BisonWare Parser Control V5.0

The BisonWare file parser VCL component adds powerful parsing capabilities to your Delphi applications.

Both ASCII text files and text strings held in memory may be parsed using the BisonWare parser.

To register this component go **SWREG #8054** on Compuserve.

Registration is only **$10**.   You will receive the registered component by EMAIL.

Please address all queries to Compuserve ID **100416,3553**.

BisonWare Parser Control V5.0 - Custom Properties

BufferSize	Defines file input buffer size
CaseConvert	Allows case conversion of the parsed fields
CommentDelimiters	Defines characters which delimit comment text
ErrorCode	The code of the last error which occurred
FieldBreaks	Allows forced field breaks at pre-defined characters
FieldCount	Number of fields in the current record
FieldDelimiters	Defines the characters to be used as field separators
FieldLength	The length of the last fields parsed
FileName	Defines the name of the file to be parsed
IgnoreLeadingDelimiters	Controls processing of leading delimiters
IgnoreLineFeeds	Controls processing of line-feed characters
IgnoreMultiDelimiters	Controls processing of multiple delimiters
IgnoreTrailingDelimiters	Controls processing of trailing delimiters
ParseField	Last field to be parsed
ParseList	List of fields contained within a record
ParsePChar	Defines a pointer to a text string in memory
PercentComplete	Defines a pointer to a text string in memory
RecordCount	Defines the number of records parsed so far
RecordDelimiters	Defines the characters to be used as record delimiters
SpaceOutTabs	Controls processing of tab characters within the text
TextQualifiers	Defines the characters which delimit text strings
TrimLeadingSpaces	Controls processing of leading spaces in parsed fields
TrimMultipleSpaces	Controls processing of multiple spaces in parsed fields
TrimTextQualifiers	Controls processing of text qualifiers in parsed fields
TrimTrailingSpaces	Controls processing of trailing spaces in parsed fields

BufferSize Property

The BufferSize property denotes how much heap memory is set aside for use as a file buffer during text file parsing operations.

The property is provided to allow the programmer to have tight control of heap memory used during a parse process.

The minimum buffer size is **1024** bytes (1K).
The maximum buffer size it **32767** bytes (32K).
The default buffer size is **4096** bytes (4K).

A large buffer size increases parsing performance on larger text files.

The BufferSize property is ignored for memory parsing operations.

Comment Delimiters Property

The CommentDelimiters property allows you to define strings of characters which delimit comment text in the incoming file.   Any text which is found between these characters is ignored during the parse process.

The starting and ending delimiters do not have to be the same.

The starting and ending delimiters can be up to two characters in length.

More than one pair of start and end delimiters can be specified.

The comment delimiter string should be formatted as follows :

start delimiter1<space>end delimiter1**;**start delimiter2<space>end delimiter2**...**

e.g.
{ };/* */;(* *)

The field can be left blank if required.

FieldBreaks Property

The FieldBreaks property is similar to the <u>FieldDelimiters</u> property except that the character used is not lost from the output fields.   Indeed the field break character is actually returned as a field in itself.

This type of processing is valuable for breaking down complex mathematical or program statements.

for example,

given that an input line contains :

X=(Y*Z)/3

and that FieldBreaks is set to :

=()/*

then the returned fields would be :

1. **X**
2. **=**
3. **(**
4. **Y**
5. *****
6. **Z**
7. **)**
8. **/**
9. **3**

CaseConvert Property

The CaseConvert property allows the conversion of parsed fields to Upper or Lower case.

This property is useful when the case of the input file or memory string is unknown yet the output text is required in upper or lower case.

The property is an enumerated type which has the following values :

ctNone   No case conversion takes place
ctUpper   Parsed fields are converted to upper case
ctLower   Parsed fields are converted to lower case

The default value is **ctNone**

FieldDelimiters Property

The FieldDelimiters property holds a string of characters which denote the characters which separate individual fields within the input text file or memory string.

e.g.

String	Field Delimiters	Parsed Fields
12345,Ann Other,Finance;789.10	,	12345 Ann Other Finance;789.10
12345,Ann Other,Finance;789.10	,;	12345 Ann Other Finance 789.10
12345,Ann Other,Finance;789.10	\<space\>	12345,Ann Other,Finance;789.10

A maximum of 255 field delimiters may be specified.   There are no default field delimiters.   It is not necessary to have a field delimiter.

FileName Property

The FileName property holds the file name of the file to parse in standard DOS format :

[d:][path][filename][.fileext]

e.g.

c:\delphi\employee.txt

This field is ignored for Memory Parse operations.

IgnoreLeadingDelimiters Property

The IgnoreLeadingDelimiters property is a Boolean field which when set to true tells the parser to ignore any initial delimiters which would otherwise return null fields.

e.g. given input record :

,,,Ann Other,Finance,0341

if IgnoreLeadingDelimiters is set to false then 6 fields are returned as follows :

1.   (null)
2.   (null)
3.   (null)
4.   Ann Other
5.   Finance
6.   0341

if IgnoreLeadingDelimiters is set to true then 3 fields are returned as follows :

1.   Ann Other
2.   Finance
3.   0341

IgnoreMultiDelimiters Property

The IgnoreMultiDelimiters property is a Boolean field which when set to true tells the parser to ignore any duplicate delimiters which would otherwise return null fields.

e.g. given input record :

Ann Other,,,Finance,0341

if IgnoreMultiDelimiters is set to false then 5 fields are returned as follows :

1.   Ann Other
2.   (null)
3.   (null)
4.   Finance
5.   0341

if IgnoreMultiDelimiters is set to true then 3 fields are returned as follows :

1.   Ann Other
2.   Finance
3.   0341

IgnoreTrailingDelimiters Property

The IgnoreTrailingDelimiters property is a Boolean field which when set to true tells the parser to ignore any trailing delimiters which would otherwise return null fields.

e.g. given input record :

Ann Other,Finance,0341,,,

if IgnoreTrailingDelimiters is set to false then 6 fields are returned as follows :

1.   Ann Other
2.   Finance
3.   0341
4.   (null)
5.   (null)
6.   (null)

if IgnoreTrailingDelimiters is set to true then 3 fields are returned as follows :

1.   Ann Other
2.   Finance
3.   0341

IgnoreLineFeeds Property

during a parse operation the line feeds character is normally taken as the end of each input record.   i.e. The <u>OnRecordParse</u> event is fired after which the <u>field list</u> is cleared out and the next record is started.

For some file types a single record may be spread across multiple lines with the record terminated by another character such as the semi-colon (;).

To allow records to occupy multiple lines set the IgnoreLineFeeds character to True and set the <u>RecordDelimiters</u> property to an appropriate value.

ParsePChar Property

The ParsePChar property is used for Memory Parsing operations.   At run time set this property to a PChar variable which points to the string of characters to be parsed.

The parse operation can then be started by executing the ParseMemory method.

RecordDelimiters Property

The RecordDelimiters property is a string which should be set to the list of characters which delimit records in a file or memory parse operation.

e.g. setting the RecordDelimiters property to :

;:!

Tells the parser to fire a OnRecordParse event each time one of those characters is encountered in the input.

Only use this property if records are not delimited by Carriage Return/Line Feed characters.

SpaceOutTabs Property

The SpaceOutTabs property is a Boolean property.   When set to true any TAB characters which are encountered in the input are converted to space characters.

TextQualifiers Property

The TextQualifiers property defines a set of characters which delimit text strings within the parser input.

Any other delimiters which occur with text qualifiers are treated as part of the text.

More than one TextQualifier can be defined, however text qualifiers are only matched in exact pairs.

e.g.

Given the input text :

Ann Other,California,Finance,1034,Customer,Number 123

and TextQualifiers set to

Then 4 fields would be returned :

1. **Ann Other,California**
2. **Finance**
3. **1034**
4. **Customer,Number 123**

TrimTextQualifiers Property

The TrimTextQualifiers property is a Boolean property which when set to true causes the delimiters to be removed from any delimited text fields returned by the parser

e.g.

Given the input text :

Ann Other,Finance,1234

and the TextQualifers property set to :

Then if TrimTextQualifiers is set to False then these 3 fields are returned :

1. **Ann Other**
2. **Finance**
3. **1234**

if TrimTextQualifiers is set to true then these 3 fields are returned :

1. **Ann Other**
2. **Finance**
3. **1234**

TrimLeadingSpaces

The TrimLeadingSpaces property is a Boolean property which when set to True tells the parser to trim leading spaces from any returned fields.

e.g.

Given the input text :

Ann Other,   Finance Department,   1234

and TrimLeadingSpaces is set to False then the following 3 fields are returned :

1.   **Ann Other**
2.   **Finance Department**
3.   **1234**

if TrimLeadingSpaces is set to True then the following 3 fields are returned :

1.   **Ann Other**
2.   **Finance Department**
3.   **1234**

TrimMultipleSpaces Property

The TrimMultipleSpaces property is a Boolean property which when set to True tells the parser to trim multiple spaces from any returned fields.

e.g.

Given the input text :

Ann Other,Finance   Department,1234

and TrimMultipleSpaces is set to False then the following 3 fields are returned :

1. **Ann Other**
2. **Finance   Department**
3. **1234**

if TrimMultipleSpaces is set to True then the following 3 fields are returned :

1. **Ann Other**
2. **Finance Department**
3. **1234**

TrimTrailingSpaces Property

The TrimTrailingSpaces property is a Boolean property which when set to True tells the parser to trim trailing spaces from any returned fields.

e.g.

Given the input text :

Ann Other,Finance Department   ,1234

and TrimTrailingSpaces is set to False then the following 3 fields are returned :

1.   **Ann Other**
2.   **Finance Department |** (28 characters)
3.   **1234**

if TrimTrailingSpaces is set to True then the following 3 fields are returned :

1.   **Ann Other**
2.   **Finance Department**|(18 characters)
3.   **1234**

BisonWare Parser Control V5.0 - Custom Methods

AbortParse	Aborts the parse operation at the next record
ParseFile	Begins a file parse operation
ParseMemory	Begins a memory parse operation

AbortParse Method

Executing the AbortParse method during a parse operation causes the operation to be halted at the end of the next record.

ParseFile Method

Executing the ParseFile method begins a parse on the file identified by the FileName parameter.

The parse operation can be stopped by executing the AbortParse method.

ParseMemory Method

Executing the ParseMemory method begins a parse on the memory block identified by the <u>ParsePChar</u> parameter.

The parse operation can be stopped by executing the <u>AbortParse</u> method.

BisonWare File Parser V5.0 - Custom Events

OnFieldParse   Occurs after each field is parsed
OnParseBegin   Occurs at the start of the parse operation
OnParseEnd   Occurs at the end of the parse operation
OnParseError   Occurs when an internal error occurs
OnRecordParse   Occurs after each record is parsed

OnFieldParse Event

The OnFieldParse event occurs each time that a complete field is parsed from the input.

The field parsed in contained in the ParseField property.

The length of the field is in the FieldLength property.

The number of fields so far in the current record is in the FieldCount property.

A list of all the fields in the current record is in the ParseList property.

OnParseBegin Event

The OnParseBegin event occurs once at the start of the parse operation.   This allows the programmer to perform any initialization routines that are required.

OnParseEnd Event

The OnParseEnd event occurs once at the end of the parse operation.   This allows the programmer to perform any special termination routines that are required.

OnParseError Event

The OnParseError event occurs each time that an error occurs during the parse operation.

The error code is stored within the ErrorCode property.

OnRecordParse Event

The OnRecordParse event occurs each time that a complete record is parsed from the input.

The number of fields in the current record is contained within the FieldCount property.

A list of all the fields in the current record is contained within the ParseList property.

ErrorCode Property

The ErrorCode property contains an integer value denoting the last error that occurred during the parse operation.

The list of errors codes is as follows :

0.   No Error
1.   File Not Found
2.   Field Overflow
3.   Unterminated Text String
4.   Unterminated Comment

The error code should be read and processed in the <u>OnParseError</u> event handler

This property is read only at run time.

FieldLength Property

The FieldLength property is an integer value which contains the length of the last field to be parsed.

The field itself is stored within the ParseField property.

This property is read only at run time.

FieldCount Property

The FieldCount property is an integer value which contains the total number of fields contained within a parsed records.

If the property is read during the <u>OnFieldParse</u> event then it contains the total number of fields so far.

This property is read only at run time.

ParseField Property

The ParseField property contains the last field to be parsed.

This property should be used with the <u>OnFieldParse</u> event in order to process a file field by field.

This property is read only at run time.

ParseList Property

The ParseList property is of type TStringList.

It contains a list of individual fields contained within a record.

PercentComplete property

The PercentComplete property is an integer value which contains a number between 0 and 100 which indicates the percentage of the input file or memory block which has been processed.

This value can be used to update a gauge control to show the progress of the parse operation.

This value is read only at run time.

RecordCount Property

The RecordCount property is an integer value which contains the number of records which have been processed from the input file or memory block.

This property is read only at run time.