

Supervisor & Viewer

Version 3.07



m Mobile Shell, Supervisor & Viewer, Version 3.07
Written by Lukas Knecht

www.m-shell.net

Document AB-M-SUP-887

© 2004-2011 airbit AG, 8008 Zürich, Switzerland

The information contained herein is the property of airbit AG and shall neither be reproduced in whole or in part without prior written approval from airbit AG. All rights are reserved, whether the whole or part of the material is concerned, specifically those of translation, reprinting, reuse of illustration, broadcasting, reproduction by photocopying machine or similar means and storage in data banks. airbit AG reserves the right to make changes, without notice, to the contents contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the material as presented.

Typeset in Switzerland.

Contents

1 Introduction **3**
1.1 Installation 3

2 The m Supervisor **5**
2.1 The Supervisor Dialog 5
2.2 SMS Control 7
2.3 Supervisor Properties File 9

3 The m Viewer **11**

Index **13**

© 2011 airbit AG

1. Introduction

The **m** Supervisor & Viewer is an additional component for the **m** application which must be installed separately. Its function is twofold:

- The ''supervisor'' is a separate background process which allows to start **m** scripts and executables at device startup, and to control **m** via SMS.
- The ''viewer'' adds automatic recognition of **m** script and executable files. This makes it particularly easy to add **m** scripts or programs to the **m** application by simply receiving them e.g. via Bluetooth®.

Note that on Symbian 3rd/5th Edition devices, the **m** Supervisor & Viewer requires you signing the installation package (online or offline via DevCert). But it works fine with a self signed **m** application.

1.1 Installation

The installation file for your device is part of the normal **m** shell distribution. You can always download the most recent files from www.m-shell.net, i.e. the Windows® `.exe` or the `.zip` file.

We recommend installing the **m** Supervisor & Viewer on the same storage device as the **m** application.

Currently, there are four supported device types. For the two 2nd edition device types, there are just two installation files:

S60 2nd edition	mViewer-S60-2nd.sis
UIQ2	mViewer-UIQ2.sis

For Symbian 3rd/5th Edition devices (Symbian 9 and higher), there are two variants due to platform security:

	Online signable	DevCert signable
S60 3rd Edition	mViewer-S60-3rd-OS.sis	mViewer-S60-3rd-DC.sis
S60 5th Edition	mViewer-S60-5th-OS.sis	mViewer-S60-5th-DC.sis
UIQ3	mViewer-UIQ3-OS.sis	mViewer-UIQ3-DC.sis

The two variants are (see also 5.2 (Reference, p. 68) for details):

- The ''Online signable'' packages must be signed online on www.symbiansigned.com before installation.
- The ''DevCert signable'' packages must be signed with your own Symbian Developer Certificate before installation.

The capabilities required when signing are: `NetworkServices`, `PowerMgmt`, `ProtServ`, `ReadDeviceData`, `ReadUserData`, `SwEvent`, `WriteDeviceData`, `WriteUserData`.

Due to the functionality it provides, the **m** Supervisor & Viewer cannot be offered as a self signed package.

2. The m Supervisor

The supervisor can start **m** (and the ''auto go'' scripts) automatically whenever the device is turned on, can even make sure that **m** is always running, and allows to control **m** remotely via SMS.

It is the **m** philosophy to make developing and running powerful applications on your phone as easy and straightforward as possible. However, this extra control has its price: for instance, the combination of automatic startup and ill behaving **m** script can make a device very difficult to use or completely unusable. Please take this into account when enabling the supervisor.



We also recommend installing **m** or at least the **m** Supervisor & Viewer on the removable memory card, so it can be disabled ''mechanically'' by removing the memory card from the device.

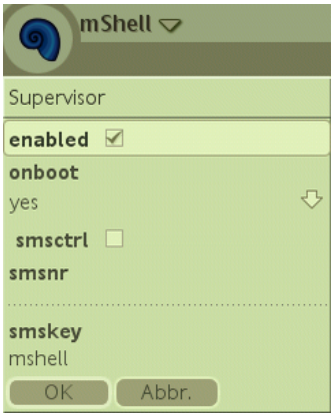


2.1 The Supervisor Dialog

The supervisor dialog is accessed via **View→Supervisor** from the **m** application. The supervisor properties determine the startup behaviour of **m** and the SMS control parameters (see also section 2.2 (p. 7)).



Series 60 sample screen



UIQ sample screen

The individual properties are:

`enabled` Select whether the supervisor should be enabled at all. Disable it if you don't like the idea of a supervisor process running in the background. Note that reenabling the supervisor may require rebooting the device to actually restart the process.

`onboot` Select whether the **m** application starts automatically when the phone is turned on. For testing, set to `once` to only autostart once. To automatically restart **m** whenever it exits or crashes, select `restart`.

`smsctrl` Select whether the **m** application can be controlled via SMS commands.

`smskey` The keyword prefix for all SMS commands. SMS not starting with this keyword are ignored and end up in the normal inbox.

`smsnr` The last digits of the sender phone number which can control the **m** application via SMS. For instance, with `smsnr=1234`, the SMS from all phones with a number ending in `1234` can control **m**. If empty, anybody knowing `smskey` can control **m** and eventually the device.



See also 2.3 (p. 9).

2.2 SMS Control

The supervisor allows to control the **m** application can be controlled via SMS commands. SMS control must be enabled, and commands must be prefixed by the *`smskey`* configured in the supervisor properties.

The available SMS commands are:

- *`smskey`* `reboot`: immediately reboot the device. This can be used to reset the device if it stops working properly. Note that the compatibility restrictions mentioned in 2.7 (Library, p. 62) apply.
- *`smskey`* `run` *`script args`*: starts the **m** application if it is not already running, then starts the script *`script`* with the arguments *`args`*. Use function `proc.args` to get the arguments from within the script. If the script is already running, this command is ignored.

- *smskey* `shutdown`: stops all scripts and exits the **m** application. If **m** is not running, this command is ignored.
- *smskey* `start`: starts the **m** application. If **m** is already running, this command is ignored.
- *smskey* `status`: **m** status inquiry, replies with an SMS describing the status of the **m** application and some GSM information. If **m** is running, the reply will look like:

```
m status: running, mem=mem,
  net=mcc,mnc, loc=lac,cid, sig=signal
```

  If **m** is not running, the reply will look like:

```
m status: NOT running (category reason),
  net=mcc,mnc, loc=lac,cid, sig=signal
```

  The meaning of the fields is the following:

`mem`	bytes of memory used by **m** (only on Symbian 2nd Edition)
`category`	**m** exit category (if panicked)
`reason`	**m** exit reason (if panicked)
`mcc`	GSM mobile country code
`mnc`	GSM mobile network code
`lac`	GSM location area code
`cid`	GSM cell id
`signal`	GSM signal strength

- *smskey* `status` *phone*: like `status` above, but the response is sent to phone number *phone*. *phone* must not contain white space.
- *smskey* `stop` *script*: stops execution of script *script*. If *script* is not running, this command is ignored.

The following examples require the `smsctrl` property to be enabled, and `smskey` to be set to `mshell`:

1. SMS to start the **m** application:

```
mshell start
```

2. SMS to start the `MySupervisor` script, passing it `0769988776` as an argument:

```
mshell run MySupervisor 0769988776
```

3. SMS to check the status of the **m** application:

```
mshell status
→ m status: NOT running (E32USER-CBase 71),
   net=228,115, loc=1616,17689, sig=3
```

m is not running because it crashed with a E32USER-CBase 71 panic. The phone is somewhere near cell 17689 in area 1616 of the Swisscom GSM network.

2.3 Supervisor Properties File

The behaviour of the **m** Supervisor is configured in the supervisor properties. Selecting **View→Supervisor** opens a dialog to edit them.

The properties are stored in an 8-bit text file with the same format as the **m** property file described in the previous section. The file resides in `"c:\iwlib\mSupervisor.prp"` or `"c:\shared\iwlib\mSupervisor.prp"` (on UIQ3 devices).

The following keys are recognized by the supervisor:

- `enabled=true|yes|y|1 | false|no|n|0`
  If set to `true`, `yes`, `y` or `1`, the supervisor background process will be created and persistent as long as the device is on.

- `onboot=true|yes|y|1 | false|no|n|0 | once | restart`
  If set to `true`, `yes`, `y` or `1`, the **m** application will be started automatically when the phone is booted up, i.e. switched on.
  If set to `once`, **m** is only started at the next bootup, as the entry is automatically set to `n` afterwards. This is the recommended setting for disaster prevention during script testing.
  If set to `restart`, **m** is started automatically when the phone is booted up, and restarted each time about 20 seconds after it exits (orderly or because of a crash).

- `smsctrl=true|yes|y|1 | false|no|n|0`
  If set to `true`, `yes`, `y` or `1`, the **m** application can be controlled via SMS commands, even if it is not running. See chapter 2.2 (p. 7).

- `smskey=keyword`
  Any SMS containing *`keyword`* as the first characters (ignoring case) is considered a command and sent to the **m** application.



- `smsnr=suffix`
  The last digits of the sender phone number which can control the **m** application via SMS. If empty, anybody knowing `smskey` can control **m**.

A sample supervisor properties file might look as follows:

```
enabled=yes
onboot=once
smsctrl=yes
smsnr=4561234
smskey=mshell
```

3. The m Viewer

The **m** viewer automatically recognizes **m** scripts (`.m` files), **m** modules (`.mm` files), and **m** executables (`.mex` files). For instance, an **m** script received via Bluetooth® can be added directly to the **m** application.



Series 60 sample screen



UIQ sample screen

Index

.m, 11
.mex, 11
.mm, 11
.prp file, 9

enabled, Supervisor property, 7

installation file, 3

onboot, Supervisor property, 7

recognition, 3

SMS control, 7
smsctrl, Supervisor property, 7
smskey, Supervisor property, 7
smsnr, Supervisor property, 7
Supervisor, 7
    properties file, 9
supervisor, 5