Program Execution

Description

Program Execution component can start execution of the external program (there are the following executed files: cmd, bat, exe, etc.)

> **Important:** This component is not safe as it provides access to programs and file system. Rights to this operation are defined in Administration.

Ports

Input

- — Input variables (variables) — optional port. The variables names can be used in *Command line*, namely, macro replacement of commands is supported: `%VariableName%` means the variable that contains the command that is supplied to the input of this node (refer to Examples).
- — Control variables (variables) — optional port. They can be used to set such parameters as *Node execution timeout* and *Terminate program by timeout*.

Output

- — Source variables and variables that contain information on the program execution success:
  - 12 **Execution result**: the integer variable that contains *0* in the case of the successful completion of the executed program;
  - ab **Error message**: the string variable that contains an error message if it occurred while execution or preparation to the program execution.

Node Wizard

- **Node execution timeout (ms)** enables to set the time of the node execution:
  - **0**: the node will be completed at once.

    - **1 and more**: number of milliseconds upon expiration of which the node will not be executed anymore. Arrows enable to increase or decrease the timeout at 100 ms interval.
    - **Infinity**: in this case the node will be executed until the program is closed. To set the node execution time, it is required to enter *-1* in the timeout field in the *Infinity* value, or it is required to delete all digits.
- **Terminate program by timeout**: the switch enables/disables the node execution timeout.

> **Important:** The time of node execution is set in the settings (not the time of the program execution) that is why if checkbox is not selected for *Terminate program by timeout*, the node execution will be terminated upon expiration of some time period set by the timeout, whereas the program execution will not be terminated.

Writing Commands

It is required to enter the command in the *Command text* field as follows:

- If it is a standard Windows program:
    - `cmd.exe` : the program name and its extension are specified.
- If it is a third-party program:
    - `C:\Program Files (x86)\Microsoft Office\Office14\EXCEL.EXE` : the absolute path to the executed file is specified.

To run commands using *cmd.exe*, it is required to state the following keys:

- `/C` : execution of the given command with subsequent completion;
- `/K` : execution of the given command without subsequent completion.

Other start keys are available on corresponding sites related to the Windows command line.

Current Project Directory

If the project has been saved, the current package directory is considered to be the current directory for the process started by the *Program execution* node.

> **Note:** The derived node is operated in the context of the package in which it is located as any other node. Thus, the current directory of the derived node will be the current package directory in which exactly the derived node is located but not the base one.

Desktop Megaladata Version

If the project is not saved, the current directory matches the current Megaladata directory in the Desktop version.

Server Megaladata Version

As far as the Megaladata server is concerned, if the project is not saved, the current directory matches the user directory.

Examples:

Example of the directory created via the command line:

```
cmd.exe /C mkdir D:\Dir\
```

Example of macro substitution:

```
cmd.exe /C %VariableName%
```

Where `%VariableName%` : the variable that contains any command to be executed on the Windows command line.