
    - 4DSLI.Bundle
    - ByteSwap.TXT
    - EnginedServer.xml
    - FormWiz.4xr
    - KeyboardMapping.xml

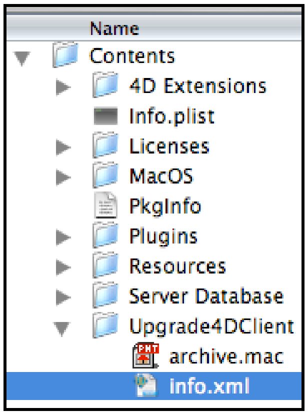
Location of EnginedServer.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<Connection>

<IPAdress>10.5.0.5:19815</IPAdress>

</Connection>
```

Contents of EnginedServer.xml

Server application

Tip: *In a merged server application on Mac OS X, the "info.xml" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



Name

- Contents
  - 4D Extensions
  - Info.plist
  - Licenses
  - MacOS
  - PkgInfo
  - Plugins
  - Resources
  - Server Database
  - Upgrade4DClient
    - archive.mac
    - info.xml

Location of info.xml

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
<BuildName>ExamplesPart1</BuildName>
<BuildIPAdress>10.5.0.5</BuildIPAdress>
<BuildIPPort>19815</BuildIPPort>
...
```

Contents of info.XML (partial)

HardLink

The value specified by this key appears in the “info.XML” file, which is created when building a merged server with the automatic client upgrade feature enabled.

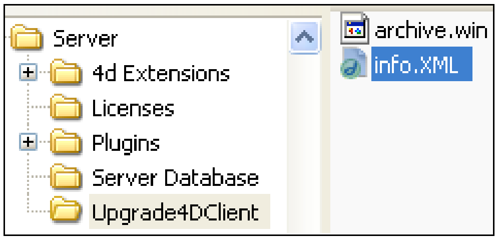
However, note that this key has no effect on the runtime behavior of the client-server application. For example, a client with a different HardLink can still connect to the merged server.

Example

The HardLink key was set as follows:

```
<HardLink>MyLink</HardLink>
```

Windows



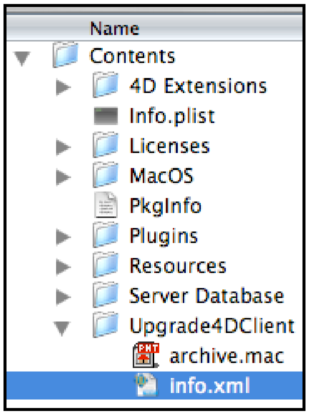
Location of info.XML

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
 <BuildName>ExamplesPart1</BuildName>
 <BuildDoSplash>True</BuildDoSplash>
 <BuildHardLink>MyLink</BuildHardLink>
...
```

Contents of info.XML (partial)

Mac OS X

Tip: *In a merged server application on Mac OS X, the “info.xml” file is located within the application package. To get inside the package, right-click on the application and choose “Show Package Contents”.*



Location of info.XML

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
 <BuildName>ExamplesPart1</BuildName>
 <BuildDoSplash>True</BuildDoSplash>
 <BuildHardLink>MyLink</BuildHardLink>
...
```

Contents of info.XML (partial)

RangeVersMin, RangeVersMax, CurrentVers

These three keys are concerned with the automatic client upgrade feature. Generally speaking, if the CurrentVers of the client does not match the CurrentVers of the server, then the client will be asked to upgrade. If the CurrentVers of the client falls within the range specified by RangeVersMin and RangeVersMax then it passes the version check and is allowed to attempt to connect. For in-depth information on the automatic client upgrade feature see Technical Note 07-22, *Building Applications with 4D 2004: Automatic Client Upgrade*.

The values should be integers; alphabetical characters are ignored, as are floating point numbers. If only CurrentVers is used, RangeVersMin and RangeVersMax will have the same value as CurrentVers.

Note: *These values are not exposed externally so they need not represent the "version" of the product. They are only used in determining if the client should be upgraded or not.*

The version information is stored in a file called "info.XML" within the merged server package after building. This information is not stored externally in the merged client.

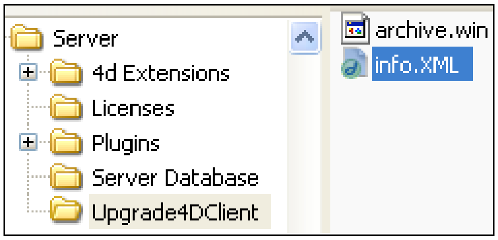
Tip: *The automatic client upgrade feature cannot and will not override 4D's own version checking. For example, if the client connecting had CurrentVers 1 when it was built with 4D 2004.4 and the server has a CurrentVers of 1 when it was built with 4D 2004.7 then the CurrentVers for both obviously matches; the client will not be asked to upgrade. However, the client will still fail to connect because the 4D versions are not compatible.* ***Always be sure to increment the CurrentVers (and RangeVersMin if applicable) when merging with a new 4D version.***

Example

The keys were set as follows:

```
<CurrentVers>5</CurrentVers>
<RangeVersMin>2</RangeVersMin>
<RangeVersMax>5</RangeVersMax>
```

Windows



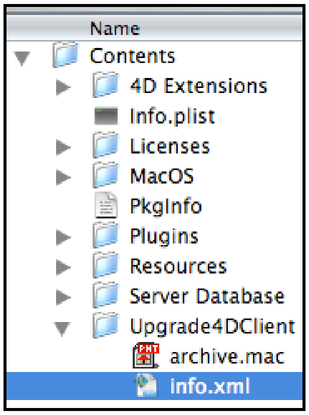
Location of info.XML

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
...
  <BuildRangeVersMin>2</BuildRangeVersMin>
  <BuildRangeVersMax>5</BuildRangeVersMax>
  <BuildCurrentVers>5</BuildCurrentVers>
</BuildInfo>
```

Contents of info.XML (partial)

Mac OS X

Tip: *In a merged server application on Mac OS X, the "info.xml" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



Name
- Contents
  - 4D Extensions
  - Info.plist
  - Licenses
  - MacOS
  - PkgInfo
  - Plugins
  - Resources
  - Server Database
  - Upgrade4DClient
    - archive.mac
    - info.xml

Location of info.XML

```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<BuildInfo>
...
  <BuildRangeVersMin>2</BuildRangeVersMin>
  <BuildRangeVersMax>5</BuildRangeVersMax>
  <BuildCurrentVers>5</BuildCurrentVers>
</BuildInfo>
```

Contents of info.XML (partial)

SourcesFiles: RuntimeVL

The “SourcesFiles” theme is split into two subthemes: “RuntimeVL” and “CS”. This section covers examples of the XML keys from the “RuntimeVL” subtheme.

RuntimeVLIncludeIt

This key tells 4D whether or not to build a merged single-user application.

This key can be set to “True” or “False”.

This key depends on the BuildApplicationSerialized key. If BuildApplicationSerialized is not set to “True” then the merged application will not be built.

This key also depends on the RuntimeVLWinFolder or RuntimeVLMacFolder key, on Windows and Mac OS X respectively. If these keys do not contain a valid path to the 4D Runtime Volume License software then the merged application will not be built.

The resulting application is placed in a folder named “Final Application” at the location specified by the key BuildWinDestFolder or BuildMacDestFolder or, if those keys were not used, next to the structure file.

Example

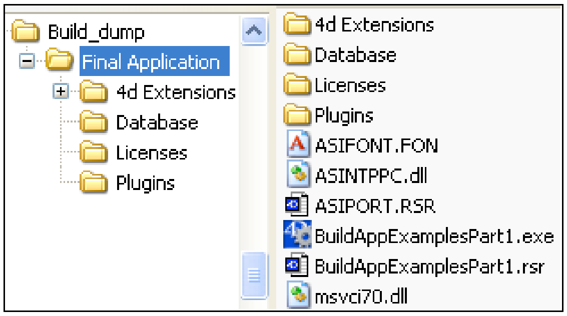
The keys were set as follows:

```
<BuildApplicationSerialized>True</BuildApplicationSerialized>
<RuntimeVLIncludeIt>True</RuntimeVLIncludeIt>
```

Windows

The Windows keys were set as follows:

```
<RuntimeVLWinFolder>C:\4D Runtime Volume License 2004.7\</RuntimeVLWinFolder>
```

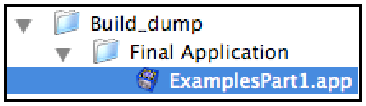

Build_dump
Final Application
4d Extensions
Database
Licenses
Plugins

4d Extensions
Database
Licenses
Plugins
ASIFONT.FON
ASINTPPC.dll
ASIPORT.RSR
BuildAppExamplesPart1.exe
BuildAppExamplesPart1.rsr
msvci70.dll

Mac OS X

The Mac OS X keys were set as follows:

```
<RuntimeVLMacFolder>MacHD:4D Runtime Volume License.app</RuntimeVLMacFolder>
```



```
▼ Build_dump
    ▼ Final Application
          ExamplesPart1.app
```

RuntimeVLWinFolder

This key tells 4D the location of the “4D Runtime Volume License” software on a Windows machine for use in building merged single-user applications (both Volume and Light).

The path can be absolute or relative to the structure file.

The path must point to the folder that contains “4DRuntimeVolumeLicense.4DE”.

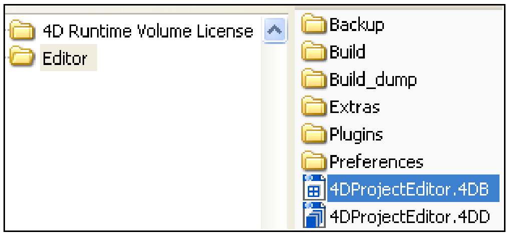
Example

This example uses a relative path.

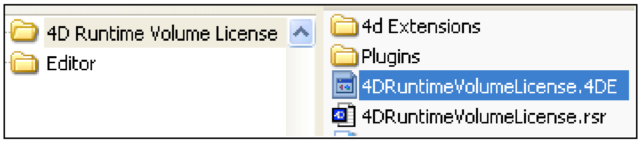
Windows

The RuntimeVLWinFolder was set as follows:

```
<RuntimeVLWinFolder>..\4D Runtime Volume License\</RuntimeVLWinFolder>
```



Structure file location


4D Runtime Volume License
Editor
4d Extensions
Plugins
4DRuntimeVolumeLicense.4DE
4DRuntimeVolumeLicense.rsr

4DRuntimeVolumeLicense.4DE location

Mac OS X

This key has no effect on Mac OS X.

RuntimeVLMacFolder

This key tells 4D the location of the “4D Runtime Volume License” software on a Mac OS X machine for use in building merged single-user applications (both Volume and Light).

The path can be absolute or relative to the structure file.

The path must point to the 4D Runtime Volume License software, e.g. “4D Runtime Volume License.app”.

Example

This example uses a relative path.

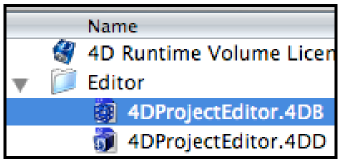
Windows

This key has no effect on Windows.

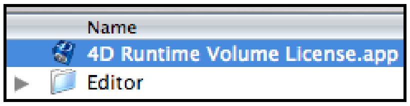
Mac OS X

The RuntimeVLWinFolder was set as follows:

```
<RuntimeVLMacFolder>:..:4D Runtime Volume License.app</RuntimeVLMacFolder>
```



Structure File location



4D Runtime Volume License.app location

RuntimeVLIconWinPath

This key allows the icon of a merged single-user application to be changed for Windows. The path to the icon file should be specified with this key.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, then the icon for “4DRuntimeVolumeLicense.4DE” will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Windows icon file (ends with .ico).*

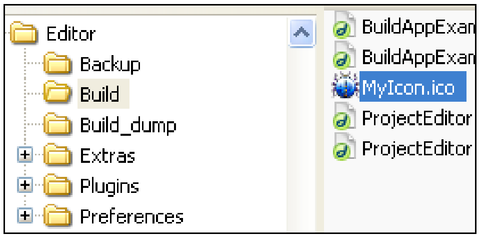
Example

This example uses a relative path.

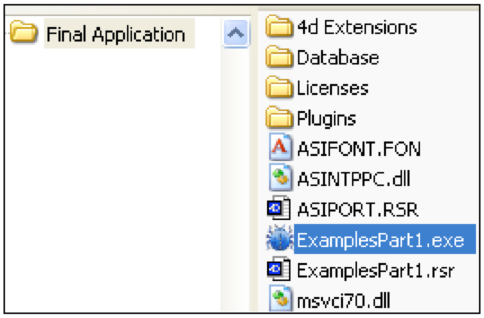
Windows

The RuntimeVLIconWinPath was set as follows:

```
<RuntimeVLIconWinPath>\Build\MyIcon.ico</RuntimeVLIconWinPath>
```



Icon file location



Merged app with icon

Mac OS X

This key has no effect on Mac OS X.

RuntimeVLIconMacPath

This key allows the icon of a merged single-user application to be changed on Mac OS X. The path to the icon file should be specified with this key.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for “4D Runtime Volume License.app” will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Mac OS X icon file (ends with .icns).*

Example

This example uses a relative path.

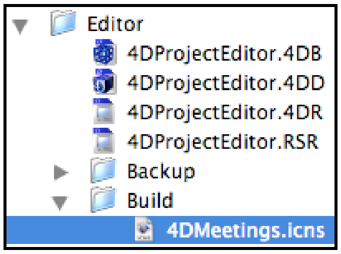
Windows

This key has no effect on Windows.

Mac OS X

The RuntimeVLIconMacPath was set as follows:

```
<RuntimeVLIconMacPath>:Build:4DMeetings.icns</RuntimeVLIconMacPath>
```



```
▼ Editor
      4DProjectEditor.4DB
      4DProjectEditor.4DD
      4DProjectEditor.4DR
      4DProjectEditor.RSR
   ▶ Backup
   ▼ Build
         4DMeetings.icns
```

Icon file location

```
▼ Final Application
      ExamplesPart1.app
```

Merged app with icon

IsOEM

This key tells 4D to integrate OEM deployment licenses into the merged single-user application.

This key can be set to “True” or “False”.

This key has no effect if no OEM licenses are available.

SourcesFiles: CS

The “SourcesFiles” theme is split into two subthemes: “RuntimeVL” and “CS”. This section covers examples of the XML keys from the “CS” subtheme.

ServerIncludeIt

This key tells 4D whether or not to build a merged server application.

This key can be set to “True” or “False”.

This key depends on the BuildServerApplication key. If BuildServerApplication is not set to “True” then the merged application will not be built.

This key also depends on the ServerWinFolder or ServerMacFolder key, on Windows and Mac OS X respectively. If these keys do not contain a valid path to the 4D Server software then the merged application will not be built.

The resulting application is placed in a folder at “Client Server executable/Server”, at the location specified by the key BuildWinDestFolder or BuildMacDestFolder or, if those keys were not used, next to the structure file.

Example

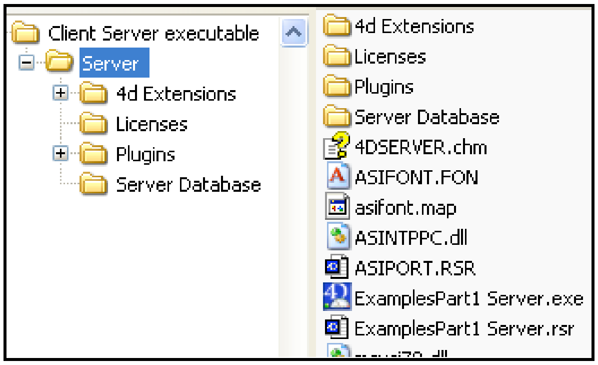
The keys were set as follows:

```
<BuildServerApplication>True</BuildServerApplication>
<ServerIncludeIt>True</ServerIncludeIt>
```

Windows

The Windows keys were set as follows:

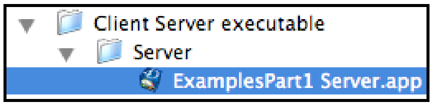
```
<ServerWinFolder>C:\Share\4D\4D 2004.7r3\4D Server 2004.7\</ServerWinFolder>
```



Mac OS X

The Mac OS X keys were set as follows:

```
<ServerMacFolder>MacHD:Applications:4D Server.app</ServerMacFolder>
```



ServerWinFolder

This key tells 4D the location of the “4D Server” software on a Windows machine for use in building merged server applications.

The path can be absolute or relative to the structure file.

The path must point to the folder that contains “4DServer.exe”.

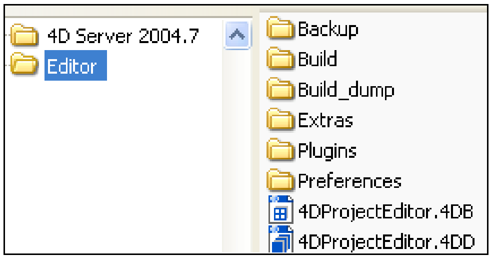
Example

This example uses a relative path.

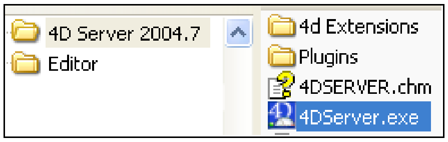
Windows

The ServerWinFolder was set as follows:

```
<ServerWinFolder>..\4D Server 2004.7\</ServerWinFolder>
```



Structure file location



4DServer.exe location

Mac OS X

This key has no effect on Mac OS X.

ServerMacFolder

This key tells 4D the location of the “4D Server” software on a Mac OS X machine for use in building merged Server applications.

The path can be absolute or relative to the structure file.

The path must point to the 4D Server software, e.g. “4D Server.app”.

Example

This example uses a relative path.

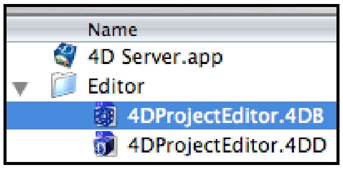
Windows

This key has no effect on Windows.

Mac OS X

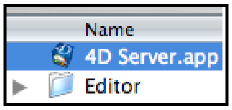
The ServerMacFolder was set as follows:

```
<ServerMacFolder>:..:4D Server.app</ServerMacFolder>
```



Name
4D Server.app
Editor
4DProjectEditor.4DB
4DProjectEditor.4DD

Structure file location



Name
4D Server.app
Editor

4D Server.app location

ClientWinIncludeIt

This key tells 4D whether or not to build a merged client application for Windows. This includes both the **install client** for Windows and the **upgrade client** for Mac OS X.

This key can be set to “True” or “False”.

Note: *You cannot build the **install client** for the opposite platform.*

This key depends on the BuildServerApplication key. If BuildServerApplication is not set to “True” then the merged application will not be built.

This key also depends on the ClientWinFolderToWin or ClientWinFolderToMac key, on Windows or Mac OS X respectively. If these keys do not contain a valid path to the 4D Client software then the merged application will not be built.

Install Client
On Windows, the resulting application is placed in a folder at “Client Server executable\Client”, at the location specified by the key BuildWinDestFolder or, if this key was not used, next to the structure file.

Upgrade Client
On Mac OS X the resulting application is archived and placed in a folder called “Upgrade4DClient” within the merged server application. The upgrade client for Windows is called “archive.win”. Note that the BuildCSUpgradeable key must be set to “True” in order to build the upgrade client.

Note: *If the merged server application fails to build for any reason, the merged client application will not be built.*

Example

The keys were set as follows:

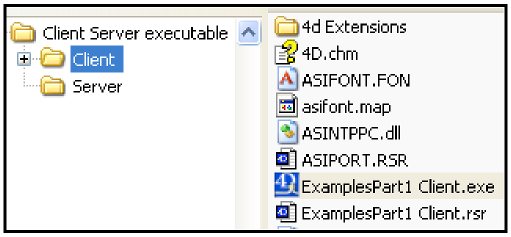
```
<BuildServerApplication>True</BuildServerApplication>
<ClientWinIncludeIt>True</ClientWinIncludeIt>
<BuildCSUpgradeable>True</BuildCSUpgradeable>
```

Windows

In this example the install client was created.

The Windows keys were set as follows:

```
<ClientWinFolderToWin>C:\4D 2004.7r3\4D Client 2004.7\</ClientWinFolderToWin>
```



Client Server executable
Client
Server

4d Extensions
4D.chm
ASIFONT.FON
asifont.map
ASINTPPC.dll
ASIPORT.RSR
ExamplesPart1 Client.exe
ExamplesPart1 Client.rsr

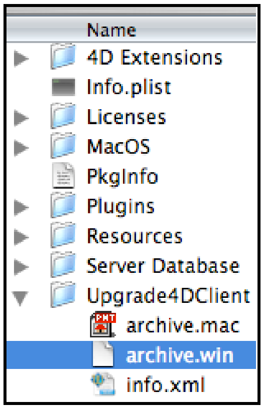
Mac OS X

In this example the upgrade client was created.

The Mac OS X keys were set as follows:

```
<ClientWinFolderToMac>MacHD:Applications:4DClient 2004.7:</ClientWinFolderToMac>
```

Tip: *In a merged server application on Mac OS X, the "archive.win" file is located within the application package. To get inside the package, right-click on the application and choose "Show Package Contents".*



Name
4D Extensions
Info.plist
Licenses
MacOS
PkgInfo
Plugins
Resources
Server Database
Upgrade4DClient
archive.mac
archive.win
info.xml

ClientWinFolderToWin

This key tells 4D the location of the “4D Client” software on a Windows machine for use in building merged client applications.

The path can be absolute or relative to the structure file.

The path must point to the folder that contains “4DClient.exe”.

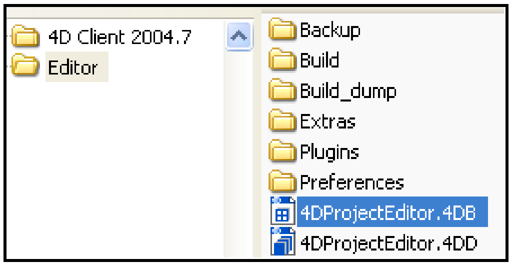
Example

This example uses a relative path.

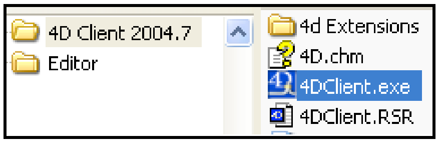
Windows

The ClientWinFolderToWin was set as follows:

```
<ClientWinFolderToWin>..\4D Client 2004.7\</ClientWinFolderToWin>
```



Structure file location



4DClient.exe location

Mac OS X

This key has no effect on Mac OS X.

ClientWinFolderToMac

This key tells 4D the location of the “4D Client” software on a Mac OS X machine for use in building merged client applications. Note that this is the path to the **Windows** software. This key is used to build the **upgrade client**.

The path can be absolute or relative to the structure file.

The path must point to the folder that contains “4DClient.exe”.

Example

This example uses a relative path.

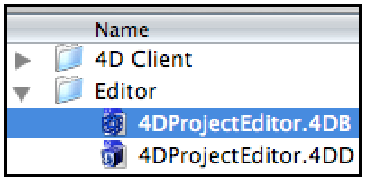
Windows

This key has no effect on Windows.

Mac OS X

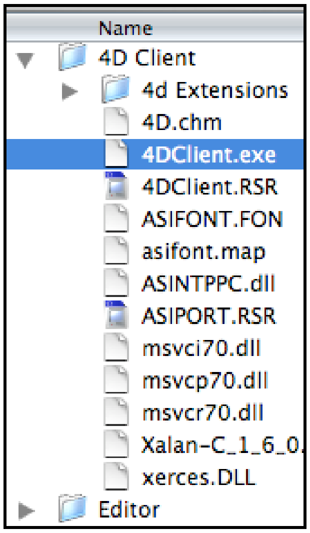
The ClientWinFolderToMac was set as follows:

```
<ClientWinFolderToMac>:..:4D Client:</ClientWinFolderToMac>
```



Name
▶ 4D Client
▼ Editor
4DProjectEditor.4DB
4DProjectEditor.4DD

Structure file location



Name
▼ 4D Client
▶ 4d Extensions
4D.chm
4DClient.exe
4DClient.RSR
ASIFONT.FON
asifont.map
ASINTPPC.dll
ASIPORT.RSR
msvci70.dll
msvcp70.dll
msvcr70.dll
Xalan-C_1_6_0
xerces.DLL
▶ Editor

4DClient.exe location

ClientMacIncludeIt

This key tells 4D whether or not to build a merged client application for Mac OS X. This includes both the **install client** for Mac OS X and the **upgrade client** for Windows.

This key can be set to “True” or “False”.

Note: *You cannot build the* ***install client*** *for the opposite platform.*

This key depends on the BuildServerApplication key. If BuildServerApplication is not set to “True” then the merged application will not be built.

This key also depends on the ClientMacFolderToMac or ClientMacFolderToWin key, on Mac OS X or Windows respectively. If these keys do not contain a valid path to the 4D Client software then the merged application will not be built.

Install Client
On Mac OS X the resulting application is placed in a folder at “Client Server executable:Client”, at the location specified by the key BuildMacDestFolder or, if this key was not used, next to the structure file.

Upgrade Client
On Windows the resulting application is archived and placed in a folder called “Upgrade4DClient” within the merged server application. The upgrade client for Mac OS X is called “archive.mac”. Note that the BuildCSUpgradeable key must be set to “True” in order to build the upgrade client.

Note: *If the merged server application fails to build for any reason, the merged client application will not be built.*

Example

The keys were set as follows:

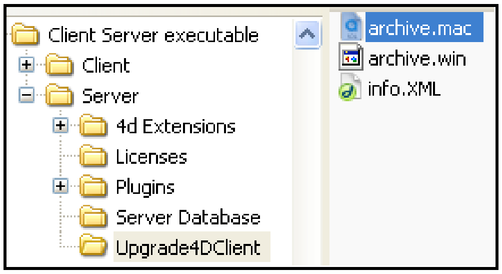
```
<BuildServerApplication>True</BuildServerApplication>
<ClientMacIncludeIt>True</ClientMacIncludeIt>
<BuildCSUpgradeable>True</BuildCSUpgradeable>
```

Windows

In this example, the upgrade client was created.

The Windows keys were set as follows:

```
<ClientMacFolderToWin>C:\4D 2004.7r3\4D Client.app\</ClientMacFolderToWin>
```

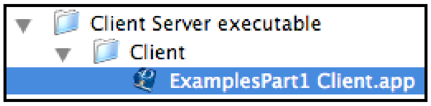


Mac OS X

In this example, the install client was created.

The Mac OS X keys were set as follows:

```
<ClientMacFolderToMac>MacHD:Applications:4D Client.app</ClientMacFolderToMac>
```



ClientMacFolderToWin

This key tells 4D the location of the “4D Client” software on a Windows machine for use in building merged client applications. Note that this is the path to the **Mac OS X** software. This key is used to build the **upgrade client**.

The path can be absolute or relative to the structure file.

The path must point to the 4D Client software for Mac OS X, i.e. the folder named “4D Client.app”.

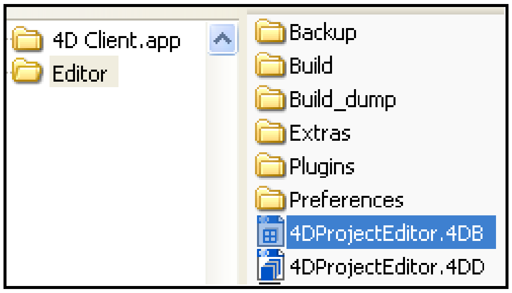
Example

This example uses a relative path.

Windows

The ClientMacFolderToWin was set as follows:

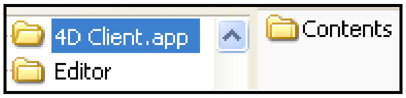
```
<ClientMacFolderToWin>..\4D Client.app\</ClientMacFolderToWin>
```



4D Client.app
Editor

Backup
Build
Build_dump
Extras
Plugins
Preferences
4DProjectEditor.4DB
4DProjectEditor.4DD

Structure file location



4D Client.app
Editor

Contents

4D Client.app location

Mac OS X

This key has no effect on Mac OS X.

ClientMacFolderToMac

This key tells 4D the location of the “4D Client” software on a Mac OS X machine for use in building merged client applications.

The path can be absolute or relative to the structure file.

The path must point to the 4D Client software, i.e. “4D Client.app”.

Example

This example uses a relative path.

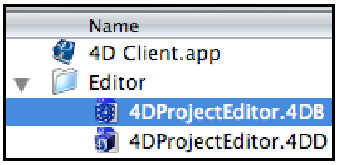
Windows

This key has no effect on Windows.

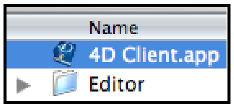
Mac OS X

The ClientMacFolderToMac was set as follows:

```
<ClientMacFolderToMac>:..:4D Client.app</ClientMacFolderToMac>
```



Structure file location



4D Client.app location

ServerIconWinPath

This key allows the icon of a merged server application to be changed on Windows. The path to the icon file should be specified with this key.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for “4DServer.exe” will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Windows icon file (ends with .ico).*

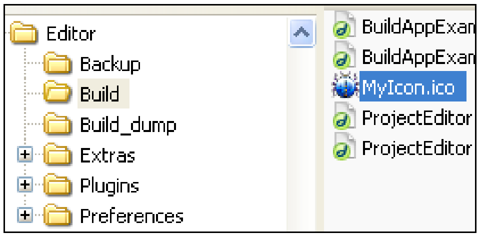
Example

This example uses a relative path.

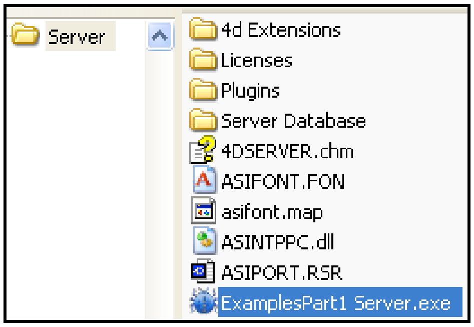
Windows

The ServerIconWinPath was set as follows:

```
<ServerIconWinPath>\Build\MyIcon.ico</ServerIconWinPath>
```



Icon file location



Merged app with icon

Mac OS X

This key has no effect on Mac OS X.

ServerIconMacPath

This key allows the icon of a merged server application to be changed on Mac OS X. The path to the icon file should be specified with this key.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for “4D Server.app” will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Mac OS X icon file (ends with .icns).*

Example

This example uses a relative path.

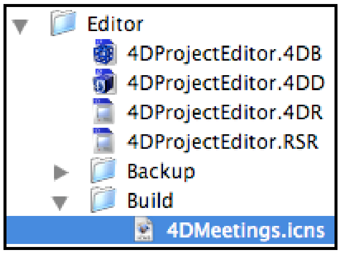
Windows

This key has no effect on Windows.

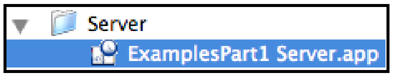
Mac OS X

The ServerIconMacPath was set as follows:

```
<ServerIconMacPath>:Build:4DMeetings.icns</ServerIconMacPath>
```



Icon file location



Merged app with icon

ClientMacIconForMacPath

This key allows the icon of a merged client application to be changed on Mac OS X. The path to the icon file should be specified with this key. Note that this is the icon for the Mac OS X **install client**. The icon for the **upgrade client** must be specified separately.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for "4D Client.app" will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Mac OS X icon file (ends with .icns).*

Example

This example uses a relative path.

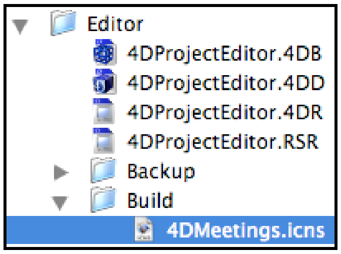
Windows

This key has no effect on Windows.

Mac OS X

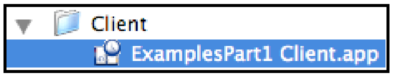
The ClientMacIconForMacPath was set as follows:

```
<ClientMacIconForMacPath>:Build:4DMeetings.icns</ClientMacIconForMacPath>
```



```
▼ Editor
    4DProjectEditor.4DB
    4DProjectEditor.4DD
    4DProjectEditor.4DR
    4DProjectEditor.RSR
  ▶ Backup
  ▼ Build
      4DMeetings.icns
```

Icon file location



```
▼ Client
    ExamplesPart1 Client.app
```

Merged App with Icon

ClientWinIconForMacPath

This key allows the icon of a merged client application to be changed on Mac OS X. The path to the icon file should be specified with this key. Note that this key is for the Windows **upgrade client** on Mac OS X.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for "4DClient.exe" will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Windows icon file (ends with .ico).*

This icon will not be visible after building. It will only be visible when the upgrade client has been installed on the target machine.

Example

This example uses a relative path.

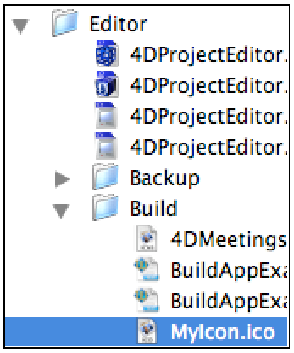
Windows

This key has no effect on Windows.

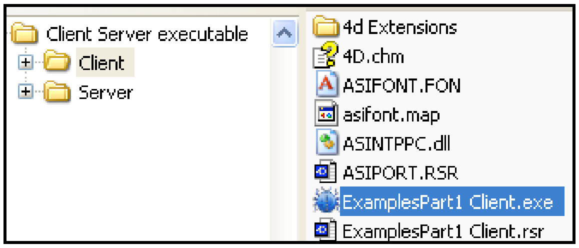
Mac OS X

The ClientWinIconForMacPath was set as follows:

```
<ClientWinIconForMacPath>:Build:MyIcon.ico</ClientWinIconForMacPath>
```



Icon file location



Merged app with icon after upgrade

ClientMacIconForWinPath

This key allows the icon of a merged client application to be changed on Windows. The path to the icon file should be specified with this key. Note that this key is for the Mac OS X **upgrade client** on Windows.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for “4D Client.app” will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Mac OS X icon file (ends with .icns).*

This icon will not be visible after building. It will only be visible when the upgrade client has been installed on the target machine.

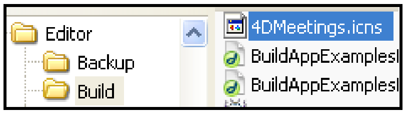
Example

This example uses a relative path.

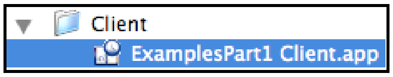
Windows

The ClientMacIconForWinPath was set as follows:

```
<ClientMacIconForWinPath>\Build\4DMeetings.icns</ClientMacIconForWinPath>
```



Icon file location



Merged app with icon after upgrade

Mac OS X

This key has no effect on Mac OS X.

ClientWinIconForWinPath

This key allows the icon of a merged client application to be changed on Windows. The path to the icon file should be specified with this key. Note that this is the icon for the Windows **install client**. The icon for the **upgrade client** must be specified separately.

The path can be absolute or relative to the structure file.

If the path is not valid, the file is of the wrong type, or this key is omitted, the icon for "4DClient.exe" will be used. Note that there is no error message if this is the case. The bad or missing icon is simply ignored.

Note: *The icon file must be a Windows icon file (ends with .ico).*

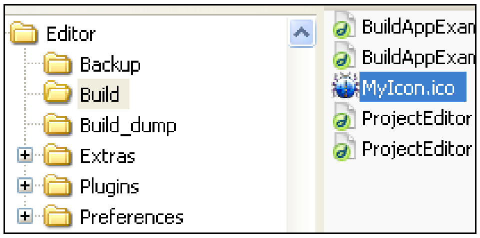
Example

This example uses a relative path.

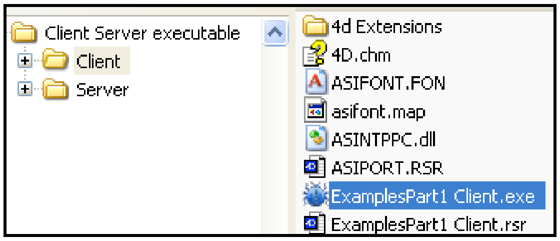
Windows

The ClientWinIconForWinPath was set as follows:

```
<ClientWinIconForWinPath>\Build\MyIcon.ico</ClientWinIconForWinPath>
```



Icon file location



Merged App with Icon

Mac OS X

This key has no effect on Mac OS X.

IsOEM

This key tells 4D to integrate OEM deployment licenses into the merged server application.

This key can be set to “True” or “False”.

This key has no effect if no OEM licenses are available. The server application will simply run in demo mode after building.

Conclusion

This Technical Note provided examples of the XML keys from the CS and SourcesFiles themes. These examples built upon the information presented in the XML keys documentation. Through these examples the 4D developer should have a better understanding of what each XML key from these themes does.

Future editions of this Technical Note series will cover the remaining XML keys.

Related Resources

4D 2004 XML Keys Documentation:
http://www.4d.com/support/documentation.html

BUILD APPLICATION command:
http://www.4d.com/4ddoc2004/CMU/CMU00871.HTM

Tech Note: Building Applications Examples series:

- Building Applications Examples – Part 1 - 4D 2004 Project Editor
- Building Applications Examples – Part 2 – General Parameters and Licenses

http://www.4d.com/support/technotes.html

Tech Note: Building Applications with 4D 2004: Automatic Client Upgrade
http://www.4d.com/support/technotes.html