

Release Notes

Version 4.3.1

October 1999

CONSTRUCT SPECTRUM™ AND SDK

Manual Order Number: SPE431-008IBW

Copyright © SAGA SOFTWARE, Inc., 1999. All rights reserved.

SAGA, SAGA SOFTWARE, the SAGA logo, Free Your Information, the FYI logo, CRIS, Construct, Construct Spectrum, Construct Spectrum SDK, iXpress, Sagacertify, Sagagallery, Sagavista, and Your Fastest Route to Enterprise Integration are trademarks or registered trademarks of SAGA SOFTWARE, Inc. in the U.S. and/or other countries. Adabas, Adabas Delta Save Facility, Adabas Fastpath, Adabas SQL Server, Adabas Vista, Adaplex+, Bolero, Com-plete, Entire, Entire Access, Entire Net-work, EntireX, EntireX DCOM, Entire Broker, Entire Broker SDK, Entire Broker APPC, Entire SAF Gateway, Natural, Natural Architecture, Natural Elite, Natural New Dimension, Natural Lightstorm, Natural Vision, New Dimension, PAC, Predict, Software AG, and Super Natural are developed by Software AG of Darmstadt, Germany and are distributed in the U.S., Latin America, Canada, Israel and Japan exclusively through SAGA SOFTWARE, Inc. and its subsidiaries and distributors. Adabas and Natural are registered trademarks of Software AG of Darmstadt, Germany. Except for Adabas and Natural, these products developed by Software AG of Darmstadt, Germany are either registered trademarks or trademarks of SAGA SOFTWARE, Inc., in the U.S. and/or other countries.

Other company or product names mentioned are used for informational purposes only and may be trademarks or servicemarks of their respective owners.

TABLE OF CONTENTS

Release Notes	6
Support for Natural 3.1	6
Visual Basic 6.0	6
Software Prerequisites	6
Filenames	6
Migrating Older Spectrum Applications to this Version	8
Documentation Changes	8
Drop Support Dates	9
New Features	10
Spectrum Dispatch Client Bug Fix	10
Web-Enabled Applications	10
Class Changes	10
Application Class	11
DataAreaDefinition Class	11
NaturalDataArea Class	12
Dispatcher Class	13
DispatcherServices Class	14
DispatcherProperties Class	14
NaturalFieldDef Class	15

CONSTRUCT SPECTRUM AND SDK V4.3.1 RELEASE NOTES

This document contains information not provided in previous Construct Spectrum and Construct Spectrum SDK documentation. It describes the new features, support requirements, and changes in this release of Construct Spectrum and Construct Spectrum SDK.

The following topics are covered:

- **Release Notes**, page 6
- **New Features**, page 10

Release Notes

This section describes the software and support required to use Construct Spectrum and Construct Spectrum SDK. It also explains the file name changes, dates for dropping support for previous versions, and documentation changes pertaining to this release.

Support for Natural 3.1

This version of Construct Spectrum and Construct Spectrum SDK is compatible with Natural 3.1.

Visual Basic 6.0

All Construct Spectrum client components (.exe and .dlls) have been compiled with Visual Basic 6.0, and need the Visual Basic 6.0 run-time DLLs.

Software Prerequisites

This release of Construct Spectrum and Construct Spectrum SDK only support Visual Basic 6.0. To develop or run Spectrum applications with Visual Basic 5.0, you must install the previous release of Construct Spectrum and the SDK. To develop and run applications with Visual Basic 6.0, you must install this release of Construct Spectrum and Construct Spectrum SDK.

Note: You can install both versions of Spectrum and the SDK on the same PC.

Filenames

The default installation folder in your Program Files folder has changed from CONSTRUCT Spectrum to Construct Spectrum 4.3. The following files have also been renamed:

Construct Spectrum 4.2.1 Filename	Construct Spectrum 4.3.1 Filename
CSTVBF5.dll	CSTVBF6.dll
CSTAddIn5.dll	CSTAddIn6.dll
FrameWrk5 folder	FrameWrk6 folder
Demos5 folder	Demos6 folder

These filename changes indicate that this version of Construct Spectrum only supports Visual Basic 6.0. If you access the **References** dialog in Visual Basic to mark the ActiveX components you will be using in your application, you will also see different names for the following components:

Construct Spectrum 4.2.1 Name	Construct Spectrum 4.3.1 Name
Construct Spectrum Add-In (VB5)	Construct Spectrum Visual Basic Add-In 6.0
Construct Spectrum BDT Functionality	Construct Spectrum BDT Functionality 6.0
Construct Spectrum Framework Classes (VB5)	Construct Spectrum Framework Classes 6.0
Construct Spectrum ETB Queue Manager	Construct Spectrum ETB Queue Manager 6.0
Construct Spectrum Dispatch Client (VB5)	Construct Spectrum Dispatch Client 6.0

Migrating Older Spectrum Applications to this Version

If you want to convert existing Construct Spectrum projects created with Visual Basic 4.0 or 5.0 to Visual Basic 6.0, you must:

- Create a new project using the Visual Basic 6.0 Add-In
- Copy the hand-coded and generated modules from your old application(s) into the new project.

Documentation Changes

The following documentation changes have been made to the Construct Spectrum and Construct Spectrum documentation suites.

- Look for the DocMenu.pdf file on the Documentation CD supplied with Natural Construct, Construct Spectrum, and Construct Spectrum SDK. This PDF provides descriptions of the manuals, links to them, and tips for using PDFs.
- The *Construct Spectrum Programmer's Guide* now contains information common to client/server and web applications.
- Developing Client/Server Applications was added.
- Developing Web Applications was added.
- *Construct Spectrum Messages* describes the error messages you may receive when developing and running Construct Spectrum applications.
- The *Construct Spectrum Client Installation Guide* contains client installation instructions for both Construct Spectrum and Construct Spectrum SDK.
- The *Construct Spectrum Mainframe Installation Guide* contains mainframe installation procedures for Construct Spectrum and Construct Spectrum SDK.

Drop Support Dates

SAGA SOFTWARE Inc. provides support for the following Construct Spectrum and Construct Spectrum SDK releases and SMs:

Natural Construct Release	Date Released	End-of-Support Date
Version 4.2.1	December 1998	June 2000
Version 4.1.2	January 1998	June 2000
Version 4.1.1	July 1997	June 2000

Support for a version level will be provided through the month specified in the End-of-Support Date column.

New Features

This section describes the new features, bug fixes, and class changes in this release.

Spectrum Dispatch Client Bug Fix

The memory leak that occurred when the Application.Initialize method was called repeatedly is fixed.

Web-Enabled Applications

Construct Spectrum SDK now generates feature-rich web applications that allow you to use either Microsoft Internet Explorer (IE) or Netscape Navigator to access Natural business applications. To facilitate the rapid generation of web applications, SDK now includes:

- A new web application framework containing the base functionality common to all Spectrum web applications.
- Two new project wizards that create the Visual Basic projects to which you will add generated code.
- New code generation wizards that rapidly generate Visual Basic classes, modules, and HTML templates entirely on the client. You no longer need to use the Construct generation environment on the server to generate these components in Natural, and then download them to the client.
- A new Visual Basic 6.0 add-in that places a Spectrum menu on the Visual Basic menu bar. The add-in invokes the new code generation wizards and provides other functionality that make it easy to generate web application components within the Visual Basic development environment.

Class Changes

The following Construct Spectrum classes have changed:

Application Class

The Allocate2 method has been removed. You can now use the Allocate method to allocate Natural data areas defined in an external LIF file or in the code by passing the actual data area definition as a multi-line string. For example:

```
Dim sdc As New SDCLib6.Application
Dim nda As NaturalDataArea

' Allocate a data area defined in an external LIF file.
sdc.Initialize App.Path, "SPECDEMO"
Set nda = sdc.Allocate("CUSTMSA")

' Allocate a data area defined in the code.
Set nda = sdc.Allocate("01 CUSTMSA" & vbCrLf & _
    " 02 CUSTOMER-NUMBER (N5)" & vbCrLf & _
    " 02 BUSINESS-NAME (A30)" & vbCrLf & _
    "...
    " 02 CUSTOMER-TIMESTAMP (T)" & vbCrLf)
```

The following changes have also been made to this class:

- The CreateDispatcher2 method has been removed. This method had an optional parameter to allow you to specify which dispatch service definition to use. This parameter has been added to the CreateDispatcher method.
- A new method called ParseDataArea has been added. This method has the same parameters as the Allocate method, but returns a DataAreaDefinition object.

DataAreaDefinition Class

The DataAreaDefinition class is similar to the NaturalDataArea class because it stores the definition of a Natural data area and provides properties to return information about the fields in the Natural data area. However, it is different from the NaturalDataArea because you can use it to allocate a data area with 1:V arrays without resolving the Vs. The DataAreaDefinition is usually used to parse a Natural data area definition so that you can examine its structure, as shown in the following example:

```
Public Sub Main()
    Dim i As Integer
    Dim sdc As New SDCLib6.Application
    Dim dad As DataAreaDefinition
    Dim nfd As NaturalFieldDef
    Set dad = sdc.ParseDataArea("01 CUSTMSA" & vbCrLf & _
        " 02 CUSTOMER-NUMBER (N5)" & vbCrLf & _
        " 02 BUSINESS-NAME (A30)" & vbCrLf & _
        "...
        " 02 CUSTOMER-TIMESTAMP (T)" & vbCrLf)
    For i = 1 To dad.FieldDefs
        Set nfd = dad.FieldDef(i)
        Debug.Print Space$(2 * (nfd.Level - 1)) & nfd.DisplayLine
    Next
End Sub
```

NaturalDataArea Class

The following changes have been made to this class:

- A new property called `DataAreaDefinition` has been added. This property returns the `DataAreaDefinition` object that is used internally to store the structure of the Natural data area definition.
- The `GetField`, `SetField`, and `Reset` methods now allow you to pass index values in an array. This is useful if you want to create helper routines that are capable of processing scalar fields and array fields of any rank identically. For example:

```
Public Sub Main()
    Dim i As Integer
    Dim iindices(1 To 2) As Integer
    Dim sdc As New SDCLib6.Application
    Dim nfd As NaturalFieldDef

    Set nda = sdc.Allocate("01 TEST" & vbCrLf & _
        " 02 ROWS (I2)" & vbCrLf & _
        " 02 GRID-DATA (A3/1:10,1:5)" & vbCrLf)

    Debug.Print ReadField(nda, "ROWS")
    Debug.Print ReadField(nda, "GRID-DATA", 1, 2)

End Sub

Public Function ReadField(DataArea As NaturalDataArea, _
    FieldName As String, _
    ParamArray Indices() As Variant) As Variant

    ReadField = DataArea.GetField(FieldName, Indices)

End Function
```

Dispatcher Class

The CallNat2 method has been removed. You can now use the CallNat method to call application service definitions defined in an external LIF files, or in the code. For example:

```
Dim sdc As New SDCLib6.Application
Dim nda As NaturalDataArea
Dim disp As Dispatcher

Set nda = sdc.Allocate("01 GCD-DATA" & vbCrLf & _
    " 01 #OPERAND-1 (I4)" & vbCrLf & _
    " 01 #OPERAND-2 (I4)" & vbCrLf & _
    " 01 #RESULT (I4)" & vbCrLf)

Set disp = sdc.CreateDispatcher()

' Call an application service defined in an external LIF file.
disp.CallNat "GCDN", nda

' Call an application service defined in the code.
disp.CallNat "DEMO,GCD,1.1.1,DEFAULT,1,1"
```

When using the CallNat method, you no longer need to pass each level 01 field as a separate parameter.

DispatcherServices Class

The Service2 property has been removed. The Service property is now the default property for this class.

DispatcherProperties Class

The Property2 property has been removed. The Property property is now the default property for this class.

NaturalFieldDef Class

The following properties have been added to this class:

- A new property called `FullName` has been added.
This property returns the fully-qualified name of the field (which includes the level 01 structure name and the field name).
- A new property called `Occ` has been added.
This property returns the number of occurrences in each dimension of an array field.
- A new property called `DisplayLine` has been added.
This property returns a string containing the definition of the field suitable for display purposes.
- A new property called `LineType` has been added.
This property returns the type of line, and is always the same as the last character in the `LevelTypeTrail` property.

