



Proper Case Conversion & Standardization Component

The Software Company, Inc.
www.SoftwareCompany.com



NetCase for .NET is a data quality component that converts existing data formats into proper cased formats that are more attractive and easier to read. NetCase applies proper case to your data through the use of special algorithms and Style tables that you've defined. NetCase can easily translate, punctuate, abbreviate, unabbreviate as well as correct common misspellings before they enter your system.

With NetCase you can process all types of data including Names, Addresses, City/State/Zip, Sentences and Titles, each with its own capitalization rules. You control capitalization through the use of 8 user-definable Styles. The use of Style tables allows you to apply different capitalization rules to the same word depending on its context.

To use NetCase, set the Text_Style to the desired style then set the Text_In property to the text to be adjusted. NetCase comes with 8 predefined Styles each of which is customizable by you via an external table. The same word can now have different capitalization rules applied to it depending on the Style you've chosen.

NetCase does more than just proper case conversions. Based on your specifications, it can punctuate, abbreviate and even expand existing abbreviations. Instantly correct typos using the built-in list of "Common Misspellings".

Benefits

- Save Keying Time - Automatically Convert Symbols and Translate Abbreviations into Their Correctly Capitalized Full Spelling
- Instantly Correct Typos - Built-In List of "Common Misspellings"
- User Updatable Tables So Your Applications Never Go Out-of-Date
- Over 1,000 Proper-cased and Punctuated Irish Surnames Included
- FREE Upgrades for a Full Year

Features

- Apply Different Capitalization Rules to the Same Word Depending on its Context: "Co" Is Company and "CO" Is Colorado
- Easily Add Punctuation Such as: "abc inc" -to- "ABC, Inc."
- Royalty-Free Runtime
- Designed for Use with All .NET Compatible Programming Languages

NetCase for .NET utilizes 8 unique capitalization Styles. Each customizable Style has its own unique capitalization rules. NetCase starts by applying the default capitalization rules for the selected style. Next, every word / phrase from the input string is compared to the NetCase style table for that particular style. When a match is found, a replacement is made using the corresponding word / phrase from the table overriding any default capitalization rules. The result is a precise transformation to proper case no matter what type of input string.

Examples

[NameStyle] – Standardize Prefixes & Suffixes

Text_In: MR / MRS JOHN VAN DER CAMP PHD

Text_Out: Mr. and Mrs. John Van der Camp, Ph. D.

[AddressStyle] – Eliminate Conflict Between Street Suffix & City / State / Zip

Text_In: 1234 SE 41ST CT APT 26E (CT is also the abbreviation for Connecticut)

Text_Out: 1234 SE 41st Ct, Apt. 26E

[CityStateZipStyle] – Avoid Confusion Between “Company” & “Colorado”

Text_In: PINE, CO 80470

Text_Out: Pine, CO 80470

[TitleStyle] – Capitalize the First Letter of Each Word, Convert Symbols

Text_In: (c) THE SOFTWARE COMPANY, INCORPORATED

Text_Out: © The Software Company, Inc.

[SentenceStyle] – Capitalize the First Letter of Each Sentence

Text_In: ACTIVECASE(tm) IS A TRADEMARK OF THE SOFTWARE COMPANY.

Text_Out: ActiveCase™ is a trademark of The Software Company.

Text_In

Syntax: Text_In = String
Text2_In = String (optional)
Text3_In = String (optional)
Text4_In = String (optional)

Description:

Set these properties to the text strings to be processed.

When the “Convert” Method is invoked, each Text_In string is capitalized according to the corresponding Text_Style and placed into the appropriate Text_Out property.

Text_Style

Syntax: Text_Style = “StringLiteral”
Text2_Style = “StringLiteral” (optional)
Text3_Style = “StringLiteral” (optional)
Text4_Style = “StringLiteral” (optional)

Description:

Set these properties to either “Name”, “Address”, “CSZ”, “Title”, “Sentence”, “User”, “Uppercase” or “Lowercase” to indicate which Style table to use for each corresponding Text_In string. **Default is “Uppercase”.**

Default Capitalization Rules

Name, Address, CSZ, Title:

- Capitalize first letter of each word
- Capitalize each side of quote or “&” (De'Angelo) except plural possessive (John's)
- Lower case Ordinal numbers (1st, 2nd, etc.)

Sentence:

- Capitalize first letter of each sentence

User:

- Make word / phrase replacements from table - nothing else

In every case, the external table NetCase.ref that you've defined for each Style will override the default capitalization rules. See “Updating User Control Tables” later in this guide for instructions on customizing this file.

Convert_Irish_Surnames (new in v4.0+)

Syntax: Convert_Irish_Surnames = Boolean
Convert_Irish_Surnames2 = Boolean (optional)
Convert_Irish_Surnames3 = Boolean (optional)
Convert_Irish_Surnames4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to convert Irish surnames for each corresponding Text_In string. (“OHARA” converts to “O’Hara”, etc.) **Default is “False”**.

Typographic_Apostrophe (new in v4.0+)

Syntax: Typographic_Apostrophe = Boolean
Typographic_Apostrophe2 = Boolean (optional)
Typographic_Apostrophe3 = Boolean (optional)
Typographic_Apostrophe4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to use the typographic apostrophe (’) in place of the standard apostrophe (') when converting Irish surnames. These properties are only valid when “Convert_Irish_Surnames” is set to “True”. **Default is “False”**.

Correct_Common_Misspell (new in v4.0+)

Syntax: Correct_Common_Misspell = Boolean
Correct_Common_Misspell2 = Boolean (optional)
Correct_Common_Misspell3 = Boolean (optional)
Correct_Common_Misspell4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to correct common misspellings for each corresponding Text_In string. (“acn” is corrected to “can”, etc.) **Default is “False”**.

Alphanumeric_Upper (new in v4.0+)

Syntax: Alphanumeric_Upper = Boolean
 Alphanumeric_Upper2 = Boolean (optional)
 Alphanumeric_Upper3 = Boolean (optional)
 Alphanumeric_Upper4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to convert alphanumeric data to all uppercase for each corresponding Text_In string. (“3d” converts to “3D”, etc.) **Default is “False”.**

No_Vowels_Upper (new in v4.0+)

Syntax: No_Vowels_Upper = Boolean
 No_Vowels_Upper2 = Boolean (optional)
 No_Vowels_Upper3 = Boolean (optional)
 No_Vowels_Upper4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to convert alphabetic data containing no vowels to all uppercase for each corresponding Text_In string. (“mlk” converts to “MLK”, etc.) **Default is “False”.**

Ignore_Mixed_Case (new in v4.0+)

Syntax: Ignore_Mixed_Case = Boolean
 Ignore_Mixed_Case2 = Boolean (optional)
 Ignore_Mixed_Case3 = Boolean (optional)
 Ignore_Mixed_Case4 = Boolean (optional)

Description:

Set these properties to Boolean (True/False) to indicate whether or not to convert mixed-case text for each corresponding Text_In string. Use this setting to preserve existing mixed-case text. Text will only be converted when Text_In string is all uppercase or lowercase. **Default is “False”.**

Reference_File_Path (changed in v3.0+)

Syntax: Reference_File_Path = String

Description:

Set this property to the full path and file name of the user-defined file containing the Style tables. Default tables are supplied and installed in the NetCase installation folder under the name: "NetCase.ref". You can rename and relocate this file to any other folder as long as you set the Reference_File_Path property to the full path and file name. **Defaults are the NetCase installation folder and then the folder of the invoking application.**

(AppDomain.CurrentDomain.BaseDirectory)

See "Updating User Control Tables" later in this guide for instructions on customizing this file.

Static_Key_Name (new in v3.0+)

Syntax: Static_Key_Name = String

Description:

Set this property to the name portion of the static key assignment or blank.

Static_Key (new in v3.0+)

Syntax: Static_Key = String

Description:

Set this property to the key portion of the static key assignment or blank.

Text_Out (read only)

Syntax: String = Text_Out
 String = Text2_Out (optional)
 String = Text3_Out (optional)
 String = Text4_Out (optional)

Description:

When the “Convert” method is invoked, these properties are set to the converted text string of Text_In. Values will be uppercase, lowercase or mixed-case text depending on the Text_Style selected.

Return_Code (read only)(new in v3.0+)

Syntax: String = Return_Code

Description:

When the “Convert” method is invoked, this property is set to blank upon successful completion. Some common exceptions are listed below. Most exceptions usually occur on the first call to the “Convert” method. *This property should be examined on each return from NetCase.*

Common Return Codes:

- | | |
|------------------|---|
| R35 | Reference File Not Found (see Reference_File_Path property) |
| T00 | Case Table Limit Reached (16384) |
| L00 | Evaluation Period Expired |
| L01 | Static Key Validation Failed (see Static_Key property) |
| L50 - L69 | License Validation Failed |

Clear

Syntax: NetCase.Clear

Description:

When this method is invoked, all properties are set to null with the exception of Static_Key, Static_Key_Name and Reference_File_Path.

Convert

Syntax: NetCase.Convert

Description:

When this method is invoked, each Text_In property is inspected and the capitalization is converted according to the corresponding Text_Style. The transformed data is then placed into the corresponding Text_Out property. The Return_Code property is also set and should be checked after each invocation of the Convert method. See *“Return_Code”* property.

Updating User Control Tables

NetCase.ref is located in the NetCase installation folder. Use Notepad or a similar text editor to edit the file. Detailed instructions on the format of the entries are contained within the file. This file can also be relocated. *See Reference_File_Path property.*

NetCase does more than just case conversions. Based on your specifications, it can punctuate, abbreviate and even expand existing abbreviations. Turn “Bill” into “William” and “1st” into “First”.

The use of the wildcard character (*) allows you to capitalize groups of words depending on their prefix. If you need to exclude certain words from the wildcarding, place them in the table before the wildcarded word.

Extensive tables are included. Below are a few examples:

NetCase Style Table entries . . .

DIGENOVA	replaced by	diGenova
DIGITAL		digital
DIG*		DiG*
VAN DER C*		Van der C*
TSC		The Software Company, Inc.
INC		<, Inc.

After NetCase . . .

DIGENOVA	adjusted to	diGenova
DIGITAL		digital
DIGIORNO		DiGiorno
VAN DER CAMP		Van der Camp
TSC		The Software Company
ABC INC		ABC, Inc.

Every word that begins with “DIG” is capitalized as was “DiGiorno” with the exception of “DIGENOVA” and “DIGITAL” because they appeared in the table before the wildcard “DIG*”. The “<” causes left justification of the replacement.

If Installation Doesn't Start Automatically:

- Select **Start > Run** from the Task Bar.
- Type CD-ROM drive letter followed by “:\NetCase40.msi” and press enter.

In the folder “Program Files\The Software Company\NetCase 4.0” you will find a sample program named: VBSample.vbproj. There is also a compiled version called VBSample.exe that you can run to demonstrate NetCase.

Deploying Your Applications

Be sure to include the following in your deployment package:

NetCase.dll – usually placed in the application folder
NetCase.ref* – usually placed in the application folder

* NetCase.ref can be relocated anywhere on the target machine as long as the full path and file name are specified in the Reference_File_Path property.

In addition to the above, there is a small runtime package that is installed into the Global Assembly Cache (GAC) of the target machine. There are two ways to do this:

Include the **MergeModuleRuntimeNET20.msm** with your installation package

- OR -

Execute **StandaloneRuntimeNET20.exe** for each new installation

There's an explanation of each at the following link:

http://www.SoftwareCompany.com/dotnet/dotnet_runtime.htm

This product is initially licensed for a period of 30 days or up to 1000 calls. It must be registered to continue using it after this evaluation period. Please contact us at:

Sales@SoftwareCompany.com

303/838-1223 (voice)

303/838-1224 (fax)