



Identify what matters

Evolis SDK

API MODE REFERENCE GUIDE

(Evolis Services Provider and Evolis Services provider 2)

Contents

Evolis Services Provider Framework Concept	4
Overview of the services.....	4
Quick Description of the services.....	5
Request model.....	5
Services – Technical description.....	8
CMD SERVICE.....	8
CMD.SendCommand.....	8
CMD.GetStatus.....	10
Retrieve the binary status of a device	10
CMD.ResetCom	11
PRINT SERVICE.....	12
PRINT.Begin.....	13
PRINT.Set	14
PRINT.SetBitmap.....	16
PRINT.Print.....	17
PRINT.GetJobID.....	18
PRINT.End	19
SETTING SERVICE	20
SETTING.Begin	21
SETTINGS.Import	22
SETTING.Get.....	23
SETTING.Set.....	24
Evolis Premium Suite 2 :Evosettings to manage printer's settings.....	24
SETTING.Export.....	26
SETTING.End.....	27
SUPERVISION SERVICE.....	28
SUPERVISION.List	28
Supervision.AddDevice	30
SUPERVISION.RemoveDevice.....	30
SUPERVISION.GetState	30
SUPERVISION.GetEvent.....	32
SUPERVISION.SetEvent.....	33
ECHO SERVICE	35
ECHO.Echo	35

ADD-ON SERVICE.....	36
ADDON.Launch	36
ESPF SERVICE	37
ESPF.GetParam	37
ESPF.SetParam	38
SCANNER SERVICE	39
SCAN.Version.....	39
SCAN.Acquire.....	40
SCAN.GetSession	42
SCAN.Begin	43
SCAN.End	44
SCAN.GetImages	45
SCAN.SendCommand	46
APPENDIX	48
ESPF SERVER CONFIGURATION	48
Configuration file available settings:.....	49
RETURNED ERRORS	51
Major state.....	56
Minor state	56
Disclaimer.....	61

Evolis Services Provider Framework Concept

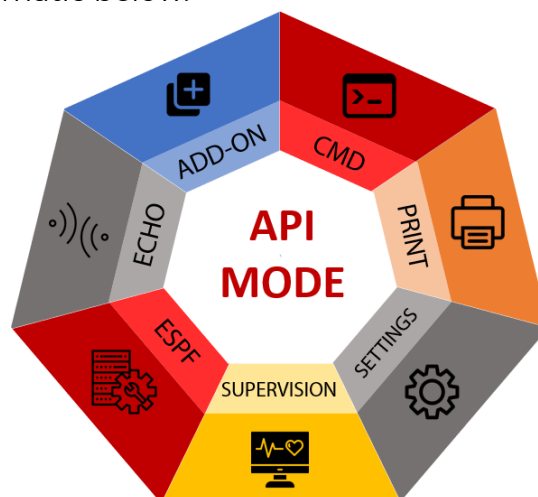
The **Evolis Services Provider Framework** and **Evolis Services Provider Framework 2** (based on Evolis Premium Suite 2) is a server and the gateway to the Evolis Suite (Evolis, Edikio, Badgy Premium Suite) features (supervision or standard mode) using requests.

All communications leverage IPC standard protocols (Socket or Named Pipe).

The request process is based on the JSON-RPC protocol, which eases the enablement of clients as described in the next sections.








Overview of the services

The Evolis Services Provider (API mode) is currently composed of 7 services, described in the schematic below:



- For each service, features could be accessed by calling a set of methods.
- Selected services require implementation of a specific call sequence.

Quick Description of the services

SERVICE NAME		PURPOSE
CMD		Bi-directional communication with the printer
Print		Direct Printing of a card
Setting		Printer configuration
Supervision		Printer's status management
Echo		Test service
ESPF		Server configuration
ADD-ON		Management of ADD-ON execution Ex : a third-party software for smart encoding

Request model

The request model used between the clients and the ESPF service leverages [JSON-RPC v2.0](#), a light-weight and stateless RPC (Remote Procedure Call) protocol using the *JSON* format.

Notification and Batch concepts of JSON-RPC v2.0 are not supported for the ESPF. The protocol is used in compliance with the following specifications:

- The **params** member is an object which member names matching the parameters expected by the server. Any value is a *String*.
- The **id** member is a *String*. For a response to a processed request, but with an unknown *id*, the *id* value is considered as *Null*.
- The **result** member is a *String*
- The **data** member is not considered



For **Primacy 2 available settings**, please refer to [this section](#)



Pay attention that the application software development made with Premium SDK are fully compliant with Primacy 2 model but could require modifications in case of new Primacy2 features (scan, rewrite...).

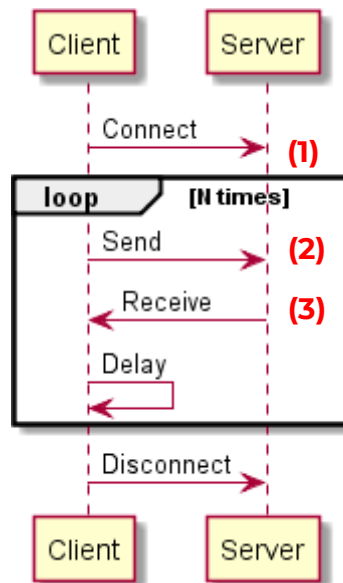
To send a request, client (software) sends a *Requests* object and receives a reply from the server in the form of a *Response* object

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "CMD.SendCommand", "params": { "command": "Rfv", "timeout": "5000", "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "2234" }</pre>



The general syntax to call a method in a JSON request is "SERVICE" Method" (method uses CamelCase), for example CMD.SendCommand (Use the **S**end **C**ommand method of the **CMD** service).

This implementation, at the communication protocol level, must be carried out as shown below:



- (1) During the connection, the IP address, server port, or Named pipe are identified according to the IPC communication protocol used.
- (2) Request is sent through a call on a full buffer(no data sent while buffer is sliced)
- (3) Response to the request is expected to be a single reply



If an unexpected error happens on the server while communicating with a connected client, server will shut down communication channel with this client. If a client disconnects from the server, it will stop any in progress process.

When sending a request, if the reception of the request takes longer than a configurable timeout (3,000 ms by default), the server will shut the communication channel with the client (support for « *half-open connections* » issues).

Services – Technical description

CMD SERVICE

- Sends commands in text or binary format
- Receives statuses in binary format
- Reset communication

CMD.SendCommand	Send commands in text or binary format
Command	Text or binary command
timeout	Timeout in milliseconds
device	Device name
result	Answer to the sent command
error	Not acknowledged

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "CMD.SendCommand",
  "params": {
    "command": "Rfv",
    "timeout": "5000",
    "device": "Evolis Primacy"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "2234"
}
```




To send a binary command, data must be [base64](#)-encoded. The command parameter is made up of the encoded data complemented by the **"base64:"** prefix

e.g., a base64-encoded Rfv command:

REQUEST OBJECT
<pre>{ { "id": "1", "jsonrpc": "2.0", "method": "CMD.SendCommand", "params": { "device": "Evolis Primacy", "command": "base64:G1Jmdg0=", "timeout": "3000" } } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "2234" }</pre>

If the response timeout is exceeded, the returned response will be an error with related error code.



If the timeout setting is not indicated, default value will be affected. This value is 3000 ms for the service. This parameter is not effective for a networked printer (Ethernet or WIFI).

CMD.ResetCom	Reset communication with a device
Timeout	Timeout in milliseconds
Device	Device name
Result	Ok if successful
Error	Error code

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "CMD.ResetCom", "params": { "timeout": "30000", "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

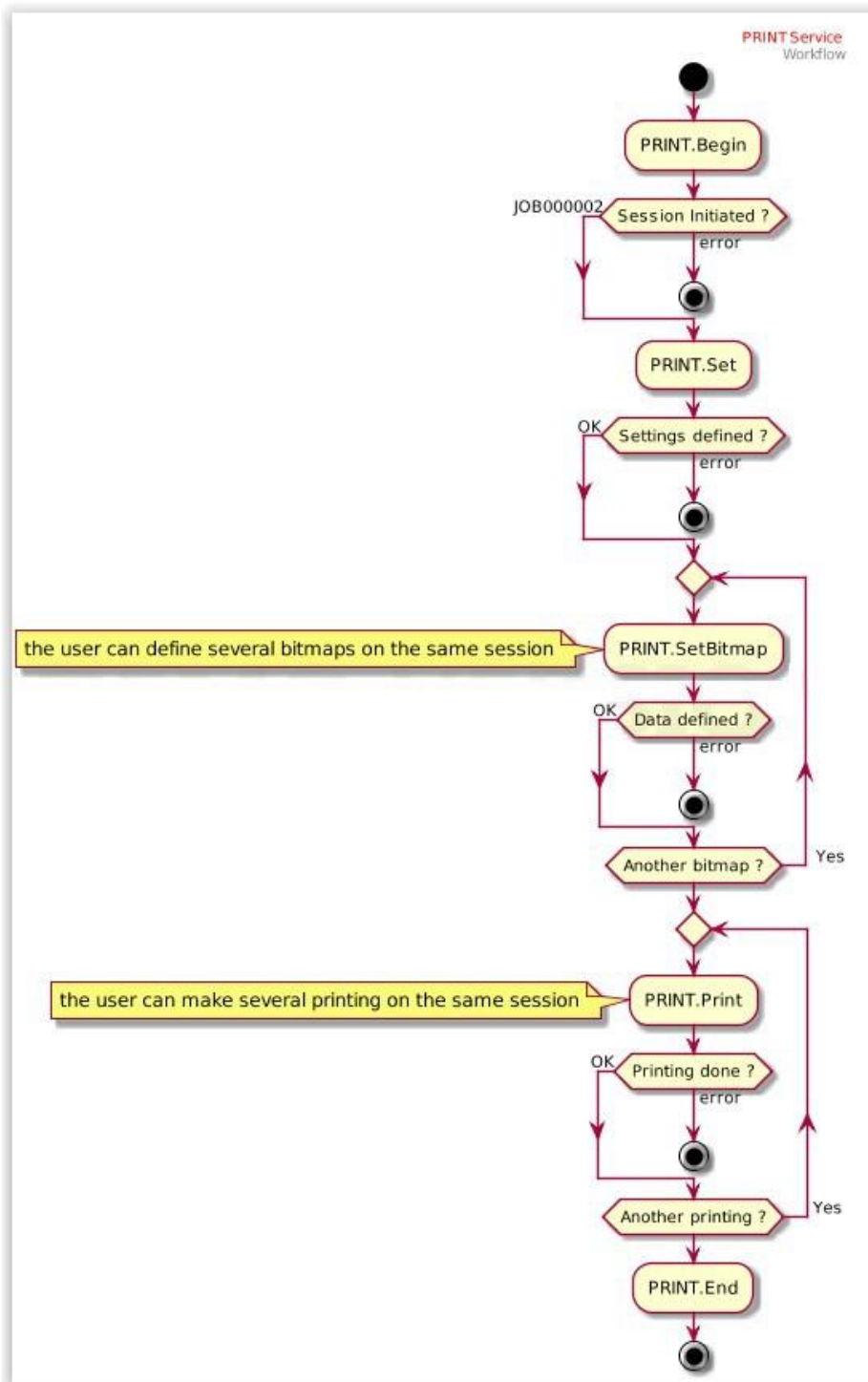


When response timeout is exceeded, returned response will be an error with related error code.

If the time out setting is not indicated, a default value will be provided. This value is of 3000 ms for the service. This parameter is not considered when the device is networked.

PRINT SERVICE

Prints a card, whatever the printing system
Configure the printings jobs



PRINT.Begin	Initiates the printing session
Device	device name
Session	optional : to set in case the automatic job ID is not adapted to your use-case
Result	Job ID is successful
Error	Error code in case of fail

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "PRINT.Begin",
  "params": {
    "device": "Evolis Primacy"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0", "result":
  "JOB000002"
}
```

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "PRINT.Begin",
  "params": {
    "device": "Evolis Primacy",
    "session": "CUSTOMJOBID"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "CUSTOMJOBID"
}
```



Printing session: a timeframe during which a printer is assigned to one or several printing jobs specified in the following format if generated automatically: JOBXXXXXX.

The job ID is not unique and is referenced in the printing logs.

The service can manage only one printing session (job) at a time per printer instance.

PRINT.Set	Set the printing parameters
session	JOB ID
data	printing parameters in the following format : key1=value1; ...; keyN=valueN
result	OK is successful
error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.Set", "params": { "session": "JOB000002", "data": "FColorBrightness=VAL12;GRibbonType=RC_YMCKO" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

The settings.xml file describes all printing parameters and their possible values for each printer model. This file is available from:

`%EPS_DIR%\Evolis Premium Suite\Model\{printer model}`

%EPS_DIR% is either the installation directory for the suite or an environment variable. To read the XML file, open it with an appropriate Web browser.

For Evolis Premium Suite 2, please refer to next page.



PRIMACY 2: Available settings for Evolis Premium Suite release above to version 7.0.0.0

To get the available settings and their possible values with newest Evolis Premium Suite, use the following command line :

```
evosettings.exe -p "Evolis Primacy 2" -lk
```

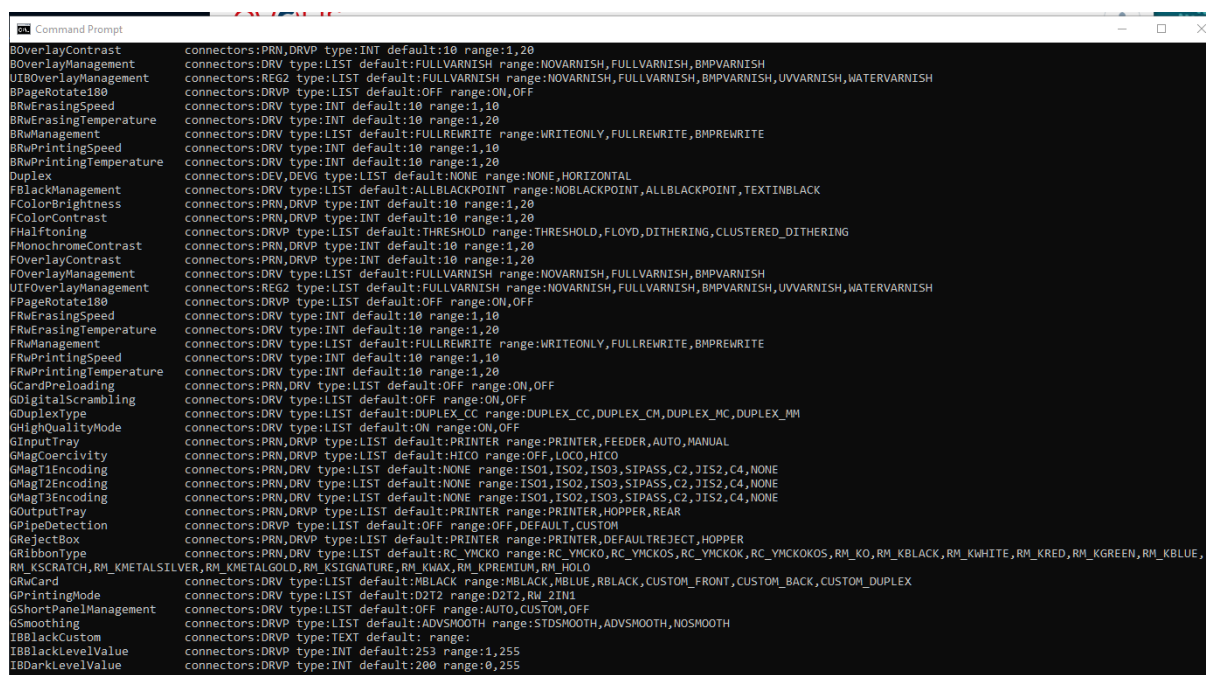
or

```
evosettings.exe -p "Evolis Primacy 2" --list-key
```

Replace "Evolis Primacy 2" by the correct name as in printer and devices.

The evosettings utility is in "...\\Installation Directory selected during setup\\Evolis Card Printer 2\\Evolis Premium Suite 2\\tools"

Below a sample screenshot :



```

Command Prompt
C:\Program Files\Evolis\Evolis Premium Suite 2\tools>evosettings.exe -p "Evolis Primacy 2" -lk

BOOverlayContrast connectors:PRN,DRV type:INT default:10 range:1,20
BOOverlayManagement connectors:DRV type:LIST default:FULLVARNISH range:NOVARNISH, FULLVARNISH, BMPVARNISH
UIBOOverlayManagement connectors:REG2 type:LIST default:FULLVARNISH range:NOVARNISH, FULLVARNISH, BMPVARNISH, UVVARNISH, WATERVARNISH
BPaperRotate180 connectors:DRV type:LIST default:OFF range:ON,OFF
BRWErasingSpeed connectors:DRV type:INT default:10 range:1,20
BRWErasingTemperature connectors:DRV type:INT default:10 range:1,20
BRWManagement connectors:DRV type:LIST default:FULLWRITE range:WRITEONLY, FULLWRITE, BMPWRITE
BRWPrintingSpeed connectors:DRV type:INT default:10 range:1,20
BRWPrintingTemperature connectors:DRV type:INT default:10 range:1,20
Duplex connectors:DEV,DEVG type:LIST default:NONE range:NONE, HORIZONTAL
FBlackManagement connectors:DRV type:LIST default:ALLBLACKPOINT range:NOBLACKPOINT, ALLBLACKPOINT, TEXTINBLACK
FColorBrightness connectors:PRN,DRV type:INT default:10 range:1,20
FColorContrast connectors:PRN,DRV type:INT default:10 range:1,20
FHaltToning connectors:DRV type:LIST default:THRESHOLD range:THRESHOLD, FLOYD, DITHERING, CLUSTERED_DITHERING
FMonochromeContrast connectors:PRN,DRV type:INT default:10 range:1,20
FOOverlayContrast connectors:PRN,DRV type:INT default:10 range:1,20
FOOverlayManagement connectors:DRV type:LIST default:FULLVARNISH range:NOVARNISH, FULLVARNISH, BMPVARNISH
UIFOOverlayManagement connectors:REG2 type:LIST default:FULLVARNISH range:NOVARNISH, FULLVARNISH, BMPVARNISH, UVVARNISH, WATERVARNISH
FPaperRotate180 connectors:DRV type:LIST default:OFF range:ON,OFF
FRWErasingSpeed connectors:DRV type:INT default:10 range:1,20
FRWErasingTemperature connectors:DRV type:INT default:10 range:1,20
FRWManagement connectors:DRV type:LIST default:FULLWRITE range:WRITEONLY, FULLWRITE, BMPWRITE
FRWPrintingSpeed connectors:DRV type:INT default:10 range:1,20
FRWPrintingTemperature connectors:DRV type:INT default:10 range:1,20
GCardPreloading connectors:PRN,DRV type:LIST default:OFF range:ON,OFF
GDigitalScrambling connectors:DRV type:LIST default:OFF range:ON,OFF
GDuplexType connectors:DRV type:LIST default:DUPLEX_CC range:DUPLEX_CC, DUPLEX_CM, DUPLEX_MC, DUPLEX_MM
GHighQualityMode connectors:DRV type:LIST default:ON range:ON,OFF
GInputTray connectors:PRN,DRV type:LIST default:PRINTER range:PRINTER, FEEDER, AUTO, MANUAL
GMagCoercivity connectors:PRN,DRV type:LIST default:HICO range:OFF, LOCO, HICO
GMagT1Encoding connectors:PRN,DRV type:LIST default:NONE range:ISO1, ISO2, ISO3, SIPASS, C2, JIS2, C4, NONE
GMagT2Encoding connectors:PRN,DRV type:LIST default:NONE range:ISO1, ISO2, ISO3, SIPASS, C2, JIS2, C4, NONE
GMagT3Encoding connectors:PRN,DRV type:LIST default:NONE range:ISO1, ISO2, ISO3, SIPASS, C2, JIS2, C4, NONE
GOutputTray connectors:PRN,DRV type:LIST default:PRINTER range:PRINTER, HOPPER, REAR
GPipeDetection connectors:DRV type:LIST default:OFF range:OFF, DEFAULT, CUSTOM
GRejectBox connectors:PRN,DRV type:LIST default:PRINTER range:PRINTER, DEFAULT, REJECT, HOPPER
GRibbonType connectors:PRN,DRV type:LIST default:RC_YNCKO range:RC_YNCKO, RC_YNCKOS, RC_YNCKOK, RC_YNCKOKOS, RM_KO, RM_KBLACK, RM_KWHITE, RM_KRED, RM_KGREEN, RM_KBLUE,
RM_KSCRATCH, RM_KMETALSILVER, RM_KMETALGOLD, RM_KSIGNATURE, RM_KMAX, RM_KPREMIUM, RM_HOLO
GRWCard connectors:DRV type:LIST default:MBLACK range:MBLACK, MBLUE, RBLACK, CUSTOM_FRONT, CUSTOM_BACK, CUSTOM_DUPLEX
GPrintingMode connectors:DRV type:LIST default:D2T2 range:D2T2, RM_2IN1
GShortPanelManagement connectors:DRV type:LIST default:OFF range:AUTO, CUSTOM, OFF
GSmoothing connectors:DRV type:LIST default:ADVSMOOTH range:STDSMOOTH, ADVSMOOTH, NOSMOOTH
TBBlackCustom connectors:DRV type:TEXT default: range:
TBBlackLevelValue connectors:DRV type:INT default:253 range:1,255
TBDarkLevelValue connectors:DRV type:INT default:200 range:0,255
  
```

Evosettings utility has different options that are described when you run the command.

PRINT.SetBitmap	Defines the graphics data to be printed
Session	JOB ID
Face	front or back
Panel	color, resin or varnish
Data	Image data (bitmap) encoded in base64
Result	OK
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.SetBitmap", "params": { "session": "JOB000002", "face": "front", "panel": "color", "data": "base64:Qk12Ix4AAAAAADYAAAAo..." } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>



Method will be called only once per card side and panel type. Color panel type includes graphical data required for the three YMC panels.

If the method is called twice or three times for a side, color panel will be the primary source, while the black, and/or overlay panel will be set as optional.



For the **YMCKOO ribbon** (available only for Evolis Primacy 2 printer), set the **panel** property to **varnish** to use the second overlay.

Options are then :

- **FOverlaySecondManagement** (for the **F**ront of the card)
- **BOverlaySecondManagement** (for the **B**ack of the card)

With the following possibilities :

- NOVARNISH
- FULLVARNISH
- BMPVARNISH

In case **BMPVARNISH** is set, then a **PRINT.SetBitmap** is required to configure the used bitmap (2 level monochrome bitmap).

PRINT.Print	Run a print job
Session	JOB ID
Result	OK if successful
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.Print", "params": { "session": "JOB000002" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>



The printing traffic is monitored during the printing job. Feature is always enabled and not configurable.

When calling the PRINT method, the GetEvent method from the SUPERVISION service must be [polled](#) on a regular basis. If an event is identified, an action must be taken so that the print job can be finalized.



The print settings are verified only at the PRINT.print stage. Please check all settings to ensure they are valid and available for the specified printer in order to prevent errors.

Please check the settings.xml file that describes all printing parameters and their possible values for each printer model.

PRINT.GetJobID	Get current printing session JOB ID
device	device name
Result	current JOB ID if successful
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.GetJobID", "params": { "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "JOB000002" }</pre>

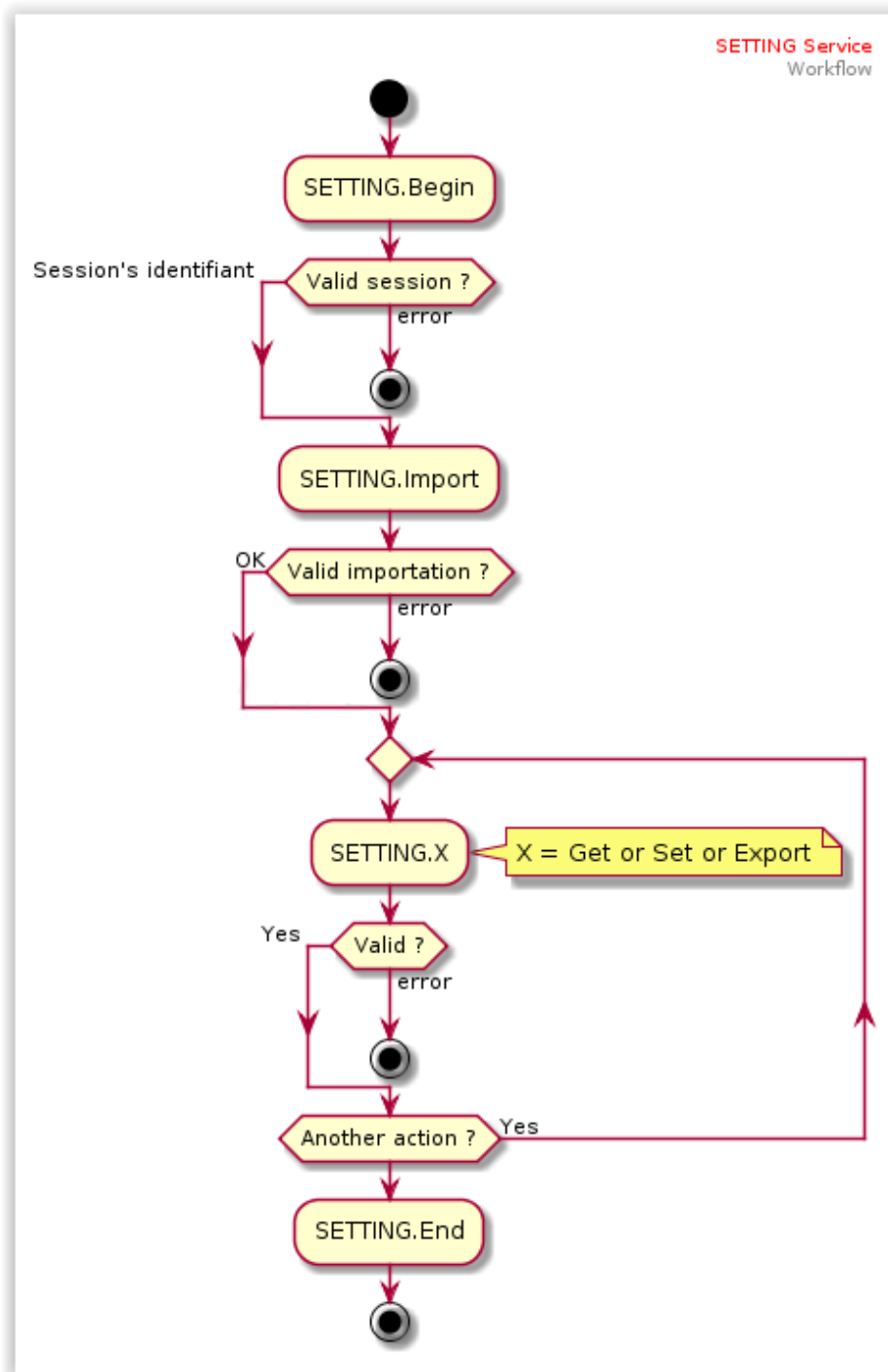
PRINT.End	Close a print session	Additional information
Session	JOB ID	to set only if you want to close the printing session by Job ID
Device	device name	to set only if you want to close the printing session by device name
Result	OK if successful	N/A
Error	error code and related message	N/A

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.End", "params": { "session": "JOB000002" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "PRINT.End", "params": { "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

SETTING SERVICE

Loads printing parameters
 Reads the loaded parameters
 Edits the loaded parameters
 Exports the printing parameters



SETTING.Begin	Start a configuration session
device	device name
Result	OK if successful
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SETTING.Begin", "params": { "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "SET000001" }</pre>

SETTINGS.Import	Import parameters
Session	Session ID
Format	printer
Data	Base64 encoded data (for XML format)
Result	OK if successful
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SETTING.Import",
  "params": {
    "session": "SET000001",
    "format": "printer"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "OK"
}
```



With this method, parameters can be imported via the active session, and in a non-persistent way (until the end of the configuration session). For further use, use the demo program in the API mode folder.

SETTING.Get	Gets the value of a parameter
Session	Session ID
Data	Key = parameter ID
Result	Parameter value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SETTING.Get",
  "parameters": {
    "session": "SET000001",
    "data": "FColorContrast"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "15"
}
```

SETTING.Set	Sets the value of a parameter
Session	Session ID
Data	Key=value ID of parameter with new value
Result	OK if successful
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SETTING.Set", "parameters": { "session": "SET000001", "data": "FColorContrast=18" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>



The settings.xml file describes all printing parameters and their possible values for each printer model. This file is available from:

`%EPS_DIR%\Evolis Premium Suite\Model\{printer model}`

`%EPS_DIR%` is either the installation directory for the suite or an environment variable.

To read the XML file, open it with an appropriate Web browser.

Evolis Premium Suite 2 :Evosettings to manage printer's settings

Primacy 2 works with the new **Evolis Premium Suite 2** (release superior to 7.0.0.0)

To get the available settings for Primacy 2, pay attention to use the new **evosettings.exe** tool (settings is no more available via the xml file).

This tool is available here (default installation path) once you have installed Evolis Premium Suite 2 :

`C:\Program Files\Evolis Card Printer 2\Evolis Premium Suite 2\tools\evosettings.exe`

The full help/manual of the evosettings utility is available by invoking **- - help** argument, see below :


```
PS C:\Program Files\Evolis Card Printer 2\Evolis Premium Suite 2\tools> ./evosettings.exe --help
VERSION: 7.0.0.0
Usage: C:\Program Files\Evolis Card Printer 2\Evolis Premium Suite 2\tools\evosettings.exe DEVICE [CONNECTORS] OPTIONS

Device:
  --printer,-p NAME      The printer to request.
  --driver,-drv NAME     Will request first printer found with driver NAME.
```

Get the current settings for a Primacy 2 thanks to evosettings

Use the following command to get the current setting for a printer:

evosettings -p "Evolis Primacy 2" -l
or
evosettings -p "Evolis Primacy 2" --list

```
Administrator: Windows PowerShell
PS C:\Program Files\Evolis Card Printer 2\Evolis Premium Suite 2\tools> ./evosettings -p "Evolis Primacy 2" -l
BBlackManagement = ALLBLACKPOINT
BColorBrightness = 10
BColorContrast = 10
BHalftoning = THRESHOLD
BMonochromeContrast = 10
BOverlayContrast = 10
BOverlayManagement = FULLVARNISH
UIBOverlayManagement = FULLVARNISH
BPageRotate180 = OFF
BRwErasingSpeed = 10
BRwErasingTemperature = 10
BRwManagement = FULLREWRITE
BRwPrintingSpeed = 10
BRwPrintingTemperature = 10
Duplex = NONE
FBlackManagement = ALLBLACKPOINT
```

Get all available settings for a Primacy 2 thanks to evosettings

To get all available settings and their possible values for a printer, please use following command :

Evosettings.exe -p "Evolis Primacy 2" --list-key

```
Administrator: Windows PowerShell
PS C:\Program Files\Evolis Card Printer 2\Evolis Premium Suite 2\tools> ./evosettings -p "Evolis Primacy 2" --list-key
BBlackManagement connectors:DRV type:LIST default:ALLBLACKPOINT range:NOBLACKPOINT,ALLBLACKPOINT,TEXTINBLACK
BColorBrightness connectors:PRN,DRV type:INT default:10 range:1,20
BColorContrast connectors:PRN,DRV type:INT default:10 range:1,20
BHalftoning connectors:DRV type:LIST default:THRESHOLD range:THRESHOLD,FLOYD,DITHERING,CLUSTERED_DITHERING
BMonochromeContrast connectors:PRN,DRV type:INT default:10 range:1,20
BOverlayContrast connectors:PRN,DRV type:INT default:10 range:1,20
BOverlayManagement connectors:DRV type:LIST default:FULLVARNISH range:NOVARNISH,FULLVARNISH,BMPVARNISH
UIBOverlayManagement connectors:REG2 type:LIST default:FULLVARNISH range:NOVARNISH,FULLVARNISH,BMPVARNISH,UVVARNISH,WATVARNISH
BPageRotate180 connectors:DRV type:LIST default:OFF range:ON,OFF
BRwErasingSpeed connectors:DRV type:INT default:10 range:1,20
BRwErasingTemperature connectors:DRV type:INT default:10 range:1,20
BRwManagement connectors:DRV type:LIST default:FULLREWRITE range:WRITEONLY,FULLREWRITE,BMPREWRITE
BRwPrintingSpeed connectors:DRV type:INT default:10 range:1,20
BRwPrintingTemperature connectors:DRV type:INT default:10 range:1,20
Duplex connectors:DEV,DEVG type:LIST default:NONE range:NONE,HORIZONTAL
FBlackManagement connectors:DRV type:LIST default:ALLBLACKPOINT range:NOBLACKPOINT,ALLBLACKPOINT,TEXTINBLACK
```

SETTING.Export	Exports parameters
Session	Session ID
Format	From printer to device text in text format xml to a data file (.dat) for the device
Result	if format=printer OK if successful if format=text list of parameters in the following format : key=value if format=xml export of base64-encoded configuration file
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SETTING.Export",
  "params": {
    "session": "SET000001",
    "format": "text"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "BBlackManagement=ALLBLACKPOINT; BColorBrightness=VAL10; BColorContrast=VAL10; BHalftoning=THRESHOLD; BMonochromeContrast=VAL10; BOverlayContrast=VAL10; BOverlayManagement=FULLVARNISH; BPageRotate180=OFF; FBlackManagement=ALLBLACKPOINT; FColorBrightness=VAL10; FColorContrast=VAL13; FHalftoning=THRESHOLD; FMonochromeContrast=VAL10; FOverlayContrast=VAL10; FOverlayManagement=FULLVARNISH"
}
```

SETTING.End	Ends the session
Session	Session ID
Result	OK if successful
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SETTING.End", "params": { "session": "SET000001" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", « result »: « OK » }</pre>

SUPERVISION SERVICE

Lists the subscribed devices and their state

Subscribe/unsubscribe a device to a service, with state notification

SUPERVISION.List	List all subscribed devices
Level	0 List of Stateless device(s) 1 List of device(s) with major state 2 List of device(s) with a major or a minor state
Result	printername (if level=0) printername,majorstate(if level=1) printername,majorstate,minorstate(if level=2)
error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SUPERVISION.List",
  "params": {
    « level » : « 2 »,
    « device » : « Evolis Primacy »
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "Primacy,READY,PRINTER_READY"
}
```



If several printers are "subscribed", response to the request will have the following format, respectively for level=2, level=1 and level=0.

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "Primacy,READY,PRINTER_READY;Primacy2,WARNING,FEEDER_EMPTY"
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": " Primacy,READY;Primacy2,WARNING"
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "Primacy,Primacy2"
}
```



If the level parameter is not specified, then level=0

The major states provide global information on the device.

The minor states offer more detailed information of the major states.

The status reported by the **SUPERVISION** service is the same status that are reported by the Print Center, before, after and during a print job

To check a specific status, outside of a print job, it is recommended to use the GetStatus job request thanks to the CMD Service (binary status) see : [evolIS sdk binary-status](#).

Supervision.AddDevice	Subscribes a new device to the notification service
Device	device name
Result	Ok if successful
error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SUPERVISION.AddDevice", "params": { "device": "Evolis Zenius" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

SUPERVISION.RemoveDevice	Unsubscribes a device to the notification service
Device	device name
Result	Ok if successful
error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SUPERVISION.RemoveDevice", "params": { "device": "Evolis Zenius" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>

SUPERVISION.GetState	Requests the state of a device
-----------------------------	---------------------------------------

Device	device name
Result	majorstate, minorstate
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SUPERVISION.GetState", "params": { "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "READY,PRINTER_READY" }</pre>



The states reported by the SUPERVISION service are the same states that are reported by the Print Center before, after and during a print job. If it is needed to test a specific status, for example not reported outside a print job by the Print Center, it is recommended to use the method GetStatus of the CMD service.

SUPERVISION.GetEvent	Returns the notification of an unexpected event, as well as the list of actions for a device
Device	device name
Result	NONE if no notification Minorstate, combination of actions
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SUPERVISION.GetEvent", "params": { "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "ERR_MECHANICAL:CANCEL,RETRY" }</pre>



An unexpected event is notified during printing by the PRINT service or by the spooler. This process is related to the pop-up notifications from the printer manager.

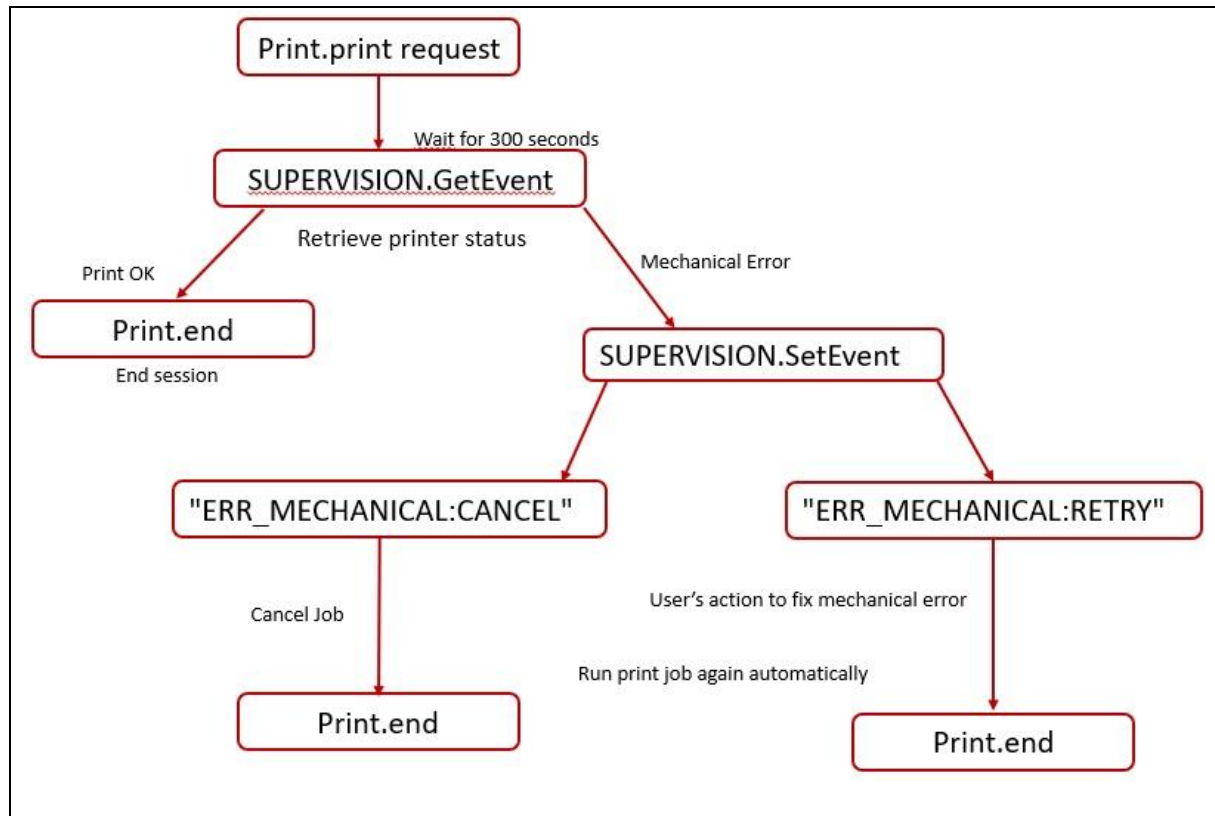
SUPERVISION.SetEvent	Executes an action when an unexpected event is notified on a device
Device	device name
Result	minorstate:action possible actions : CANCEL, OK, RETRY
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SUPERVISION.SetEvent", "params": { "action": "FEEDER_EMPTY:CANCEL", "device": "Evolis Primacy" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>



Sending an action when notified of an unexpected event is equivalent to acting on the Printer Manager's feedback notification. In this example, the feeder is empty and a request for cancelling the printing job is sent.

The main difference between the GetEvent action and SetEvent is illustrated in the following diagram:



The SUPERVISION.GetEvent returns the notification of an unexpected error while the SUPERVISION.SetEvent executes an action to correct that unexpected error.

ECHO SERVICE

Check that the ESPF is enabled.

ECHO.Echo	Check server's status
Result	String sent in request
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "ECHO.Echo",
  "params": {
    "data": "Hello World"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "Hello World"
}
```

ADD-ON SERVICE

Run a third-party add-on software (mostly for smart and contactless encoding)

ADDON.Launch	Run a third-party add-on software
Command	name of the application
Data	parameters of the application
Result	return code of the application
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "ADDON.Launch",
  "params": {
    "command": "testplugin.exe" "data": "testdata1 testdata2 testdata3"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "0"
}
```



The Launch method triggers a response as soon as the application is fully executed.

The application must be executed without involving the user. The ADD-ON service is disabled by default.

The folder hosting the add-ons can be configured in the ESPF service.

ESPF SERVICE

Read servers' parameters
Edit servers' parameters

ESPF.GetParam	Reads the value of a server parameter
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "ESPF.GetParam",
  "params": {
    "key": "ESPFServerManager.port"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "18000"
}
```

ESPF.SetParam	Edits the value of server parameter
key	Parameters name
data	new value to be set
result	OK if successful
error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "ESPF.SetParam", "params": { "key": "ESPFServerManager.port", "data": "18001" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "OK" }</pre>



The following parameters can be read:

- ESPFService.version
- ESPFService.requestlanguageversion:

Following parameters can be read/set:

- **ESPFServerManager.port:** port number to be used by clients for socket-based communication with the server.
- **ESPFTcpServerConnectionSupervisor.receiveTimeout:** timeout in ms before disconnection info data is received via a socket-based communication with a client.
- **ESPFServerManager.tcpEnabled:** the true value triggers the server's Socket layer; the false value disables it.

Other parameters can be edited only through the server's configuration file :

- For Evolis Premium Suite < 7.0.0.0 : ESPFsvc.properties file
- For Evolis Premium Suite > 7.0.0.0 (Evolis Premium Suite 2): evoservice.properties

The Socket layer parameters cannot be edited if the layer is enabled.

SCANNER SERVICE

Scanner service could be used two ways :

- *With a session*, meaning scanner is reserved the whole session lifetime. Scanned images could be retrieved in one or several times (see [SCAN.GetImages](#)).
- *Without a session* scanner is reserved only while the request execution, in case images are not get in the request, then it is lost. (see [SCAN.Acquire](#))

Only available with Evolis Primacy 2 printer.

Read scanner version

SCAN.Version	Reads the value of the scanner version
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SCAN.Version",
  "params": {
    "device": "Evolis Primacy 2"
  }
},
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "WiseCube WIS-1000 2.09.L702A (02:01) [2107-0002]"
}
```



This request allows to know the current scanner version installed. Only available with Primacy 2 and Evolis Premium Suite release above 7.0.0.0.

Acquire an image

SCAN.Acquire	Scan a card with specific parameters
key	Parameters name
Result	Parameters value
Error	error code and related message

Example without a session:

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SCAN.Acquire", "params": { "device": "Evolis Primacy 2", "options": "DPI=300, CARD_SPEED=220", "images": "TOP, TOP_IR, BOTTOM, BOTTOM_IR", "eject": false } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": { "images": { "BOTTOM": "base64:<data>", "BOTTOM_IR": "base64:<data>", "TOP": "base64:<data>", "TOP_IR": "base64:<data>" } } }</pre>

Example with a session:

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SCAN.Acquire", "params": { "session": "SCAN1", "options": "DPI=300, CARD_SPEED=220", "images": "TOP, TOP_IR, BOTTOM, BOTTOM_IR", "eject": false } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": { "images": { "BOTTOM": "base64:<data>", "BOTTOM_IR": "base64:<data>", "TOP": "base64:<data>", "TOP_IR": "base64:<data>" } } }</pre>



This request allows to scan a card and get the output as an image. If a card is inside the printer, it will be scanned, otherwise a card will be inserted.

Image is encoded in base64, it remains possible to get one or several images at a time.

Available images scan settings :

- **TOP:** Color image corresponding to the top of the card
- **TOP_IR :** IR image corresponding to the top of the card
- **BOTTOM :** Color image corresponding to the bottom of the card
- **BOTTOM_IR :** IR image corresponding to the bottom of the card

Optional parameters (one or several combinations are possible):

- **WIDTH :** image width in millimeters , default value :540mm par défaut (CR80 card).
- **HEIGHT :** image height in millimeters, default value 860mm (CR80 card).
- **DPI :** 600 or 300, default value 600. Evolis printer is calibrated to scan in 600 DPI resolution, in case scan is performed in 300 DPI resolution, it is required to adapt the CARD_SPEED value (e.g., 220).

- **CARD_OFFSET** : Synchronization of the scan with the card beginning edge. Factory setting. Value between 90 and 220.
- **CARD_LENGTH** : is related to the card's length, default value is 1200 (CR80 card).
- **CARD_SPEED** : is related to the DPI resolution. Factory settings for 600 DPI. Value between 150 and 350.

Get current scanner session

SCAN.GetSession	Reads the value of the current scanner session
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SCAN.GetSession",
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "<Current_Session_value>"
}
```



This request allows to know the current scanner session. Only available with Primacy 2 and Evolis Premium Suite release above 7.0.0.0. No required parameters.

Start a scanner session

SCAN.Begin	Start a scanner session
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SCAN.Begin", "params": { "device": "Evolis Primacy 2" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": "SCAN1" }</pre>



This request allows to start a scanner session. Only available with Primacy 2 and Evolis Premium Suite release above 7.0.0.0. Only one session at a time. The result is to use for the other requests of the same session.

End a scanner session

SCAN.End	End a scanner session
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SCAN.End",
  "params": {
    "device": "SCAN1"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": "OK"
}
```



This request allows to end a scanner session, to free the scanner. Only available with Primacy 2 and Evolis Premium Suite release above 7.0.0.0. Only one session at a time. The only parameter to use corresponds to the returned value by a previous call to "SCAN.Begin" (e.g.: SCAN1).

Get images from scanner

SCAN.GetImages	Get images from scanner
key	Parameters name
Result	Parameters value
Error	error code and related message

REQUEST OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "method": "SCAN.GetImages",
  "params": {
    "session": "SCAN1",
    "images": "TOP, TOP_IR, BOTTOM, BOTTOM_IR"
  }
}
```

RESPONSE OBJECT

```
{
  "id": "1",
  "jsonrpc": "2.0",
  "result": {
    "images": {
      "TOP": "base64:...",
      "BOTTOM": "base64:..."
    }
  }
}
```



This request allows to get the current scanner images, It only works in a session. This could get images from the previous SCAN.Acquire request. Only available with Primacy 2 and Evolis Premium Suite release above 7.0.0.0. Only one session at a time. The only parameter to use corresponds to the returned value by a previous call to "SCAN.Begin" (e.g.: SCAN1). Parameters are a list of image to get (see [SCAN.Acquire](#))

Send command to the scanner

SCAN.SendCommand	Send command to the scanner
key	Parameters name
Result	Parameters value
Error	error code and related message

Example without a session:

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SCAN.SendCommand", "params": { "device": "Evolis Primacy 2", "command": "#GV" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": { "code": 12, "data": "..." } }</pre>

Example with a session:

REQUEST OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "method": "SCAN.SendCommand", "params": { "session": "SCAN1", "command": "#GV" } }</pre>
RESPONSE OBJECT
<pre>{ "id": "1", "jsonrpc": "2.0", "result": { "code": 12, "data": "..." } }</pre>



This request allows to send a direct command to the scanner (work with or without a session). Returns will be base 64 encoded in case it contains binary data (with a "base64:" prefix). Regarding specific commands list, additional information only upon request.

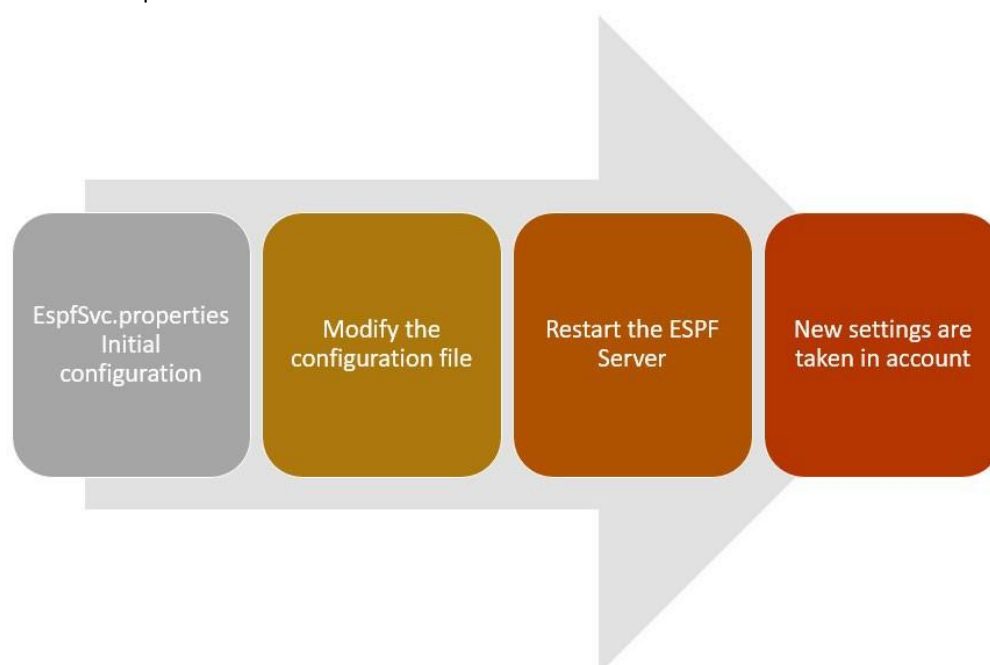
Parameters could be the following:

- **Command** : command to send, could be base 64 encoded
- **Device** : Printer's target name
- **Session** : Session ID, used only if device parameter is missing

APPENDIX

ESPF SERVER CONFIGURATION

The ESPF server can be configured using its configuration file. The server's settings are considered upon server boot.



The configuration file is in the server setup directory and is named ESPFSvc.properties (same for all the Evolis Suite as Badgy Premium Suite, Edikio Printer Suite).

This file hosts general server parameters, as well as parameters specific to services hosted on the server. The following tables describe a selection of such parameters.

For **Primacy 2**, and Evolis Premium Suite 2 please refer to this [section](#).

Configuration file available settings:

This section gathers the available settings for the Evolis Services Provider Framework for Evolis Suite (Evolis Premium Suite, Edikio Printer Suite, Badgy Printer Suite) up to version 6.

For Evolis Premium Suite release above 7.0.0.0 (Evolis Premium Suite 2), please refer to [this section](#)

Logs settings	
Name	Value
ESPFSservice.log	The true value enables the debugging logs (default). FALSE value disables it.
ESPFSservice.loglevel	Verbose level : 1 Fatal 2 Critical 3 Error (default) 4 Warning 5 Notice 6 Information 7 Debug
ESPFSservice.logrequest	TRUE value enables the requests log. FALSE value disables it (default).
ESPFSservice.logrequestoutputdir	Defines the directory for requests log files. (default value is Tmp)
ESPFSservice.isrelativeoutputdir	TRUE value indicates that the directory is defined as a relative path to the directory where the service is installed (default). FALSE value indicates an absolute path.

Named pipe transport layer settings	
Name	Value
ESPFSserverManager.serveraddress	Base name of the Named pipe channel to be used by the server's clients. Espf2Server00 for Primacy 2 (EPS > 7.0.0.0) EspfServer00 for Evolis Suite inferior to 7.0.0.0.
ESPFSserverManager.uniqueid	Supplementary name for the Named pipe channel to be used by the server's clients. (Name to use = base name + supplementary name) BADGY01 for Badgy 100 & 200 EDIKIO01 for Edikio printer Range

ESPFServerManager.disablepipeserver	TRUE value disables the Named pipe transport layer upon server boot. FALSE value enables it (default)
ESPFSservice.logrequestoutputdir	Defines the directory for requests log files. (default value is Tmp).
ESPFSservice.isrelativeoutputdir	TRUE value indicates that the directory is defined as a relative path to the directory where the service is installed (default). FALSE value indicates an absolute path.
Socket transport layer	
Name	Value
ESPFServerManager.port	States the port number to be used by the server's clients. Default port values : 18000 for the Evolis Premium Suite 18050 for the Badgy.Premium Suite 18100 For Edikio Printer Suite 18004 For Primacy 2 (EPS > version 7.0.0.0, relies on evoservice.properties file)
ESPFServerManager.enabletcpstart	Supplementary name for the Named pipe channel to be used by the server's clients. (Name to use = base name + supplementary name)

Parameters for the add-on service	
Name	Value
ServiceAddOnManager.enabled	TRUE value enables the services upon server boot. FALSE value disables it (default)
ServiceAddOnManager.addondir	Defines the directory for hosting the add-on applications that can be executed. (default value is Tmp\\SrvAddOn)
ServiceAddOnManager.isrelativeaddondir	TRUE value indicates that the directory is defined as a relative path to the directory where the service is installed (default). FALSE value indicates an absolute path.

RETURNED ERRORS

Error returned by the server has different states described in the below matrix, errors are distributed by service and a general category "All services".

All services

Code	-32700
Message	(E) Parse error
Description	Invalid JSON was received by the server. An error occurred on the server while parsing the JSON text

Code	-32600
Message	(E) Invalid Request
Description	The JSON sent is not a valid Request object

Code	-32601
Message	(E) Method not found
Description	The method does not exist / is not available

Code	-32602
Message	(E) Invalid params
Description	Invalid method parameter(s)

Code	-32603
Message	(E) Internal error
Description	Internal JSON-RPC Error

Code	-32000
Message	(E) Service not found
Description	The service does not exist / is not available

CMD service

Code	1200
Message (E)	Get status error
Description	Get status error

Code	1201
Message	(E) Send command error
Description	Send command error

Code	1202
Message	(E) Reset communication error
Description	Reset communication error

Code	1300
Message	(W) Session already reserved
Description	The communication session is already reserved

Code	1301
Message	(W) Timeout
Description	The send command timeouts

Code	1302
Message	(W) Device not supervised
Description	The target device is not supervised

Code	1303
Message	(W) No Status

PRINT service

Code	1600
Message (E)	Print error
Description	Print Error

Code	1700
Message (W)	Printing session already in progress for the device
Description	The device has already a printing session in progress

Code	1701
Message (W)	Invalid printing session.
Description	The session id sent is not a valid printing session id

Code	1702
Message	(W) Cannot perform action : action already in progress
Description	Cannot perform action: action already in progress

Code	1703
Message	(W) Cannot perform action : setting or bitmap missing
Description	Cannot perform action: setting or bitmap missing

Code	1704
Message	(W) Cannot perform action : processing error
Description	Cannot perform action: processing error

Code	1705
Message	(W) Cannot perform action : job canceled
Description	Cannot perform action: job canceled

SETTING service

Code	1400
Message	(E) Get error
Description	Get Error

Code	1401
Message	(E) Set error
Description	Set Error

Code	1402
Message	(E) Import error
Description	import Error

Code	1403
Message	(E) Export error
Description	Export Error

Code	1500
Message	(W) Invalid printing session.
Description	The session id sent is not a valid printing session id

Code	1501
Message	(W) Cannot perform action : action already in progress
Description	Cannot perform action: action already in progress

Code	1502
Message	(W) Cannot perform action : import not done
Description	(W) Cannot perform action : import not done

SUPERVISION service

Code	1502
Message	(W) Cannot perform action : import not done
Description	(W) Cannot perform action : import not done

Code	1000
Message	(E) Device supervision/un-supervision error
Description	The device cannot be supervised/un-supervised

Code	1001
Message	(E) Set event action error
Description	Set event action error

Code	1100
Message	(W) Device not found
Description	The device cannot be found

Code	1101
Message	(W) Device already supervised
Description	The device is already supervised

Code	1102
Message	(W) Device already un-supervised
Description	The device is already un-supervised

Code	1103
Message	(W) No event waiting for an action
Description	There is no event waiting for an action

Code	1104
Message	(W) Event action to set not found
Description	The event action to set does not exist / is not available

Code	1105
Message	(W) Different event waiting for an action
Description	The event waiting for an action is different

ADD-ON service

Code	1800
Message	(E) Launch error
Description	Launch error

ESPF service

Code	2000
Message	(E) Get parameter error
Description	Get parameter error

Code	2001
Message	(E) Set parameter error
Description	Set parameter error

Code	2100
Message	(W) The parameter key is invalid
Description	The parameter key is invalid

SCANNER service

Code	1900
Message	(E) No scanner in the printer
Description	No scanner in the printer

Code	1901
Message	(E) Scanner already in use by someone else
Description	Scanner already in use

Code	1902
Message	(E) Given session is invalid
Description	Session ID in the request is not valid

Code	1903
Message	(E) Invalid option value
Description	Issue during the scan

Code	1904
Message	(E) Invalid images value
Description	Images value is not in the right formatting or value is wrong

Code	1905
Message	(E) Error while scanning card
Description	An unexpected error occurred while scanning the card

PRINTERS' STATES

The printer states are those reported by the Print Center.

Major state

Designation	Description
OFF	Unable to communicate
ERROR	Printer error (requests a human intervention)
READY	Printer Ready
WARNING	Next printing unavailable (requests a human intervention)

A **WARNING** status is usually reported before the beginning of a printing but for some specific status. It can be reported only when the printing starts.

An **ERROR** status is only reported during printing.

Minor state



For **Evolis Premium Suite above 7.0.0.0 (= Evolis Premium Suite 2)**, some status changes in the answer of the request :

- **WARNING,FEEDER_EMPTY** becomes **WARNING,DEF_FEEDER_EMPTY**
- **ERROR,FEEDER_EMPTY** becomes **ERROR,ERR_FEEDER_EMPTY**
- **WARNING,RIBBON_ENDED** becomes **WARNING,DEF_RIBBON_ENDED**
- **ERROR,RIBBON_ENDED** becomes **ERROR,ERR_RIBBON_ENDED**
- **WARNING, LAMINATE_END** becomes **WARNING,DEF_LAMINATE_END**
- **ERROR, LAMINATE_END** becomes **ERROR,ERR_LAMINATE_END**

Designation	Description	Major State
PRINTER_NOT_SUPERVISED	Not supervised by Evolis Print Center	OFF
PRINTER_OFFLINE	Printer offline	OFF
PRINTER_STATUS_DISABLED	Status disabled - Printer on-line	OFF
ERR_BLANK_TRACK	Magnetic encoding failed Encoding fails on this card, please check the card position in the feeder.	ERROR
ERR_COVER_OPEN	Cover open error The cover was opened during the printing cycle. Close the cover and click on resume.	ERROR
ERR_FEATURE	Printing not supported Print cannot be run because it is not supported by the printer	ERROR
ERR_HEAD_TEMP	Too high temperature Print head temperature too high - wait until it cools down.	ERROR
ERR_HOPPER_FULL	Output Hopper Full Please remove all the printed cards from the output hopper to resume printing.	ERROR

ERR_MAGNETIC_DATA	Magnetic encoding failed Invalid data format.	ERROR
ERR_MECHANICAL	Mechanical error A mechanical error has occurred.	ERROR
ERR_READ_MAGNETIC	Magnetic encoding failed Magnetic track reading failed.	ERROR
ERR_REJECT_BOX_FULL	Reject Box Full Please remove all the cards from the reject hopper to resume printing.	ERROR
ERR_RIBBON_ERROR	Ribbon problem The ribbon is cut or stuck to the card.	ERROR
ERR_WRITE_MAGNETIC	Magnetic encoding failed Read-after-Write failure.	ERROR
FEEDER_EMPTY (ERR_FEEDER_ERROR)	Card feed problem Please check cards, position in the card feeder and gauge adjustment.	ERROR
INF_WRONG_ZONE_EXPIRED	Ribbon not valid Compatibility problem between ribbon and printouts credit limit reached. Please contact your reseller.	ERROR
RIBBON_ENDED (ERR)	Ribbon end Ribbon end, please replace by a new one.	ERROR
LAMINATE_END (ERR)	Film end Film end. Please replace the film.	ERROR
ERR_LAMINATE	Film problem Film problem. The film is cut or stuck to the card.	ERROR
ERR_LAMI_MECHANICAL	Mechanical error A mechanical error has occurred in the lamination module.	ERROR
ERR_LAMI_TEMPERATURE	Temperature error The lamination module encountered a temperature error.	ERROR
ERR_LAMI_COVER_OPEN	Door open during lamination The lamination module door got opened during the lamination process. Please close it and retry.	ERROR
INF_CLEANING	Regular cleaning required Printer cleaning required. (this state is reported only when the printing starts, every 50 cards approximatively)	READY
INF_CLEANING (INF_CLEAN_2ND_PASS)	Insert your adhesive cleaning card Please insert your sticky cleaning card. 'Cancel' if you want to proceed with printing. (this state is reported only when the printing starts, every 50 cards approximatively)	READY
INF_CLEANING_RUNNING	Cleaning in progress	READY
INF_ENCODING_RUNNING	Encoding in progress	READY
INF_PRINTING_RUNNING	Printing in progress	READY
INF_LAMINATING_RUNNING	Lamination in progress	READY

INF_LAMI_CLEANING_RUNNING	Lamination module cleaning in progress	READY
INF_LAMI_UPDATING_FIRMWARE	Lamination module firmware update in progress	READY
INF_SLEEP_MODE	Printer in Standby mode	READY
INF_UPDATING_FIRMWARE	Firmware update in progress	READY
NOT_FLIP_ACT	Single-sided Printer	READY
PRINTER_READY	Printer ready	READY
INF_RIBBON_LOW	Ribbon close to the end Ribbon close to the end, please proceed with replenishment. (this state is reported only when the printing starts, every 10 cards approximatively, and changes automatically after 10s)	WARNING
INF_FEEDER_NEAR_EMPTY	Feeder almost empty The card feeder is almost empty, please refill. (this state is reported only when the printing starts, every X cards approximatively, and changes automatically after 10s)	WARNING
BUSY	Printer busy You cannot print while the printer is busy. Please wait or click on "Cancel". (this state is reported only when the printing starts)	WARNING
CFG_FLIP (test if not raised)	Single-sided Printer Your single-sided printer cannot print your dual-sided design. (this state is reported only when the printing starts)	WARNING
CFG_MAGNETIC (test if not raised)	Magnetic coding option not installed To continue printing without magnetic coding click on resume. (this state is reported only when the printing starts)	WARNING
CFG_EXTENDED_RESOLUTION (test if not raised)	Incompatible parameter This resolution parameter is not compatible with this printer / ribbon. (this state is reported only when the printing starts)	WARNING
DEF_CARD_ON_EJECT	Remove card Remove the card from the manual feeder.	WARNING
DEF_COOLING	Cooling in progress Printer cooling in progress.	WARNING
DEF_POWER_SUPPLY	Power supply cable Please use the power cable that has been delivered with the printer in order to not damage the print quality and the printer.	WARNING
DEF_COVER_OPEN	Cover open Close your printer cover.	WARNING
DEF_HOPPER_FULL	Output Hopper Full	WARNING

	Please remove all the printed cards from the output hopper to resume printing.	
DEF_NO_RIBBON	No ribbon Replace the ribbon.	WARNING
DEF_PRINTER_LOCKED	Communication with the printer is locked Contact your dealer	WARNING
DEF_UNSUPPORTED_RIBBON	Ribbon incompatible with this printer model The ribbon inserted cannot work with this printer model.	WARNING
DEF_WAIT_CARD	Waiting for a card insertion Please insert your card manually.	WARNING
ERR_BAD_RIBBON	Ribbon installed is incompatible with settings The ribbon installed does not correspond to the manually defined settings. Printing cannot take place. (this state is reported only when the printing starts)	WARNING
FEEDER_EMPTY (DEF)	Card feed problem Please check cards, position in the card feeder and gauge adjustment.	WARNING
INF_CLEANING_ADVANCED	Advanced cleaning required Printer advanced cleaning is required. (this state is reported only when the printing starts, every 50 cards approximatively)	WARNING
INF_CLEANING_LAST_OUTWARRANTY	Regular cleaning mandatory Click on 'Cancel' and proceed with cleaning immediately. Would you continue, this will void the print head warranty. (this state is reported only when the printing starts)	WARNING
INF_CLEANING_REQUIRED	Regular cleaning mandatory - No card issuance allowed by your Administrator Click on 'Cancel' and proceed with cleaning immediately. (this state is reported only when the printing starts)	WARNING
INF_UNKNOWN_RIBBON (1)	Ribbon not identified Ribbon identification impossible. Please proceed with Manual settings.	WARNING
INF_UNKNOWN_RIBBON (2)	Ribbon not identified Ribbon identification impossible. Please proceed with Manual settings.	WARNING
INF_WRONG_ZONE_ALERT	Ribbon not valid Compatibility problem between ribbon and printer. Less than 50 printouts remaining. Please contact your reseller.	WARNING

	(this state is reported only when the printing starts, every 10 cards approximatively)	
INF_WRONG_ZONE_RIBBON	Ribbon not valid Compatibility problem between ribbon and printer. Please contact your reseller. (this state is reported only when the printing starts, every 25 cards approximatively)	WARNING
RIBBON_ENDED (DEF)	Ribbon end Ribbon end, please replace by a new one.	WARNING
DEF_NO_LAMINATE	No film No film in lamination module. Please replace the film.	WARNING
INF_LAMINATE_UNKNOWN	Film not identified Unknown film. Please contact your reseller.	WARNING
INF_LAMINATE_LOW	Film close to the end Film close to the end. Please arrange for replacement. (this state is reported only when the printing starts, every 10 cards approximatively, and changes automatically after 10s)	WARNING
LAMINATE_END (DEF)	Film end Film end. Please replace the film.	WARNING
DEF_LAMINATE_UNSUPPORTED	Incompatible film Film incompatible with lamination module. Please contact your reseller.	WARNING
DEF_LAMI_COVER_OPEN	Door open Lamination module door open. Close the lamination module door.	WARNING
INF_LAMI_TEMP_NOT_READY	Adjusting temperature Lamination module temperature adjustment in progress. Please wait...	WARNING
DEF_LAMI_HOPPER_FULL	Output jammed The lamination output is jammed. Please remove the card(s) and retry.	WARNING

Disclaimer

While Evolis makes every effort to deliver high quality products, we do not guarantee that our products are free from defects. Our SDK, samples and demo software, any content or documentation delivered in this package (Evolis SDK) is provided "as is". The use of it is at your own risk.

Evolis makes no warranties as to performance, merchantability, fitness for a particular purpose, or any other warranties whether expressed or implied.

No oral or written communication from or information provided by Evolis shall create a warranty.

Under no circumstances shall Evolis be liable for direct, indirect, special, incidental, or consequential damages resulting from the use, misuse, or inability to use this Software Development Kit (named Evolis SDK), even if Evolis has been advised of the possibility of such damages

