Statistica Visual Basic - one of the strongest development environments
## Statistica Visual Basic

### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistica Visual Basic</td>
<td>3</td>
</tr>
<tr>
<td>Application scenarios for Statistica Visual Basic programmes</td>
<td>3</td>
</tr>
<tr>
<td>Statistica Visual Basic Macro recording</td>
<td>4</td>
</tr>
<tr>
<td>- Analysis Macros</td>
<td>4</td>
</tr>
<tr>
<td>- Master Macros</td>
<td>4</td>
</tr>
<tr>
<td>- Node adaptions</td>
<td>5</td>
</tr>
<tr>
<td>Statistica Visual Basic Editor and Debugger</td>
<td>5</td>
</tr>
<tr>
<td>Visual Basic from other applications</td>
<td>6</td>
</tr>
<tr>
<td>Running Statistica Visual Basic Macros</td>
<td>6</td>
</tr>
<tr>
<td>- Performance of Statistica Visual Basic programmes</td>
<td>6</td>
</tr>
<tr>
<td>- Structure of Statistica Visual Basic</td>
<td>6</td>
</tr>
<tr>
<td>Statistica programming in .NET</td>
<td>7</td>
</tr>
</tbody>
</table>
Statistica Visual Basic

The industry standard Statistica Visual Basic (SVB) programming language is integrated into Statistica and is another alternative user interface that includes the full functionality of Statistica. It is much more than just a programming language for creating advanced applications.

Note that Statistica Visual Basic should not be confused with Microsoft Visual Basic 6.0. Statistica Visual Basic is part of Statistica and the code is currently being maintained and further developed. SVB is compatible with Microsoft’s VB.NET, Microsoft’s Visual Basic for Applications (VBA) and Microsoft’s Visual Basic 6.0 (VB6). Especially the flexibility and compatibility of the SVB scripting language should be emphasized. SVB provides access to Visual Basic for Applications (to control Microsoft Office products) and access to the .NET framework within the same file. Other programming interfaces, such as Yahoo’s Stock Quote API or Google Analytics API, can also be accessed and underline the flexibility of SVB. SVB offers a powerful 64-bit solution for system integration, expansion and customer development.

Statistica Visual Basic takes full advantage of Statistica’s object model architecture and can control every aspect and almost every detail of Statistica’s functionality. Even the most complicated analyses and graphs can be recorded as Visual Basic macros. These macros can later be run repeatedly, edited or composed of parts of other applications. Statistica Visual Basic adds over 14,000 new features to Microsoft Visual Basic’s rich standard syntax, making it one of the most comprehensive development environments available.

Application scenarios for Statistica Visual Basic programmes

The spectrum of applications for Statistica Visual Basic programmes ranges from macro recording to automating common tasks to developing complex analysis systems that combine Statistica’s optimized procedures with custom extensions in a self-created user interface.

With appropriate licensing, created analysis scripts can be integrated into system environments in order to be executed from the company’s own software applications or by means of Internet/intranet portals. In addition, SVB programmes can be attached to almost any important „event“ in a Statistica analysis, such as opening or closing files, clicking cells in a table, and so on. In this way, the STATISTICA user interface can be adapted in great detail for specific applications (e.g. data entry, &c.).

Multiple scripting languages are included in Statistica. When creating separate macro documents, SVB, Enhanced SVB, Statistica Visual Basic.NET or R can be chosen.

Enhanced Statistica Visual Basic is an extension of Statistica Visual Basic and contains additional functions. Statistica Visual Basic.NET provides direct access to .NET assemblies, so that no COM Interop is required, as required by the standard SVB.

In addition, the user-defined script nodes in the workspace are available in languages Python, Iron Python, C#, Spark Scala, SVB and R languages. R is a programming language and environment for statistical calculations. For more details about R, Python, C# and Spark Scala refer to Open Source.

With Statistica 13, the workspace is now available to all Statistica users and is enjoying increasing popularity. If data analysis includes several steps that require different dialogues to be called, these steps can be mapped in the workspace using individual nodes. The vast majority of work steps can be automated in this way with out-of-the-box nodes, without the need for a single line of script. Adjustments of the node outputs can be executed with recorded adjustment scripts, which can be optionally stored for each individual output. If standard nodes with customization scripts should not be sufficient to meet customer-specific requirements, the script nodes mentioned above can be used, thereby it is easily possible to combine script nodes of different languages in a workspace. In addition, the script nodes completed for the special requirements can be added to the node browser so that they are available to all users for repeated routine tasks.
Statistica Visual Basic Macro recording

Statistica offers a wide range of options for recording macros and/or SVB programmes to automate periodically repeating operations or to automatically create programmes that can be edited and modified for the following tasks. The macros recorded in this way can be executed exactly as they were stored, or can serve as building blocks for more complex and highly customized Visual Basic programmes. Analysis macros and master macros follow the same syntax and can both be modified later. However, due to the different recording paths of these macros, they offer different advantages and disadvantages for a particular application.

Analysis Macros

Analysis Macros automatically record the settings and selected options of a specific analysis. The term „analysis“ refers to a task called up from Statistics, Data Mining or Graphics tabs. This task can be very small and simple (e.g. a scatter plot from the Graphics tab), but also very complex (e.g. a structural equation model from the Statistics tab that can contain hundreds of output documents).

After selecting a command from Statistics, Data Mining or Graphics tabs, all actions, such as variable selection, selected options, &c., are recorded in the background. This recording (i.e. the Visual Basic code for this macro) can be displayed at any time in the Visual Basic Editor. The generated macro command required is available in every analysis dialogue after clicking the options button.

If an analysis dialogue has been minimized and is listed in the analysis item, the command can also be called up in the corresponding context menu (access by right mouse click on the analysis button in the analysis bar).

Master Macros

With a Master Macro, you can record an entire session consisting of one or more analyses. This recording combines the various analyses from Statistics, Data Mining or Graphics tabs with their different analysis options. In addition, the recording of a Master Macro can also be switched on and off in contrast to the recording of the Analysis Macro.

The recording of the Master Macro starts by selecting the command protocol recording of analysis (Master Macro) in the start tab of the macro menu, and ends when the stop recording command is selected in the same menu.

Apart from these two actions, in addition to the analyses and the selected order of these analyses, selections of files and data operations are also recorded.
Node adaptations

The third variant of script recording refers to the workspace, which has replaced the dialogue mode as the most popular user interface due to its excellent usability. In the context menu of a node, the output options can be called up so that the node adaptation dialogue is invoked.

Any output can be selected in the upper window of the node adaptation dialogue. If the record script button is then selected, the subsequent script recording is automatically added to this output.

Statistica Visual Basic Editor and Debugger

In the professional Statistica Visual Basic development environment, programmes can also be completely rebuilt. A convenient programme editor with a powerful Debugger (including breakpoints) and many other tools for efficient code building are available for this purpose.

When a macro is edited entering Visual Basic commands or SVB program instruction, an automatic word recognition for automatic completion of these commands is activated. In addition, a tooltip appears that indicates the correct syntax of the command. A help to the members and functions of each individual class can also be called.

Breakpoints can be set in the programme so that line by line it is possible to observed how the variable values change during programme execution.
In summary, Statistica Visual Basic is not just a powerful programming language, but a very powerful, professional development environment that is ideal for creating simple macros as well as complex custom applications.

**Running Statistica Visual Basic macros**

Statistica Visual Basic programmes can be executed within Statistica. However, since SVB is compatible with the industry standard, programmes can also be executed from any other compatible environment (e.g. Excel, Word or a pure Visual Basic language). In practice, this means that you can usually call up Statistica functions using Visual Basic in another application.

Note that (unlike functions from MS Visual Basic) to start an SVB programme or attempt to call Statistica functions from other applications, the Statistica libraries must be present on the running computers for the functions to run. This means that the user of the programme must be a licensed user of the Statistica libraries for the procedures. Furthermore, it should be noted that this huge Statistica function library (with more than 14,000 procedures) is not only open to access by Visual Basic, but also to call up from any other compatible programming language or environment such as C/C++, C# or Delphi.

**Performance of Statistica Visual Basic programmes**

The obvious advantages of Visual Basic (compared to other languages) are its ease of use and its popularity with a large number of PC users. A possible disadvantage of Visual Basic programmes is that the processing speed is lower than that of applications developed in a lower-level programming language (such as C).

This potential problem usually does not exist with SSVB applications, especially if they consist mainly of call-ups of analyses, graphs and data manipulations from Statistica. These procedures fully implement Statistica’s technology to run as fast as if the corresponding procedures were performed directly in Statistica.

**Structure of Statistica Visual Basic**

Statistica Visual Basic consists of two components: (1) The general Visual Basic development environment with tools and extensions for the development of user interfaces (dialogues) and (2) the Statistica libraries with thousands of functions that allow access to practically all functionalities of Statistica. The Visual Basic development environment basically follows the syntax conventions of Microsoft Visual Basic. Minor differences are mainly in the way dialogues are created to give programmers/developers more flexibility in creating user interfaces within complex programmes.

In the SVB development environment, dialogues can be processed completely within separate subroutines, which can be flexibly combined to larger programmes with multiple dialogues, whereas under MS Visual Basic the dialogues and all events within these dialogues are processed in separate programme units.
Statistica programming in .NET

Almost every aspect of Statistica is also available in the form of COM interfaces, which are registered on a computer during the Statistica installation. Due to the fact that .NET languages cannot communicate directly with COM, a wrapper class called COM Interop is used to integrate the Statistica libraries into the .NET project. If Visual Studio .NET is used as an integrated development environment, the COM Interop is created automatically when a COM interface is imported. COM Interop is responsible for all details regarding the interaction of the COM libraries within .NET. After connecting COM Interop, the STATISTICA COM interfaces can be handled like any other .NET object.