

.INI file syntax

Preference settings are maintained in an `.ini` file whose name and location are identified when starting VA Smalltalk (either explicitly or by default). The general format of the `.ini` file content is repeating groups of *stanza names* followed by *keyword=value* pairs corresponding to the settings associated with the preceding stanza name. Each line in the `.ini` file can have a maximum of 1023 characters. Comments (both full line and remainder-of-line) are also supported. Comments begin with a semi-colon (;) and are terminated at the end of the current line. So, for example, a segment of an `.INI` file would look like this:

```
[EmLibraryInterface]
; defaultName --> This is the full path (including the filename itself) of
the manager
; library. The path must be one reachable by the manager server's file
system.
; serverAddress --> This is the host name (or dotted address) of the server
machine that
; provides access to the manager library. Leave this blank for FileI\O.
DefaultName=d:\vamgr\manager\vavm020304_dev.dat
ServerAddress=192.168.1.101
OpenReadOnly = false ; This is the default value, but we put it here anyway
```

The [Preference Settings Framework](#) introduces a formalized syntax for the actual content of the `.ini` file:

The *stanza names* and *keywords* are case sensitive on all platforms; the *values* are case-preserving on Windows and case-sensitive on all Unix platforms.

The following simple types for *value* are supported:

boolean: value can be either *true* or *false*

decimal: value can be *integer.positiveInteger*

directory: value can be a string representing a file system path; a platform-specific trailing directory separator will be added to the string if it doesn't already end with a directory separator character (no other syntax checking is done)

file: value can be a string representing a file system filename with optional path information; currently handled the same as *string*

fraction: value can be the sequence *integer/positiveInteger*

integer: value can be a sequence of digits (0 through 9) with an optional leading minus (-)

multilineString: a string that may contain end-of-line indicators (\); currently handled the same as *string*

number: value can be *decimal* or *fraction* or *integer*

point: value can be *number@number*

positiveInteger: value can be a sequence of digits (0 through 9)

string: value can be any sequence of adjacent characters except for the comment delimiter (;), array delimiter (,) and leading/trailing white space; to handle these exceptions, you can surround the sequence with single quote characters (and double any embedded single quote characters)

The following complex types for *value* are supported:

array: value can be an array of simple values separated by the array delimiter (,) character. **Note:** This is a change from the previous way of specifying an array using the comment delimiter (;) character to separate the elements.

range: value can be a *number* falling within the specified range of numbers; the specification

supports including or excluding the end-points of the range.

Not all preference settings can be handled using this syntax, and not all preference settings can be processed using the [Preference Settings Framework](#). This means that there is room for expanding the syntax in the future to account for more complex settings. But some settings simply cannot be handled no matter what changes are made to the syntax rules. There are settings that are handled by the Smalltalk Virtual Machine (see [Virtual Machine options stanza](#)), settings that must be process at a different time than the standard settings (see [NLS and Locale stanzas](#)), and settings whose processing hasn't been converted to use the Preference Settings Framework yet. Still, wherever possible, the syntax described here is used for all those settings.

Copyright 2005, 2017 Instantiations, Inc. All rights reserved.

