

## R Integration in Dell Statistica

Integration options and features to leverage specialized R functionality in Dell Statistica and Dell Statistica Enterprise Server solutions



© 2016 Dell Inc.  
ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Dell Inc.

The information in this document is provided in connection with Dell products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Dell products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, DELL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL DELL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF DELL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Dell makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Dell does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Dell Inc.  
Attn: LEGAL Dept  
5 Polaris Way  
Aliso Viejo, CA 92656

Refer to our website ([software.dell.com](http://software.dell.com)) for regional and international office information.

#### Trademarks

Dell, the Dell logo, Dell Statistica, and Dell Statistica Enterprise Server are trademarks of Dell Inc. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell disclaims any proprietary interest in the marks and names of others.

#### Legend



**CAUTION:** A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.



**WARNING:** A WARNING icon indicates a potential for property damage, personal injury, or death.



**IMPORTANT, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

# Contents

## Contents

<b>Introduction .....</b>	<b>4</b>
<b>Overview .....</b>	<b>6</b>
<b>Basic architecture and features of R support in Statistica .....</b>	<b>11</b>
COM interface to R environment .....	12
R integration supports macros (R.svb / R.r) .....	13
R scripts as native Statistica macros .....	13
Retrieving results .....	15
<b>R language extensions: Passing data to R and retrieving results .....</b>	<b>17</b>
ActiveDataSet .....	17
Definitions of Functions: .....	18
RouteOutput(x,name,description) .....	19
Uses() Functions .....	21
<b>Executing R scripts from within Statistica Visual Basic .....</b>	<b>23</b>
R scripts are Statistica macros .....	23
<b>R support in Statistica Enterprise Server .....</b>	<b>29</b>
<b>Creating R-based Statistica Workspace nodes .....</b>	<b>33</b>
<b>Integrating R into Statistica Enterprise and Statistica Enterprise Server .....</b>	<b>35</b>
Creating R-based analysis configurations .....	35
Calling R scripts from SVB analysis configurations .....	37
Summary .....	37
<b>Final comments .....</b>	<b>38</b>
Error handling .....	38
Strengths and limitations .....	38
<b>About Dell .....</b>	<b>39</b>
<b>Contacting Dell .....</b>	<b>39</b>
<b>Technical support resources .....</b>	<b>39</b>

# Introduction

R is a programming language and environment for statistical computing ([www.r-project.org](http://www.r-project.org)). Most of the R environment and its source code are currently available under the GNU GPL license ([www.r-project.org/licenses](http://www.r-project.org/licenses)).

**NOTE:** None of the components of the R environment constitute unrestricted freeware. Instead, they are available only under the terms of specific licenses. If you intend to download those applications, you must accept prior to downloading and you must comply with their terms. Also, those licenses can change over time. Thoroughly familiarize yourself with the terms every time you download any components of the R environment.

Statistica can interface with third-party applications, such as Microsoft Office, through certain standards-compliant channels.

EXAMPLE: COM is an interface for interprocess communication, which is built into Microsoft operating systems.

Even though R itself does not provide a COM interface, Statistica has created a separate connector application, *COMadaptR*, licensed under LGPL (>=2.1) (with parts under GPL2) that facilitates communication to the R world. This package is based on GPL2/LGPL2 version of an earlier application, the *statconnDCOM* library.

Statistica can interface with R via **COMadaptR** and, if this library is installed on the system, allows you to open R scripts and submit them for execution to R from within the Statistica environment. It facilitates bidirectional data transfer and presentation of resulting outputs through a user-adjustable macro that is executed in Statistica in place of the R script.

This interface makes it possible for all Statistica products, ranging from Desktop to Enterprise and Web solutions, to provide comprehensive support for interaction with the R platform on systems where necessary third-party components and libraries are present, allowing you to:

- run R scripts within the Statistica environment, sending results to Statistica reports, workbooks, and graphs
- process Statistica data sets in R and import tabular results from R into Statistica spreadsheets
- call R from Statistica Visual Basic (SVB) to create new functionality that leverages R libraries
- utilize R in Statistica Enterprise, Statistica Server, and Statistica **Workspaces**

**NOTE:** You are responsible for ensuring compliance with terms of all applicable licenses for R and all components of the R environment. Always carefully review all the license agreements before accepting them, as they can change over time. Those products are external to the Statistica environment, and are not covered by any Statistica license agreements.

Automatic installation of the *COMadaptR* support library is included in Statistica Version 12.0 SP3 and above, or is available with V12A maintenance update UPD008. When the system detects that an R installation is present, it will offer you the opportunity to automatically install the *COMadaptR* package in the R environment. This package is included in Statistica distribution.

You can verify that the *COMadaptR* library is installed correctly, and that all its dependencies are satisfied, by running one of the R examples that accompany Statistica. Once all the required third-party components are

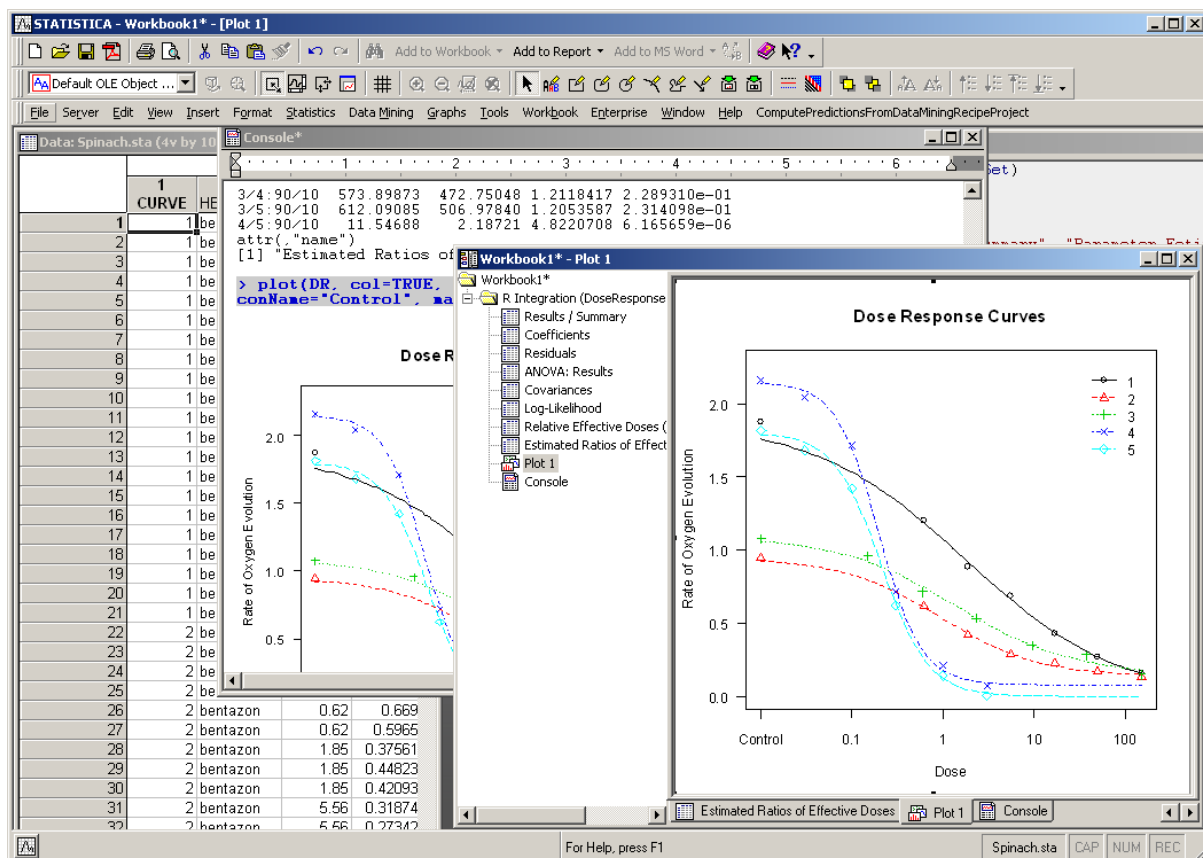
installed, you should be able to open and execute R scripts within the Statistica environment. To locate the examples, on the **File** tab, click **Open Examples** and browse to the **R** folder.

R is highly extensible. Users can submit libraries (packages), implementing a set of functions, usually for a specific area of their expertise/research. The R community maintains several centralized repositories, which make hundreds of such packages readily available to all users over the Internet. Many of these packages cater specifically to highly specialized audiences with particular data analysis needs.

# Overview

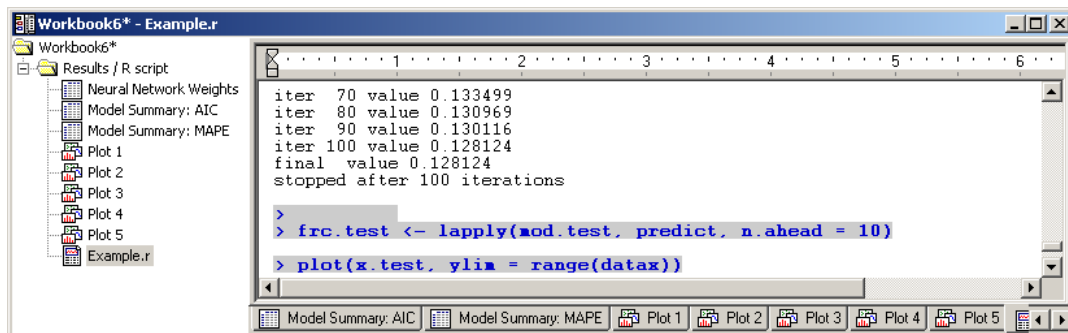
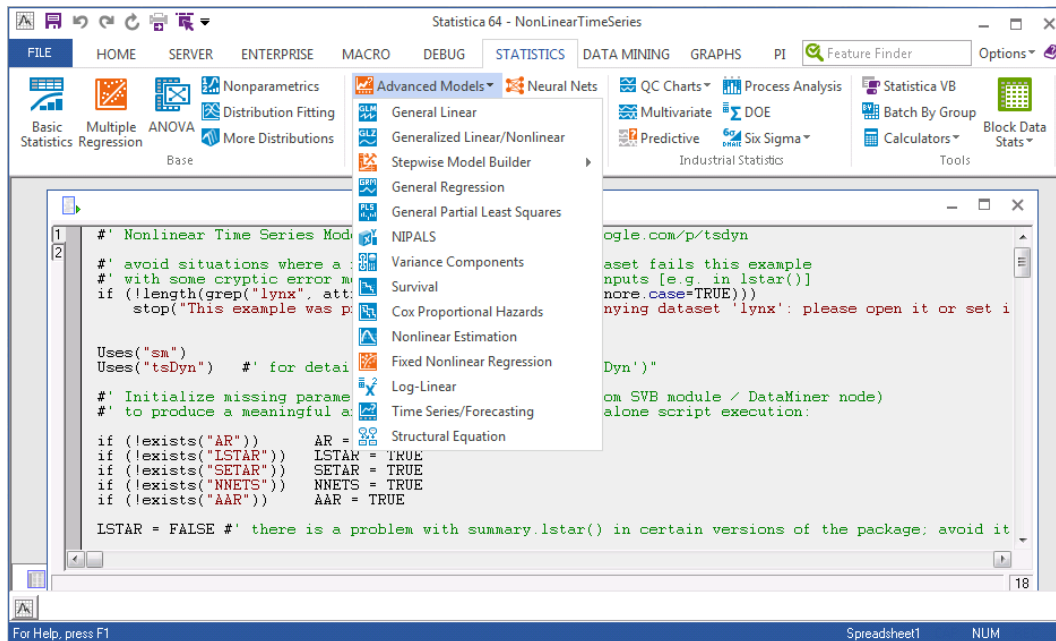
This document provides a detailed description of features that make the diversity and power of R fully available to users of all Statistica solutions. Furthermore, these features enable you to combine the unique capabilities of both Statistica and R platforms.

- You can run Native R scripts directly within Statistica, Statistica Enterprise, and Statistica Enterprise Server.
- You can retrieve R output as native Statistica spreadsheets and graphs, and manage them via highly flexible Statistica workbook containers.



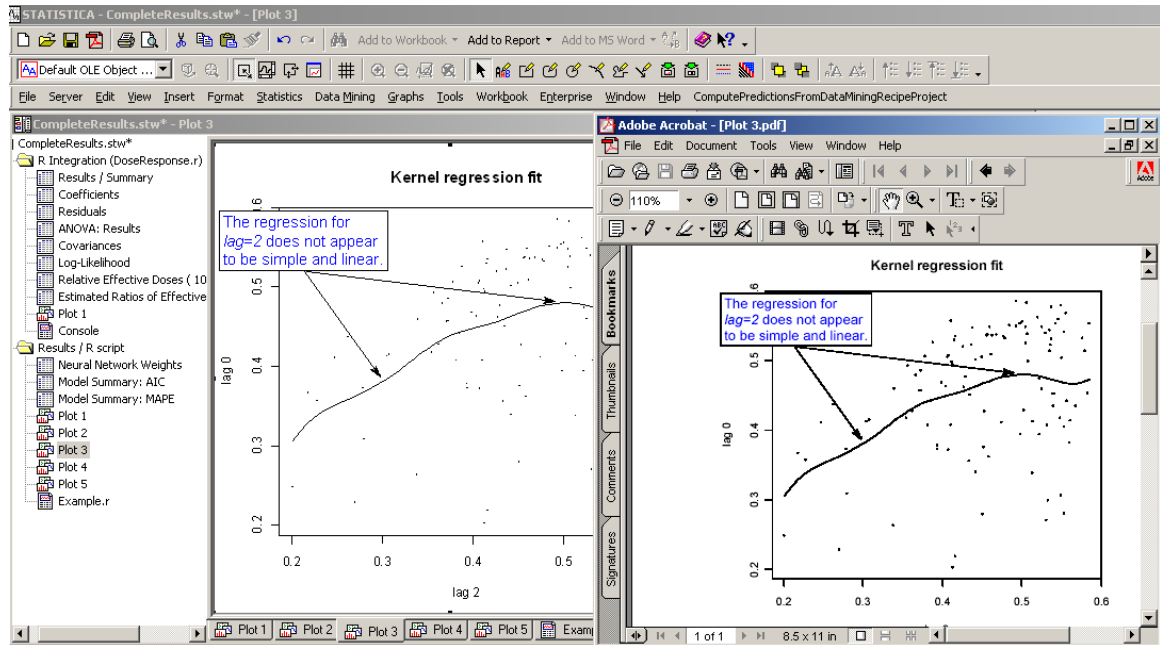
Enterprises can now use the specialized routines and capabilities of R with Statistica, Statistica Enterprise, and Statistica Enterprise Server for these functionalities:

- Add new R-based modules



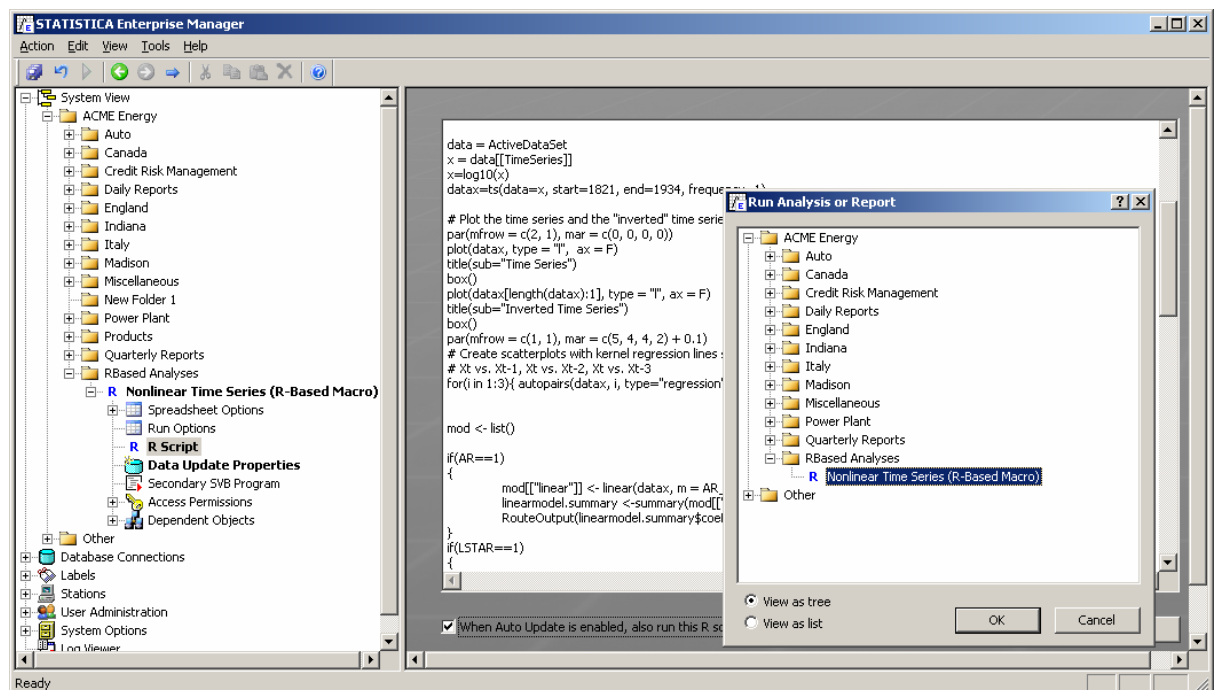
- Leverage Statistica's superior features to handle output from R

- Graphics
- Flexible spreadsheets
- Convenient workbook containers for various document types



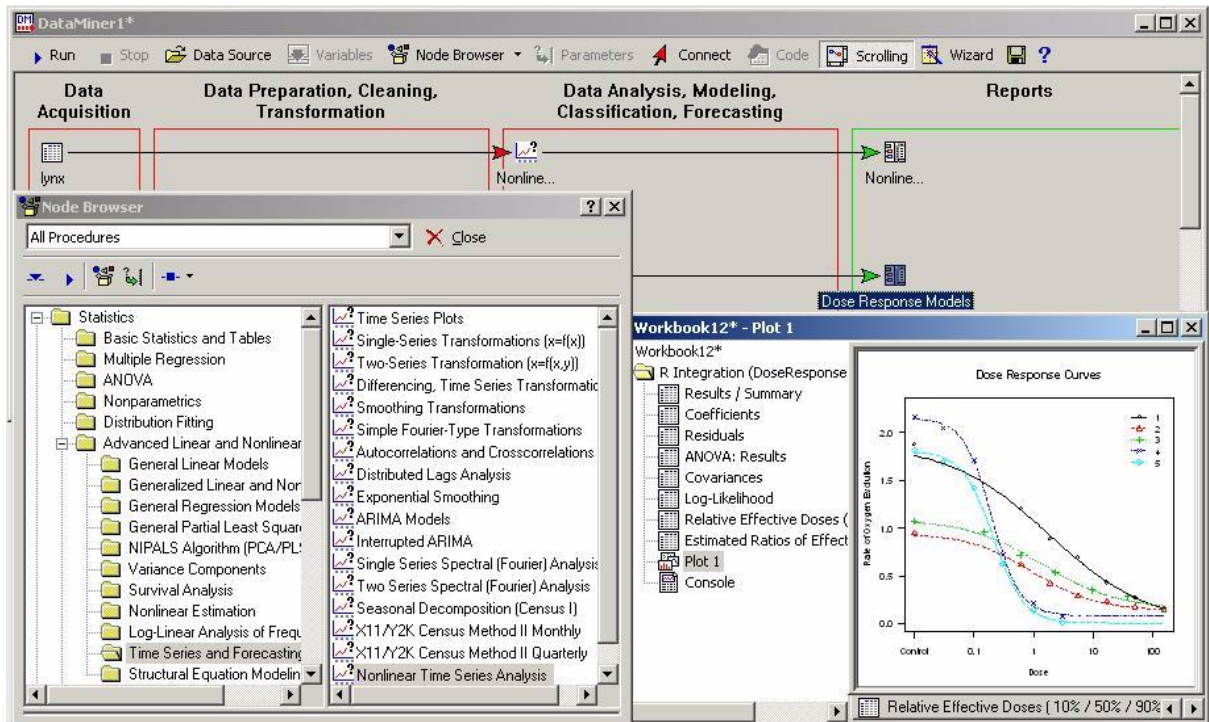
- **Integrate R into Statistica Enterprise**

Make specialized R functionality available as reusable analysis templates to users not familiar with the R language, in a secure, role-based enterprise analysis system.

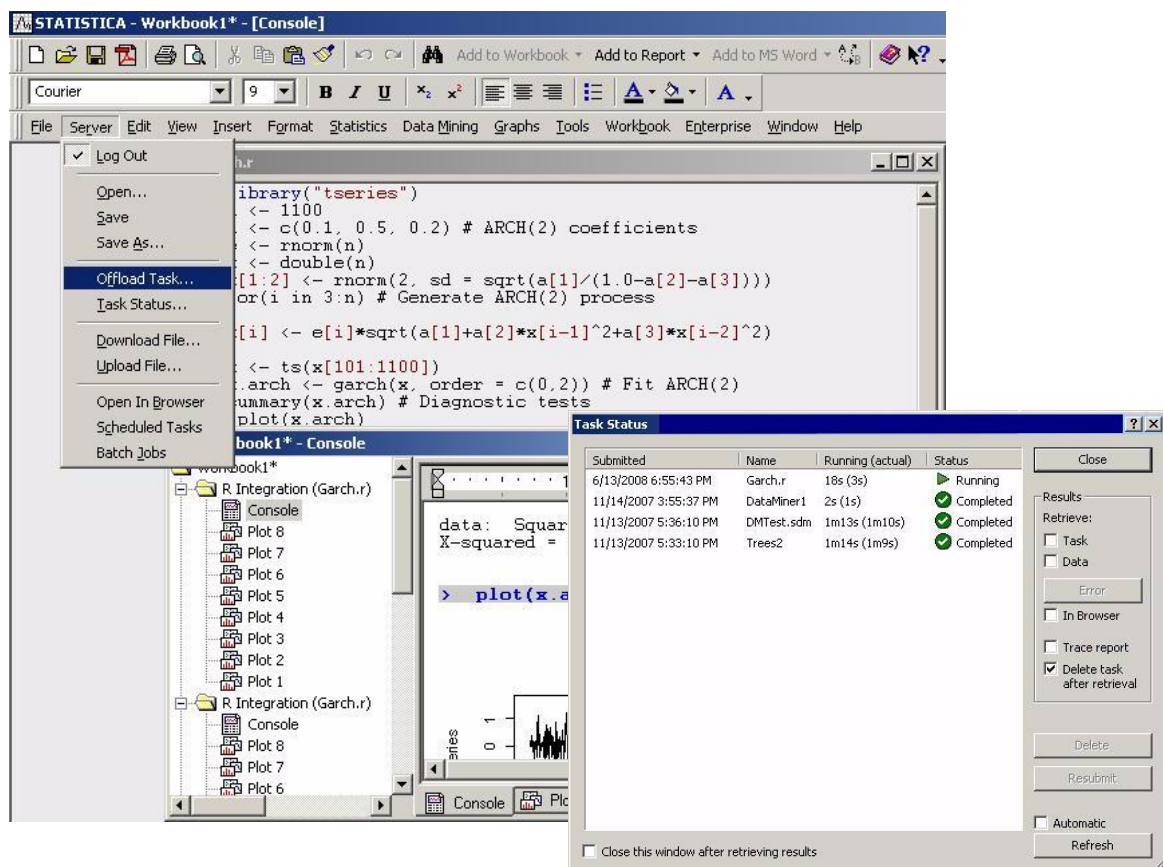


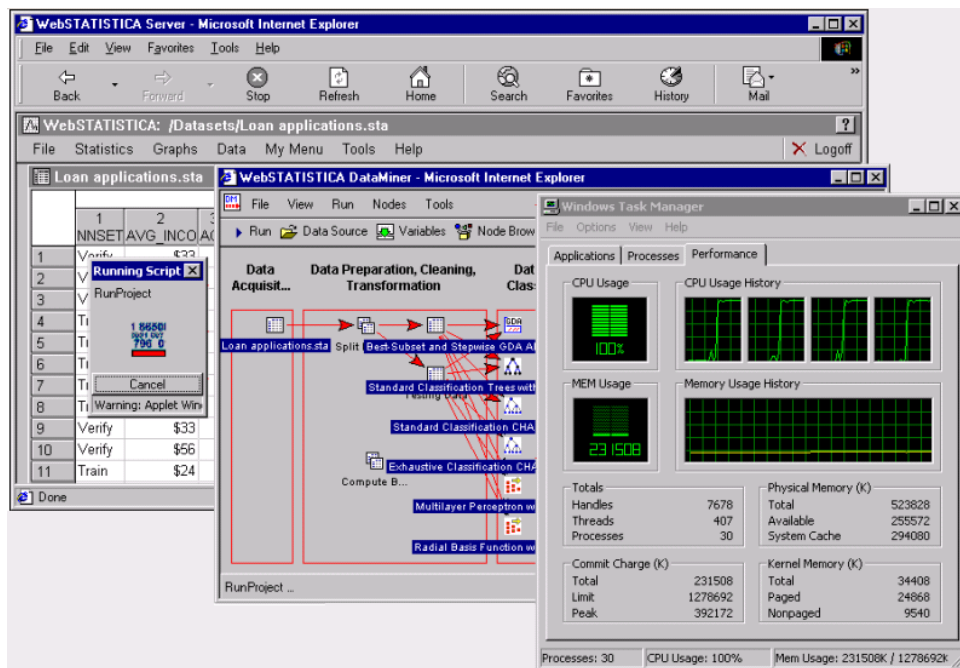
- **Add R-based analytic nodes to Statistica Data Miner**

Leverage all R capabilities inside Statistica Workspaces.



- Build scalable R servers
  - Use Statistica Enterprise Server to handle security and load balancing
  - Use it to take advantage of multiple-processor servers to run R for demanding and/or validated enterprise applications.





## Basic architecture and features of R support in Statistica

R support in Statistica creates an integrated Statistica-R platform that enables you to run R programs (scripts) directly inside Statistica. By doing so, you can take full advantage of these specialized capabilities available in R:

Enables users to run R scripts as is, and retrieve the results into Statistica reports	All R console output is copied into the report, and R commands are highlighted.
	Plots generated by the script are automatically embedded in the report as scalable images.
	These plots are also replicated as Statistica graphs (scalable metafile images are placed inside these graphs), thus enabling annotation using powerful graphical facilities in Statistica
	The graphs can be printed or exported into a variety of image formats.
Provides R language extensions functions for R scripts run from the Statistica environment	The reports can be edited, printed, and saved as PDF files.
	Transfer data from Statistica spreadsheets into R data frames.
Executes R scripts as native macros from within Statistica Visual Basic (SVB) programs	Extract tabular data from R variables into Statistica spreadsheets. These results spreadsheets (as well as all graphs produced by the script) are returned (routed) according to <b>Output Manager</b> settings, into Statistica workbooks.
	Scripts can be parameterized with a <b>Collection</b> of objects (numbers, strings, arrays, additional R code or overridden R functions, spreadsheets), which are mapped to R variables accessible to the script. This approach provides fine-grained control over scripts' behavior in repeated runs or when they are used as the backend for custom Statistica modules.
	By default, all script output is routed by Statistica <b>Output Manager</b> . Alternately, scripts can be executed using a method that returns its output as a <b>Document Collection</b> , giving developers an easy way to extract specific analysis results that could be used for further processing, such as input data for further analyses in Statistica or in R, or for graphing.

Taken together, these enhancements not only enable you to run R scripts directly in the Statistica desktop environment, but also provide a way to embed specialized R functionality into these features:

- SVB programs
- Custom interactive analysis modules
- **Workspace** nodes
- Enterprise analysis configurations

You can also offload such scripts to Statistica Enterprise Server for server-side processing.

Enterprise customers have the opportunity to integrate R into validated Statistica Enterprise solutions, such as those used by FDA-regulated industries, or to provide a very powerful Statistica Enterprise Server-based R server environment.



**CAUTION: Developing and debugging R scripts:** The R support in Statistica was not designed to supply a complete R development and debugging environment. The console application and tools supplied with standard R installation perform those functions very well and are already familiar to R users and developers.

## COM interface to R environment

In order for R integration in Statistica to transfer data between Statistica and R, the R environment must be installed in one of these locations:

- On the same computer
- On Statistica Enterprise Server
- On the server for Enterprise solutions

The latest version of the R environment can be obtained from the CRAN website (<http://cran.r-project.org>).

To access the R environment, Statistica uses the COMadaptR library distributed under GNU Lesser Public License. This library consists of two components:

- One acts as an R COM Server.
- The other is used for callbacks from R to Statistica.

Once R is installed, Statistica will automatically detect it. The next time you run Statistica, a dialog box will be displayed.

- 1 Select to enable **R Integration**.
- 2 Click **Yes**, and Statistica will automatically install the **COMadaptR** library and register one of the components with the Windows registry.



**NOTE:** Depending on your operating system version and settings, the system may ask you to confirm these actions, possibly requesting administrator credentials.

- 3 Complete **COMadaptR** library installation by accepting the installer's default settings.

You are now ready to use R within the Statistica environment. The **COMadaptR** library is independent from Statistica and will remain on your computer until it is uninstalled manually.

If you do not need **R Integration** enabled, you can suppress the dialog box. The **Integrating R into Statistica** dialog box will display when you start Statistica again.

- 1 If you do not wish it to display, select the **Do not ask me again** check box.
- 2 You can enable it again by selecting the corresponding check box on the **Documents: Macros** tab of the **Options** dialog box.

The **COMadaptR** library provides a simple yet powerful COM (Component Object Model) interface to the R environment. This interface can be used directly by SVB programs in Statistica.

An example of such a use case is included in the Statistica application, located in the **Examples\R\Dose Response** folder. To use it, open **Direct Interface To R via COM.stw** and run the embedded SVB macro.



**CAUTION:** Usage of such an interface directly by end users is very ineffective, sometimes unproductive, and usually inflexible. It may also significantly degrade the overall performance of interactions with R if performed incorrectly.

Therefore, the following architectural extensions have been added to the Statistica platform to provide a seamless and effective R Integration experience for end users. The example mentioned above also demonstrates the significant reduction in the efforts required to implement the same analysis using the new built-in features. In Statistica, simply open and run **DoseResponse.r**.

## R integration supports macros (R.svb / R.r)

Statistica installation includes two elements. **R.svb** and **R.r** contain the support code required to manage interactions through COM in Statistica and the R side respectively.

### R script called R.r

Support script **R.r** implements Statistica-specific R language extensions functions.

One of its primary responsibilities is translating Statistica spreadsheets to R data frames and back.

When an R script is executed in Statistica, the support macro parses the script and executes these processes:

- 1 Transfers data and script parameters
- 2 Submits script content to the R environment
- 3 Manages error conditions
- 4 Handles script outputs, ensuring that they are properly transferred back to Statistica.



**CAUTION:** Although the support code in **R.r** is write-protected by default, you can inspect it and modified or enhanced it to support new functionality required for specific use cases, although you do so at your own risk.

### Statistica Visual Basic macro called R.svb

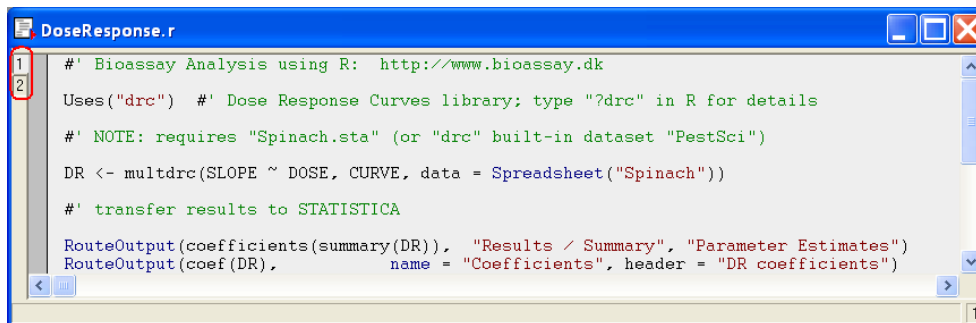
The **R.svb** macro supports standalone execution to simplify debugging and testing of modifications.

## R scripts as native Statistica macros

Statistica recognizes **.R** (and **.S**) file extensions as R scripts. Open these files by selecting the **File** tab and clicking **Open**.

It also registers (at the operating system level) **.R/.S** files as Statistica macros, so you can open these files in Statistica from a file browser by double-clicking on them.

R scripts are displayed in slightly modified **Statistica Visual Basic Macro** windows. These windows actually contain two scripts, the R script itself and the **R Integration Support Macro (R.svb)**, accessible through two tabs in the upper-left corner (circled in the next image).




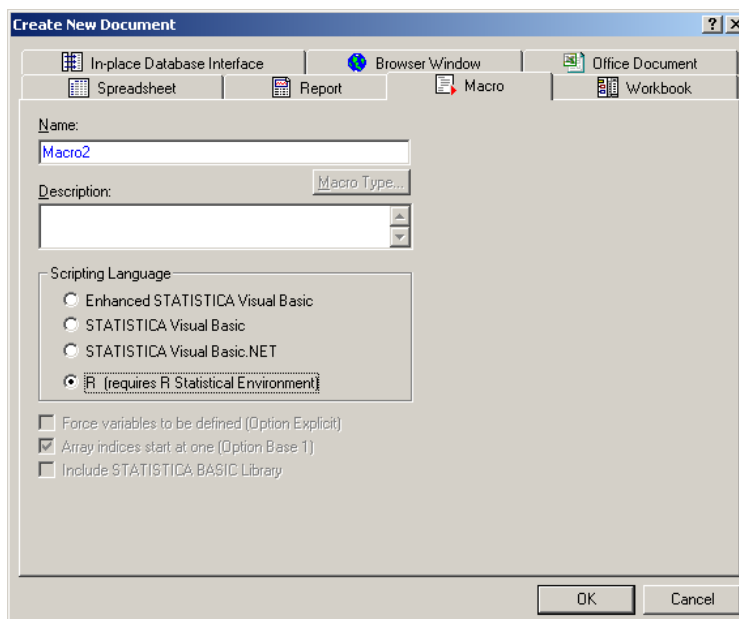
Limited R code highlighting is available (strings, language extensions, VB-style comments).

## Creating a new R script

You can also create a new R script within Statistica.

- 1 Select the **File** tab.
- 2 Select **New**.
- 3 Under the **Classic New Document Dialog** heading, click **New Document** to display the **Create New Document** dialog box.
- 4 Select the **Macro** tab.
- 5 Select the **R (requires R Statistical Environment)** option button.
- 6


 **NOTE:** This option is available only if R integration support is installed on your machine.




- 7 Click **OK** to open an empty R script window.
- 8 You can now type or paste in an R program.


R Integration support also includes an optional text file called **R.inc**, located in the default installation directory along with **R.svb**. The contents of the file are copied to the beginning of each new R script created in this manner.

## Running R scripts within Statistica

To execute the **R.svb** macro for the currently active R script, on the **Macro** tab, in the **Debug** group, click the  button or press the **F5** key on your keyboard.

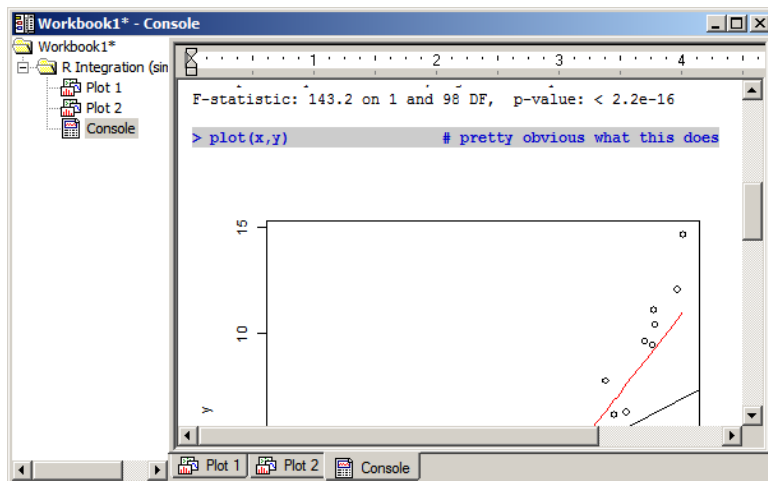
Although breakpoints are not supported for the R script itself, you can set breakpoints and debug the **R.svb** macro on the second tab of the modified Statistica Visual Basic Macro window while running the R script.

 **NOTE:** In order to take advantage of R Integration features described in this document, R scripts should be executed from within Statistica.

 **CAUTION:** Although you can develop and debug complex R programs within this environment, it was not specifically designed for these purposes. The R environment itself is better suited for such activities.

## Retrieving results

The minimal output produced during execution of an R script is a Statistica workbook that represents an R console session. It will include highlighted commands and any output generated by the R environment. This kind of report is produced even if the script is empty. The contents of this report can be edited and manipulated in the same way that you would edit any other Statistica report.



What happens to all plots created during an R session:

- They are automatically transferred into the Statistica environment as Windows metafiles (vector graphics format). They can be resized without loss of quality.
- They are placed into the R console session report. They create a natural flat report of the R session, with embedded plots tied to the graphics commands, which is not easily achieved in the R environment itself.
- They are replicated as Statistica graphs that become a part of the R script output.

- The metafile images are embedded into graph objects as locked background.

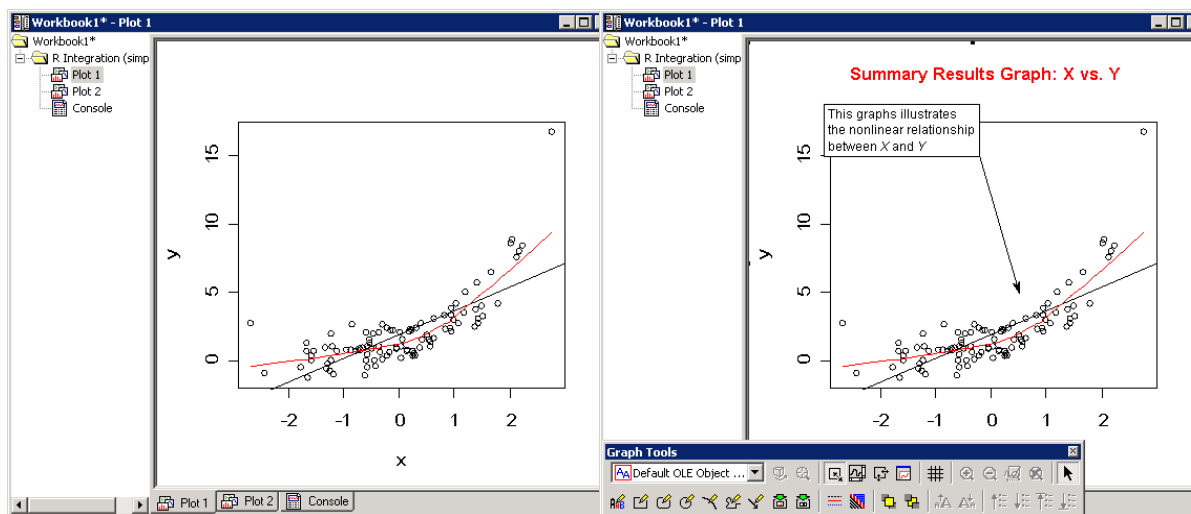
You can annotate R plots in Statistica, using a familiar point-and-click interface, with a set of text and drawing objects such as lines and arrows, rectangles and ellipses, polygons and pattern/color fill areas, etc.

Since these annotations are anchored to relative positions in the plot area, they will remain correctly attached to the plot if the graph is resized.

You can flexibly design and further enhance these graphs using Statistica graphics tools. You can save them in other formats (JPG or GIF), or print them to PDF files.

Since the individual R plot components (the structural elements of the plot) are not accessible for manipulation in Statistica graphs, the rich capabilities of Statistica for creating and then further editing graphs (scaling, point markers, fit lines, etc.) are not available.

However, integration between R and Statistica lets you extract data from R. You can then render important graphs inside the Statistica environment by writing Statistica Visual Basic macros that will execute R scripts, extract results, and then post-process those results as necessary, which will be illustrated later.



## Unique Statistica capabilities

R script output consists entirely of native Statistica objects. These objects are properly routed by Statistica according to **Output Manager** settings. Depending on user selection, they could be placed into standalone windows or in a workbook.

These are examples of ways you can manage each of these objects, using the extensive capabilities of the Statistica platform:

- Annotate
- Store in a (compressed) workbook
- Export as Microsoft Office documents
- Print
- Save as a PDF file
- Convert to one of many popular formats
- Archive as version-controlled, auditable items in the Statistica Document Management System
- Share among other users in a web-based, client-server Statistica Enterprise Server environment

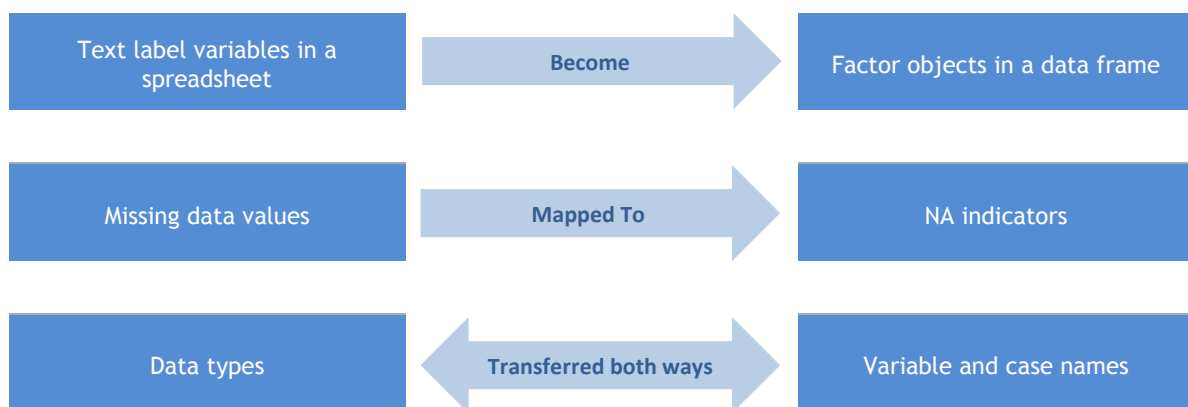
## R language extensions: Passing data to R and retrieving results

The R Integration Support Macro (**R.svb**) implements several extensions to R language. These keywords and functions can be used inside R scripts executed within the Statistica environment. These extensions enable scripts to pass data to R and retrieve results from the R environment.

**IMPORTANT:** The R language is case sensitive; therefore, R language extensions for Statistica are also case sensitive. Extensions are only recognized by the Statistica environment when typed exactly as shown.

Tabular data represented in Statistica in the form of spreadsheets are mapped into the equivalent R structures: data frames.

Mapping preserves as much information as possible for both formats, as illustrated by these examples:



### ActiveDataSet

The **ActiveDataSet** keyword has been adopted from the Statistica Visual Basic language. It performs the same function in R scripts by referencing the active Statistica data spreadsheet.

In the desktop Statistica environment, the active data set usually means the top-most visible spreadsheet that can act as a data source. However, it can also be a spreadsheet in a workbook selected as **Active Input**.

This concept is redefined and extended for server-based environments (Statistica Enterprise Server, Enterprise), but the keyword is still valid and refers to the corresponding server-side mapping of the active data source.

If no active data set is defined/available, the R script that uses it will fail. The same is true for SVB macros.

If the R Integration Support Macro encounters the `ActiveDataSet` keyword in the R script, it transfers the actual Statistica active data set into the R environment and assigns it to a variable of the same name. Therefore, this keyword represents a data frame variable and can be handled as such in the script.

EXAMPLE:

```
ActiveDataSet[1:5] # display a subset of active dataset (variables 1 through 5)
str(ActiveDataSet) # view the structure of the data frame
plot(ActiveDataSet$MEASURE01) # plot "MEASURE01" variable values
```

## Definitions of Functions:

This section explains the following functions: `Spreadsheet(path.or.object, [applySelectionConditions], [recodeTextLabelAsFactors], [recodeMissingDataToNA], [getCaseNames], [getTextValues], [attachObject], [username], [password], [connectionstring], [stationname])`

Function	Values/Description	Examples and Notes
<code>Spreadsheet()</code>	extension	Used to load a specific Statistica data file into R and to transfer the data in that file to an R data frame
<code>ActiveDataSet</code>	Keyword	Similar to <code>Spreadsheet()</code> extension
<code>path.or.object</code>	Character vector (literal or variable) or a valid COM object pointing to a Statistica spreadsheet	<ul style="list-style-type: none"> <li>C:/data/sample.sta</li> <li>Enterprise://SomeFolder1/SomeFile.sta</li> <li>Adstudy</li> </ul>
<code>[applySelectionConditions]</code>	TRUE (default) or FALSE flag	Indicates whether selection conditions defined on spreadsheet should be employed
<code>[recodeTextLabelCodesAsFactors]</code>	TRUE (default) or FALSE flag	Indicates whether text labels defined for variables should be treated as factors or text
<code>[getCaseNames]</code>	TRUE (default) or FALSE flag	Indicates whether row names should be fetched
<code>[attachObject]</code>	TRUE or FALSE (default) flag	Indicates whether spreadsheet should be attached to data frame as an Object attribute
<code>[username]</code>	character strings (default value: "")	Used to fetch spreadsheets from the enterprise system when user does not have an integrated login
<code>[password]</code>		
<code>[connectionstring]</code>		

## Spreadsheet()

Like the `ActiveDataSet` keyword, the return value of the `Spreadsheet()` function should be treated as a data frame variable with the contents closely matching that of the corresponding Statistica spreadsheet.

One useful feature supported by this function is the use of default search paths for spreadsheet files that are specified only as simple file names. This means that if the function parameter consists only of a file name, such as `Spreadsheet("some.sta")`, R integration support code looks for this file in several locations:

First, it checks the folder where the R script itself is located (if it was saved to disk).



Next, it checks the Examples\Datasets folder for the current Statistica installation.

Support code will also append the default .sta file extension if one is not present. Therefore the following options are available:

- R scripts can reference the accompanying data sets (placed in the same folder) simply by name.
- Spreadsheets that are included in every Statistica installation as demonstration (Example: data sets) can be referenced by name in much the same way as built-in R data sets.

EXAMPLE:

```
Spreadsheet("c:\\myfiles\\mydata.sta")           # display the dataset in R console
Spreadsheet("\\server\\share\\data.sta")         # same, but read from a network share
Spreadsheet("thisdemo.sta")                     # file in the same folder as this script
Spreadsheet("enterprise://folder/thisdemo.sta")  # file in STATISTICA Enterprise
# plot a histogram for MEASURE05 variable from STATISTICA example dataset Adstudy:
hist(Spreadsheet("Adstudy")$MEASURE05)
...
advert = Spreadsheet("Adstudy")$ADVERT           # retrieve a data frame variable (factor)
is.factor(advert)                               # => [1] TRUE
levels(advert)                                  # => [1] "PEPSI" "COKE"
```

## RouteOutput(x,name,description)

**RouteOutput** transfers various types of data from the R environment into Statistica spreadsheets:

<b>x</b>	a data frame, matrix, array, number, or a string
<b>name</b>	character vector containing name (in Statistica) for the object being routed
<b>description</b>	character vector containing a description for the object being routed
<b>SpreadsheetName, SpreadsheetHeader</b>	literal string, such as "A Frequency Table"

Although the function was introduced to retrieve tabular data (such as data frames, matrices, or arrays) into spreadsheets, single-value data, such as numbers or strings, can be passed as well. They will be placed in single-cell spreadsheets. **x** can be an R variable or a literal value.

The extension, **RouteOutput()**, is similar to the equivalent Statistica Visual Basic function. The results spreadsheets recreated by this function in the Statistica environment become the standard output of the R script/analysis. They follow **Output Manager** settings in Statistica.

**NOTE:** In Statistica:

- Select the **Tools** tab.
- Click **Options**.
- In the **Options** dialog box, select the **Analyses/Graphs: Output Manager** tab.

The results are routed either to individual windows or to a workbook (or multiple workbooks) for each analysis, with optional output reports, such as a Microsoft Word document. The most popular setting is a single results workbook.

Optional parameter name and description specify the name and header of the resulting spreadsheet. Provide a value for the spreadsheet name for visual distinction in the tree view of the results workbook.



**NOTE:** R plots transferred into Statistica as native graphs do not require explicit output routing. All plots generated during a script run are automatically transferred and routed according to Output Manager settings.

Many functions in R, specifically the ones that perform statistical modeling, represent their results as structured objects, sometimes of significant complexity.

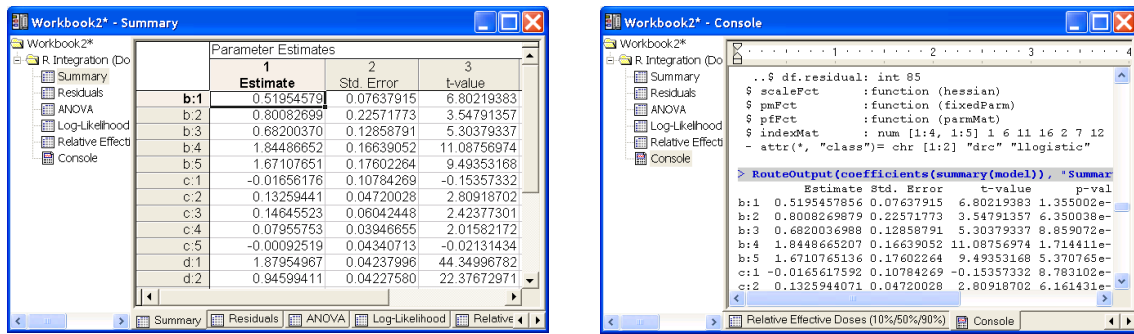
These objects cannot be reduced to a single table, and therefore cannot be handled by the **RouteOutput()** extension. They could be automatically searched for tabular components, but since the object structures are specific to a particular method, such an approach would generate a significant amount of “junk” output.

However, since the results (the actual data of interest) are either stored in such objects as tabular components or produced by applying an object’s method to some input data, this limitation does not pose any problems. Particular results can be easily extracted from such a statistical model object and routed back to Statistica.

EXAMPLE:

```
# build a statistical model (Dose Response Curves)
model <- multdrc(SLOPE ~ DOSE, CURVE, data = PestSci)
...
str(model) # display the complex structure of this model/results in R console
...
# transfer results to Statistica
RouteOutput(coefficients(summary(model)), "Summary", "Parameter Estimates")
RouteOutput(residuals(model), name = "Residuals", header = "Some Header")
RouteOutput(anova(model), "ANOVA") # inherits the default heading from data frame
RouteOutput(logLik(model), "Log-Likelihood")
RouteOutput(ED(model, c(10, 50, 90)), "Relative Effective Doses (10%/50%/90%)")
```

The results for this example may look like this:



## Uses() Functions

This section covers the following functions: `Uses(package,[lib], [repos], [quiet], [attachImports])`.

The `Uses()` function ensures that the respective package (library), named `package`, and its dependencies (packages that are required in order for this package to run) are:

- 1 installed on the computer/server where the R script is running
- 2 loaded into the R environment automatically

If some of these libraries are not present, they will be installed from CRAN repository (<http://cran.r-project.org>) and then loaded.


Functions	Descriptions
<code>package</code>	Character vector containing name of package, such as <code>drc</code>
<code>lib</code>	Character vector containing path to library directories to install package if needed
<code>repos</code>	List of repositories to use to download package if needed
<code>quiet</code>	TRUE or FALSE (default) flag indicating whether interactive dialog boxes should be suppressed during package installation
<code>attachImports</code>	TRUE OR FALSE (default) flag indicating whether to attach imports from the package and its dependencies to the current R session

R defines several levels for specification of package dependencies.

One of them, **Imports**, lists packages with **namespaces** that have to be loaded for the package to run. However, the namespaces do not need to be attached to the current/global environment, so the package accesses such imported packages using namespace-qualified names, such as `tseriesChaos::mutual(x)` instead of `mutual(x)`.

Three ways to use some of these namespace-enclosed functions in your own R code:

- Explicitly use namespace qualification [`tseriesChaos::mutual(x)`].
- Attach the library to current environment [`Uses("tseriesChaos")` or `library(tseriesChaos)`].
- Set the optional `Uses()` parameter `attachImports` to `TRUE`, which will attach all the imports of the package.

 **NOTE:** This extension is not necessary for interaction between Statistica and R. You can implement it within the R language itself. It simplifies the process of conditional library installation and loading by encapsulating it in a single call.

EXAMPLE:

```
Uses("drc") # make sure that the respective package is installed and loaded
...
DR <- multdrc(SLOPE ~ DOSE, CURVE, data = PestSci) # call the package methods
```

This program fits dose response curves to the respective variables of the built-in PestSci data set by calling the **multdrc** function defined in the “**drc**” package. **Uses(“drc”)** ensures that the function is available by installing and loading the package, if necessary.

## Executing R scripts from within Statistica Visual Basic

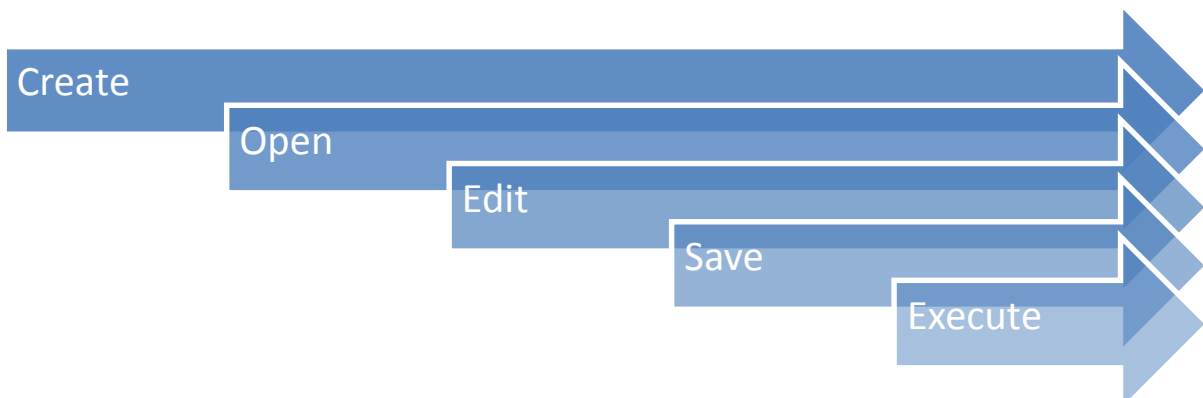
Leverage specialized R functionality within Statistica by calling R scripts from inside a Statistica Visual Basic (SVB) macro to accomplish the following tasks:

- Build new modules using the SVB User Interface library and methods implemented in R scripts.
- Create Statistica Enterprise SVB analysis configurations.
- Create Statistica **Workspace** nodes (for Statistica or Statistica Enterprise Server) that leverage R.

To provide any non-trivial functionality within an R script in such use cases, you must parameterize that script with user-selected parameters, variables lists, input spreadsheets, etc. Statistica provides a simple and powerful way to pass such parameters to R scripts.

### R scripts are Statistica macros

In almost all cases, Statistica treats R scripts in the same way as it treats native SVB macros. This rule applies to the Statistica Object Model as well. Now Macro objects in SVB programs can represent R scripts. Therefore, R scripts can be created, opened, edited, saved, and executed from within SVB scripts.



R functionality is available in Statistica Enterprise analysis configurations and Statistica **Workspace** nodes, since they are SVB-based.

Existing R script files can be opened with **Macros.Open("path\to\some.r")** or created on-the-fly with **Macros.New()** and **Macro.Code**.

In the latter case, Statistica needs help in distinguishing R scripts from SVB macros, which can be achieved by either of two methods:

- Specify the name for a new macro with the .R extension, even if you are not going to save it on disk.
- Explicitly set **Macro.Scripting** to 5 (R Macro Type).

Run the scripts by calling **Macro.Execute**.

**NOTE:** The Macro.Scripting type for R scripts is 5. It will later be mapped to a symbolic constant.

EXAMPLE:

```
Sub Main
  Dim R As New Macro
  R.Code = "ActiveDataSet"      ' simple R script created on-the-fly
  R.Scripting = 5              ' R Macro Type = 5
  R.Execute
End Sub
```

This Statistica Visual Basic macro runs a simple R script containing only a single command **ActiveDataSet**. As described in the previous section, this script is an R language extension for Statistica that will transfer (and in this case display) the currently active Statistica data file in R.

EXAMPLE: If you run this macro after opening the example data file **Exp.sta**, a listing of that file will be displayed in a report window that represents the R console session, as shown below:

## Passing parameters to R scripts: Collection Object

A **Collection** is a COM library, introduced with Statistica version 8 MR3, that implements generic **Collection** objects. Such objects enable users to store keyword-tagged Variant values. Variants are COM structures that can store one of many supported data types, including arrays and references, to other COM objects, such as spreadsheets.

So in R integration, you can now store {parameter name = parameter value} pairs in such a **Collection**, then pass this parameter set to Statistica prior to R script execution, to achieve script parameterization. The R Integration Support macro will transfer all the parameters from the **Collection** to the R environment as R variables.

The variables are named using the respective parameter keyword tags, which makes it possible for the script that is executed in this environment to reference all these variables (script parameters) directly by name.

R integration supports the following types of named parameters:

- Strings
- Numbers
- Arrays
- Spreadsheets

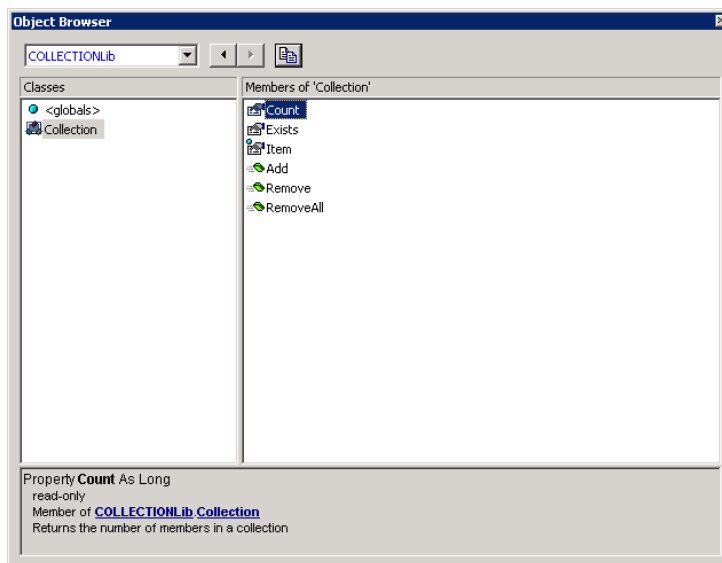
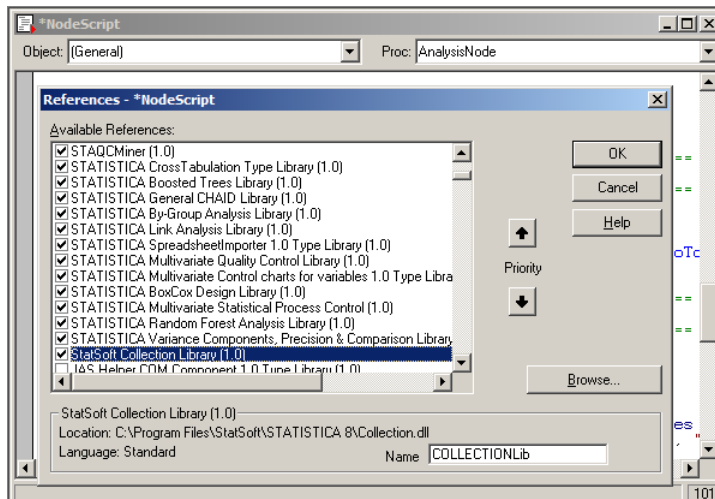
Support for Statistica spreadsheets is particularly useful.

EXAMPLE: Such a parameter can override the **ActiveDataSet** keyword, allowing a single script to serve as an R engine for an interactive module, Enterprise analysis configuration, and **Workspace** node implementations.



**NOTE:** String parameters without a keyword tag are treated as R code that should be executed prior to execution of the script itself. This is analogous to SVB hidden code and can be used to, for example, define a common set of new functions or global variables/constants.

In order to use **Collection** objects, SVB scripts must include a reference to the Statistica **Collection** Library. Add it via **Tools > References** while editing the macro.



The **Collection** object has several generic properties and methods that are needed to manipulate the contents of a **Collection**:

- **Count**
- **Exists (key)**
- **Item (key)**, which adds a member to the collection with the specified key or returns an existing member of the collection with the specified index or key
- **Add (item, [key])**, which adds a member to the collection with an optional associated key
- **Remove (Index or key)**

- Remove All

However, due to the use of so-called default object properties (**Item** is the default property of a **Collection** and it returns its **Value** property by default), interaction with a **Collection** object is reduced to intuitively clear assignment operations:

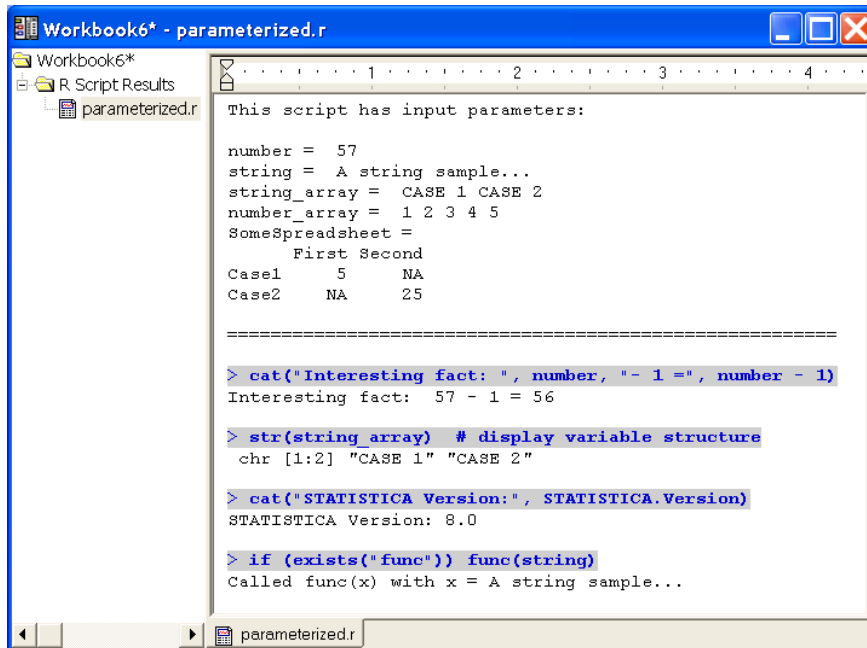
EXAMPLE:

```
Dim param As New Collection
param("number") = 57
param("string") = "A string sample..."
```

Once you have assembled the parameter **Collection**, execute the parameterized R script by calling **Macro.ExecuteWithArgument(Parameters As Collection)**.

EXAMPLE:

```
Dim s1 As New Spreadsheet, s2 As New Spreadsheet ' ... populate s1, s2 with data
var1 = Array("CASE 1", "CASE 2")
var2 = Array(1, 2, 3, 4, 5)
' * don't use spaces in parameter names
' * some names are "locked" and can't be used [e.g. 'text', 'str', 'sample']
Dim param As New Collection
param("number") = 57
param("string") = "A STRING sample..."
param("string_array") = var1 ' add items with an assignment operator
param.Add(var2, "number_array") ' OR using explicit Add() method
param("SomeSpreadsheet") = s1
param("ActiveDataSet") = s2 ' override the value of 'ActiveDataSet' keyword
' string parameters without associated keys will be treated as R code and
' will be executed before the script - an analog of SVB 'hidden code'
' * define a function that will be available to the R script
param.Add("func <- function(x) { cat('Called func(x) with x =', x) }")
' * another way to define a global constant or variable
param.Add("STATISTICA.Version = "" & Version & """)
' now run the R script with this collection of parameters
' (parameters become R variables - the script can reference them by name)
Dim m As Macro
Set m = Macros.Open(MacroDir & "\parameterized.r")
m.ExecuteWithArgument(param)
```



## Managing script results

Both `Macro.Execute()` and `Macro.ExecuteWithArgument()` methods handle R script output (spreadsheets, graphs, reports) in exactly the same way as the output from SVB macros. The output is routed by Statistica according to **Output Manager** settings.

EXAMPLE: The output is placed into a results workbook.

This approach might be sufficient for many use cases, but is not flexible enough in situations where the output from an R script is treated as an intermediate result for subsequent processing in Statistica.

EXAMPLE: An SVB script implementing a custom interactive module might implement a multiple stage analysis workflow, calling into R, analyzing the results, requesting additional inputs from the user, and calling into R repeatedly with previous results as inputs. Or, an SVB macro could use tabular results returned from R to produce graphs that are more expressive and flexible than what R is able to generate.

Since the items processed by the Statistica **Output Manager** are intended to be a final representation of analysis results, accessing them individually for further processing is not a trivial task. You would need to know the type of representation selected (single or multiple workbooks, individual windows) and programmatically iterate through the respective set of output items (windows, workbook contents represented as a tree), searching the item of interest by name.

### `Macro.ExecuteNoRouteOutput([Parameters As Collection]) As StaDocCollection`

For cases such as this, the above method has been added, where R script output is not routed at all. As a result, the script runs silently, and all of the output from the script is collected into a **StaDocCollection** object. This standard Statistica container for documents such as spreadsheets, graphs, reports, or workbooks is returned as the result of script execution. It also accepts an optional **Collection** for parameterization.

SVB macro developers can easily access individual components of R script output. For instance, you can extract individual cell data from spreadsheets or create a complex graph based on multiple columns from several spreadsheets.

## EXAMPLE:

```
Dim m As Macro
Set m = Macros.Open(MacroDir & "\some.r")

Dim Routput As StaDocCollection
Set Routput = m.ExecuteNoRouteOutput() ' silent, no output displayed

' do something with "Routput": extract and operate on individual documents

' trivial case: results are routed (displayed), as if from Macro.Execute()
RouteOutput(Routput, "R Script Results").Visible = True
```

## More examples

We have demonstrated all the functional components required to build custom applications within the Statistica platform that can take advantage of the specialized functionality available in R.

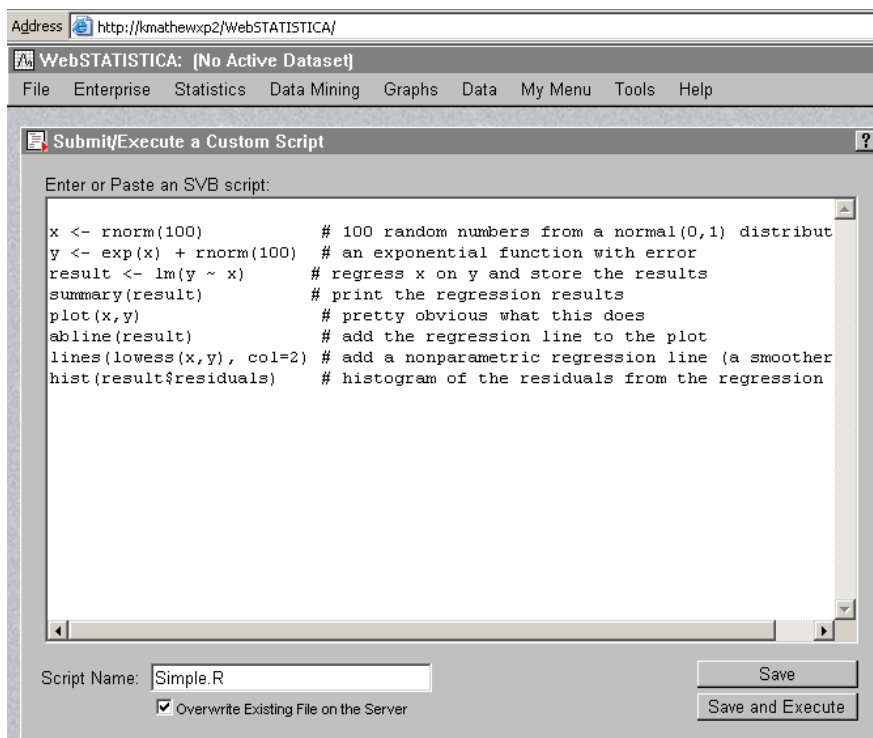
All installations of Statistica contain a set of examples that provide a detailed demonstration of the R integration features. You will find these examples in the [Statistica]\Examples\R folder. These examples can also be used as templates for your own development.

# R support in Statistica Enterprise Server

Statistica and Statistica Enterprise Server are based on identical Statistica libraries, and support identical functionality. This is true for R support in Statistica as well. You can execute R in Statistica Enterprise Server in much the same way as from desktop (thick-client) Statistica.

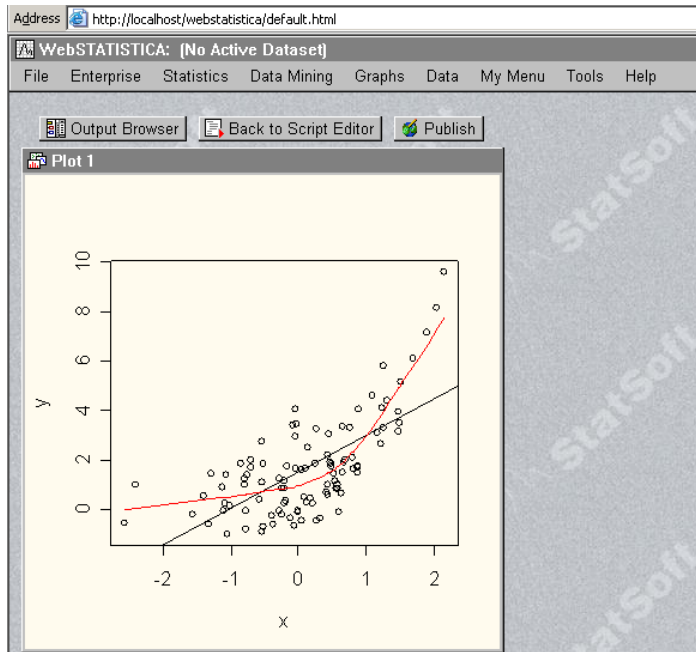
EXAMPLE:

- 1 Log on to Statistica Enterprise Server.
- 2 Select **Tools > Submit/Execute a Custom Script**.
- 3 Type in the macro shown below (available as **standalone.r** in the **Examples\R** folder).



**NOTE:** Save the script with the **.R** extension.

- 4 To run the script, click **Save and Execute** or use the **File** menu to open and run it.



**NOTE:** This script does not require a data file. If you are using **ActiveDataSet** or **Spreadsheet()** extensions, keep in mind that the data sources must be located on the server:

- **Spreadsheet()** should reference data sets by Statistica Enterprise Server URL paths.
- **ActiveDataSet** will be mapped to the spreadsheet that you open in Statistica Enterprise Server.

This rule can be overridden by a parameter of the same name passed to the R script from SVB. Also, in order to execute any scripts in Statistica Enterprise Server, you must have the proper permissions on the server.

## Statistica Enterprise Server is a powerful R server

Statistica Enterprise Server was designed as a powerful and flexible server/web-based analytical platform, relying on the Statistica Visual Basic engine for the diversity of its functionality, as well as extensibility.

R scripts are handled by Statistica Enterprise Server in much the same way as standard SVB macros. The third-party components that the Statistica platform relies upon to provide its R runtime environment (such as R library and R COM Server library) are also well suited for handling multiple simultaneous R sessions.

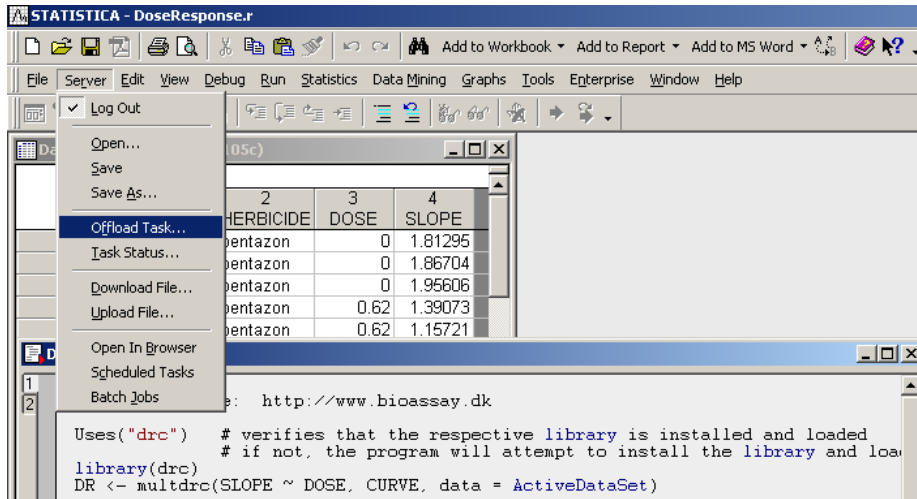
Thus, Statistica Enterprise Server represents an ideal platform for a powerful and flexible multiprocessor R server that can handle a large number of users, providing security, scheduling, load balancing, etc.

## Offloading R scripts from Statistica to Statistica Enterprise Server

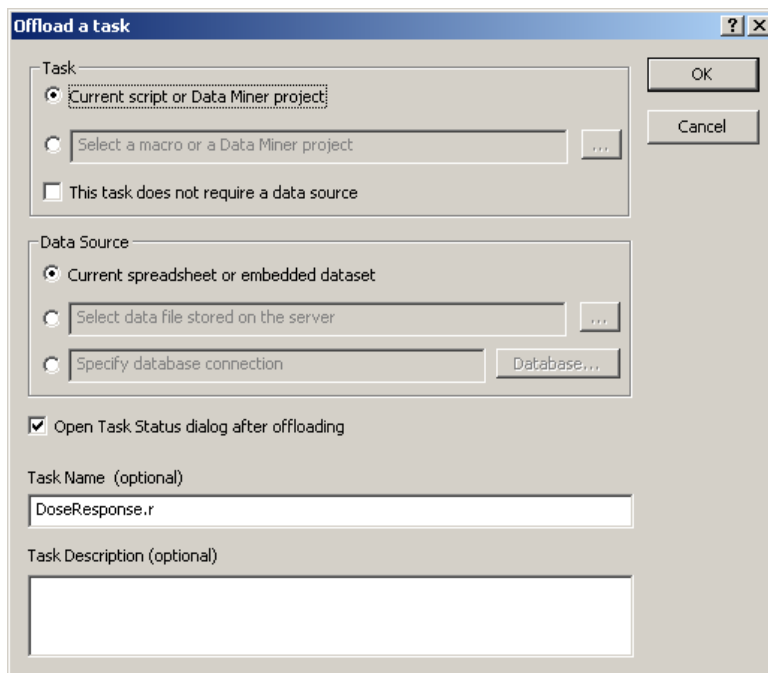
Like SVB scripts or **Workspaces**, R scripts can be offloaded from a desktop environment for processing on Statistica Enterprise Server. This capability is particularly useful when an R script is expected to require significant computing resources. You can submit such scripts to the server and retrieve the results at a later time.

EXAMPLE: Allow them to process overnight.

If your installation is set up to work with Statistica Enterprise Server (**Tools > Options: Server/Web**), you can offload any R task by selecting **Server > Offload Task....**

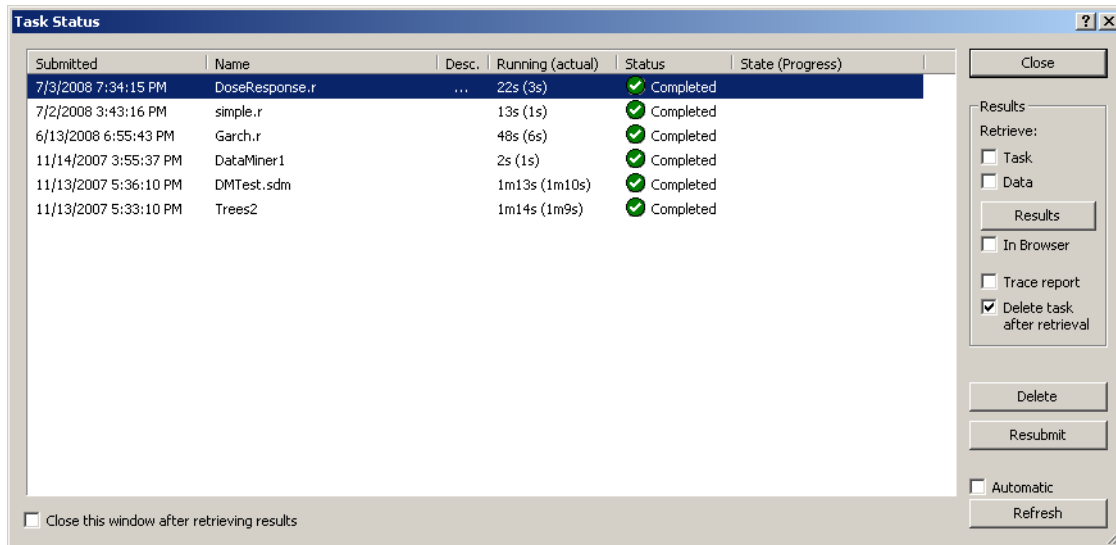


Options are also provided to transfer the data file to the server side. These options are described in detail in the Statistica documentation.



After clicking **OK**, the respective R script will execute on Statistica Enterprise Server. It might be scheduled to be executed rather than be executed immediately, depending on the server load.

The progress of the analysis can be monitored via the **Server > Task Status** dialog box.



You can retrieve the results from the server via the **Results** button or you can double-click on the task. The representation of the results of an offloaded task is equivalent to the same task running locally.

## Creating R-based Statistica Workspace nodes

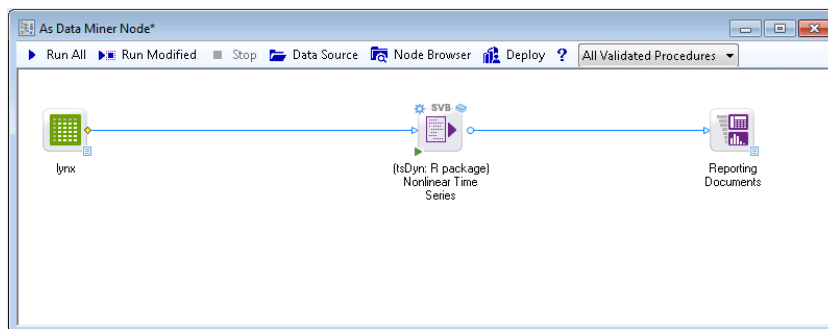
You can use the techniques described in previous sections to implement a custom R-based **Workspace** node:

- 1 Edit the SVB node's source to generate a **Collection** based on the node's input parameters.
- 2 Use it to execute one or more parameterized R scripts.
- 3 Retrieve the results as a **StaDocCollection**, which becomes the output of the node.

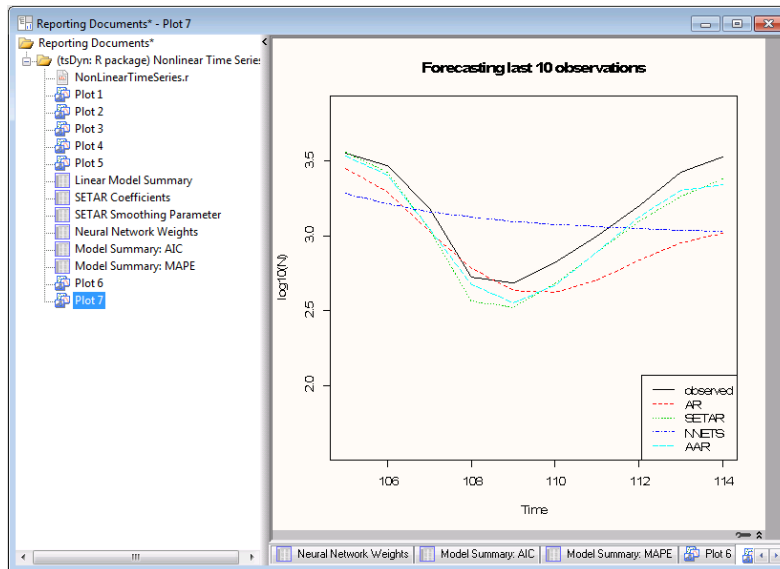
EXAMPLE:

- 1 Open Examples\R\NonLinear Time Series\As Data Miner Node.sdm.
- 2 In the **Workspace**, right-click the analysis node.
- 3 Select **Edit Code** to review the underlying SVB code (**NonLinearTimeSeries.svx**).

**NOTE:** The corresponding **.dmi** file describes the node's input parameters. The R script file (**NonLinearTimeSeries.r**) performs nonlinear time series analysis using the **tsDyn** R package. This script can serve as a template for your own R-based **Data Miner** nodes.



- 4 After running the node, double-click the **Reporting Documents** node to display the workbook.



This example demonstrates that the ability to incorporate specialized R functionality into Statistica Data Miner provides very powerful opportunities for creating analytic workflows or **Collections of Data Miner** nodes and node-browser configurations dedicated to highly specialized analyses (such as nonlinear time series, dose response curve fitting, and so on).

# Integrating R into Statistica Enterprise and Statistica Enterprise Server

R Integration in Statistica would be incomplete without support for Statistica Enterprise and Statistica Enterprise Server solutions. Now, R programs can be used as analysis templates, and R analyses can be distributed to end users who are not familiar with VB or R programming.

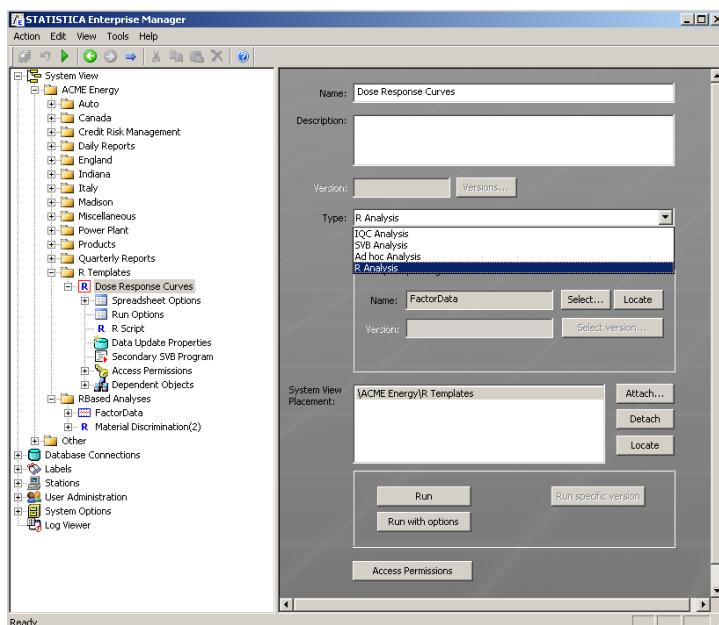
The Enterprise platform provides numerous options and features to specifically enable the application of templated analyses for validated applications, such as for manufacturers who need to comply with the requirements of FDA 21 CFR Part 11, such as version control and audit trails. These features make R available for deployment in mission-critical manufacturing applications.

## Creating R-based analysis configurations

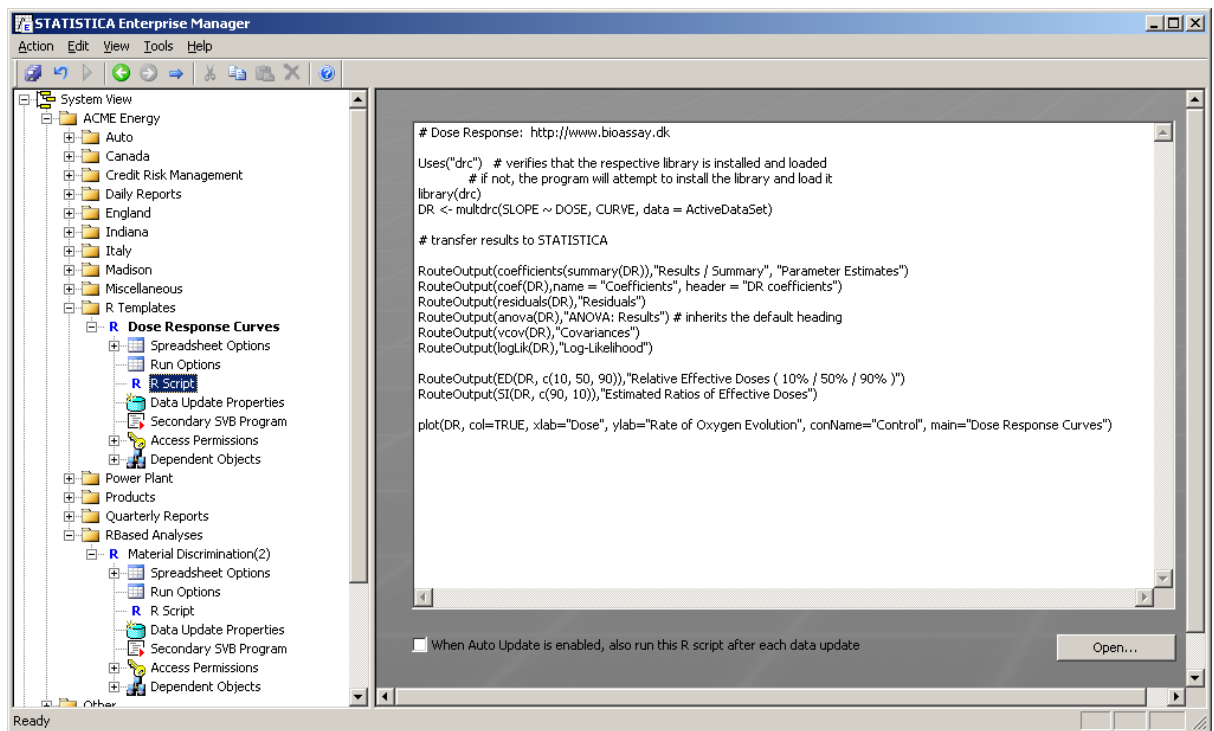
The Statistica Enterprise system implements standard control charting and SPC analyses out-of-the-box. Non-trivial analysis templates are usually created using Statistica Visual Basic, for example, by using Statistica macro-recording capabilities. In addition, Statistica Enterprise and Statistica Enterprise Server now recognize R scripts and can store and process them in the same way as SVB scripts.

When creating a new Analysis Configuration in Statistica Enterprise:

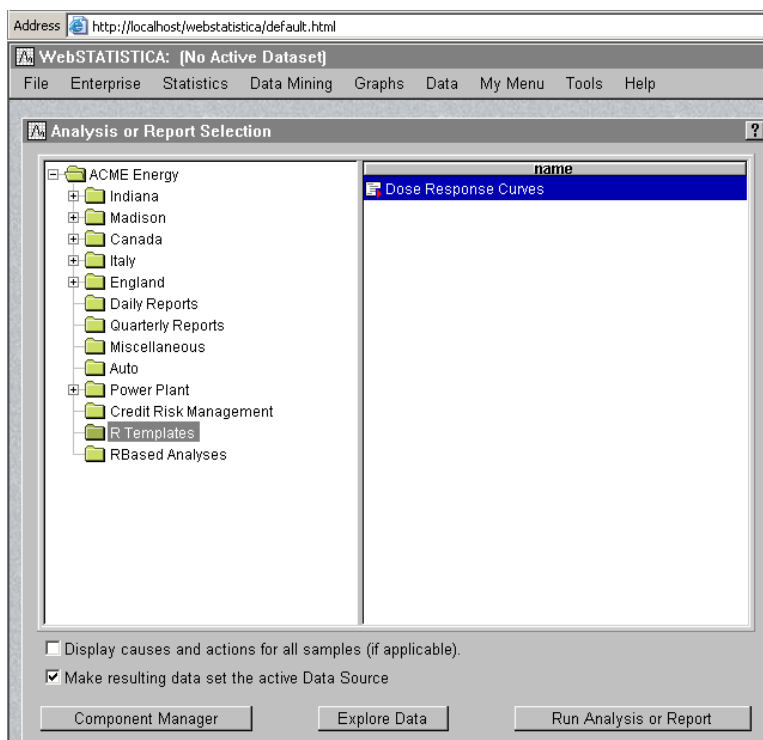
- 1 Select the **R Analysis** option.



- 2 Type, paste in, or load the respective R script.
- 3 All other options are identical to those available for SVB-based analysis templates.



Depending on the configuration of the Statistica Enterprise system, this reusable analysis template is now available to the respective users of both Statistica and Statistica Enterprise Server environments. The results of this template can be combined into standard reports.



Moreover, if the respective Statistica Enterprise installation is integrated with the Statistica Document Management System, these R scripts can be locked down. For instance, you could version them with complete audit trails to support FDA 21 CFR Part 11 requirements.

## Calling R scripts from SVB analysis configurations

R functionality can also be utilized from inside SVB analysis templates. They can retrieve the results from R for further processing or display. In this case, R scripts that are loaded for the respective templated analyses in Statistica Enterprise can be called in either of two ways:

- They can be placed into a secure folder repository, from which they can be loaded, and where they can be managed via Statistica Document Management System integrated with Statistica Enterprise.
- They can be embedded into the SVB code. To use this method, assign R script text to the SVB macro's **Code** property.

## Summary

The following capabilities, made possible by the full integration of the R environment into Statistica Enterprise, make Statistica Enterprise arguably the most powerful enterprise analysis system to date:

- Utilize extensive and highly specialized functionality of R.
- Leverage it inside a secure, enterprise-wide, role-based analysis system that insulates end users from implementation details and that can be deployed into validated manufacturing environments.
- Run R-based templates on the Statistica Enterprise Server platform.
- Extract results into pre-formatted standard reports.

## Final comments

The features provided in Statistica for integration with R are quite flexible. Using them, thousands of highly specialized R functions and features become available to all Statistica solutions. However, if you plan to exploit these features, please consider the following system limitations of the R platform.

### Error handling

Most of the error conditions generated within the R environment, such as syntax and runtime errors caused by an R script or by the integration support libraries, such as broken R installation or missing components, are intercepted and handled by Statistica. Developers can use error handling facilities available in either environment.

**EXAMPLE: On Error** handlers in SVB macros calling R scripts.

However, R programs can occasionally crash or hang. In the latter case, program control does not return to Statistica. Therefore, careful validation of the respective R scripts is crucial for enterprise-level deployment of R analysis templates.

### Strengths and limitations

#### A word of caution regarding the quality of R algorithms

R comes without warranty or guarantees. In practice, many (most) of the algorithms available in R are the result of diligent work over many years by one or a few experts in their respective methodology or domain.

However, this does not mean that the software was created following rigorous software development lifecycle methodology, or stringent standard operating procedures for software requirements gathering, design, implementation, and testing.

Therefore, in order to build a mission-critical or validated application around a component that depends on R, it is absolutely critical that you carefully validate all results for the use cases to which the software is to be applied.

#### A word of caution regarding scalability, large data sets, etc.

Another caveat regarding R that needs to be considered before building solutions around R concerns its basic architecture. Unlike Statistica, data in R must be, in practically all cases, resident in the computer's memory. This restriction, in combination with hardware-level and operating system-level memory limitations, may or may not pose an obstacle for any one individual user, but will need to be considered carefully when building R-based server applications accessible to multiple users.

#### A word of caution regarding consistency

Results generated using R might vary across versions of R, platforms (Windows, MacOS, Linux), and platform bitness (32-bit vs 64-bit) for identical input values and identical programs. Statistica faithfully transmits to and from R, and is not responsible for tracking, reporting, or adjusting for such changes.

# About Dell

Dell listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit [www.software.dell.com](http://www.software.dell.com).

## Contacting Dell

Technical Support:

[Online Support](#)

Product Questions and Sales:

(800) 306-9329

Email:

[info@software.dell.com](mailto:info@software.dell.com)

## Technical support resources

Technical support is available to customers who have purchased Dell software with a valid maintenance contract and to customers who have trial versions. To access the Support Portal, go to <http://software.dell.com/support/>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. In addition, the portal provides direct access to product support engineers through an online Service Request system.

The site enables you to:

- Create, update, and manage Service Requests (cases)
- View Knowledge Base articles
- Obtain product notifications
- Download software. For trial software, go to [Trial Downloads](#).
- View how-to videos
- Engage in community discussions
- Chat with a support engineer