



### **Proprietary Notice**

This document is the property of MDS Aero Support Corporation, and is provided on condition that it be used exclusively for evaluation purposes. Any duplication or reproduction, in whole or in part, without prior written consent of an authorized MDS Aero Support Corporation representative is prohibited.

**TABLE OF CONTENTS**

<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1 Purpose .....	1
1.2 Scope .....	1
1.3 Applicable Documents .....	1
1.4 Codes and Standards.....	1
1.5 Abbreviations and Definitions.....	2
<b>2. DESIGN .....</b>	<b>3</b>
2.1 Introduction .....	3
2.2 Interface "Output".....	3
2.2.2 Design.....	3
2.2.3 Use Cases .....	3
2.2.4 Definition in IDL.....	4
2.2.5 Methods .....	5
2.2.6 Properties.....	6
2.2.7 Events Fired.....	8
2.2.8 Usage Conditions and Restrictions.....	8
2.2.9 Persistent Data.....	8
2.2.10 Example.....	8
2.3 Interface "Interactive".....	10
2.3.2 Design.....	10
2.3.3 Use Cases .....	10
2.3.4 Definition in IDL.....	10
2.3.5 Methods .....	10

2.3.6	Properties.....	10
2.3.7	Events Fired.....	10
2.3.8	Usage Conditions and Restrictions.....	11
2.3.9	Persistent Data.....	11
2.3.10	Example.....	11

## **1. INTRODUCTION**

### **1.1 Purpose**

1.1.1 NewDAS (New Data Acquisition System) is a data acquisition system for gas turbine test cells. This specification defines the technical requirements for the interface offered by the Print Server, which is used for continuous printing.

1.1.2 This document defines the COM interface to be used by the clients.

### **1.2 Scope**

1.2.1 This document is intended for programmers of the COM client components using the COM interface(s) specified herein as well as the programmers of the server program offering the COM interface(s).

### **1.3 Applicable Documents**

<b>Number</b>	<b>Title</b>
ES78001.2620	Functional Requirements Document for NewDAS
ES78018.2658	Engineering Specification for Continuous Printing

### **1.4 Codes and Standards**

<b>Number</b>	<b>Title</b>
	ICD Template - COM

## 1.5 Abbreviations and Definitions

Term	Definition
May	An option or permission
Shall	A mandatory requirement
Should	A recommendation
Will	A statement of intent
COM	Component Object Model
DCOM	Distributed Common Object Model
GUI	Graphical User Interface
ICD	Interface Control Document
IDL	Interface Definition Language
NewDAS	New Data Acquisition System

## **2. DESIGN**

### **2.1 Introduction**

- 2.1.1.1 The server shall be accessible over the network via DCOM.
- 2.1.1.2 The server shall be an out-of-process server.
- 2.1.1.3 The server shall be an executable, i.e. it shall have the file extension *.exe*.
- 2.1.1.4 There shall only be one instance of the server running simultaneously within NewDAS.
- 2.1.1.5 The server shall be thread-safe, i.e. several clients shall be able to access it simultaneously.
- 2.1.1.6 The server shall be self-registering by calling it with the argument *"/RegServer"*. Registration will take place when NewDAS is installed.
- 2.1.1.7 The server shall have a configurable time-out, i.e. it shall terminate the configured time after the last connection has been destroyed.
- 2.1.1.8 There will be no standard interfaces implemented other than *IUnknown*, which is implemented with every COM interface.

### **2.2 Interface "Output"**

- 2.2.1.1 The identification of the interface shall be *"PrintServer.Output"*.

#### **2.2.2 Design**

- 2.2.2.1 This interface shall be a dispatch interface.
- 2.2.2.2 This interface shall be an automation interface.

#### **2.2.3 Use Cases**

- 2.2.3.1 The use cases are defined in the Engineering Specification for the Continuous Printing.

## 2.2.4 Definition in IDL

### 2.2.4.1 The definition of the interface in IDL shall be as follows.

```
// Printing of text, graphics and formfeed
[id(1)] HRESULT Print([in] BSTR Text);
[id(2)] HRESULT Flush();
[id(4)] HRESULT PrintGraphicFile([in] BSTR FileName, [in] BOOL Remove);

// Reprinting a protocol page and protocol file
[id(6)] HRESULT PrintProtocolPage([in] BSTR FileName, [in] long PageNumber);
[propget, id(10)] HRESULT Protocol([out, retval] BSTR *pVal);
[propput, id(10)] HRESULT Protocol([in] BSTR newVal);

// Font definition
[id(3)] HRESULT ChangeFont([in] BSTR FaceName, [in] long Height, [in] long
Weight, [in] BOOL Italic);
[propget, id(21)] HRESULT FontFaceName([out, retval] BSTR *pVal);
[propget, id(22)] HRESULT FontHeight([out, retval] long *pVal);
[propget, id(23)] HRESULT FontWeight([out, retval] long *pVal);
[propget, id(24)] HRESULT FontItalic([out, retval] BOOL *pVal);

// Logo file and position
[id(5)] HRESULT DefineLogo([in] BSTR FileName, [in] short Position);
[propget, id(19)] HRESULT LogoFile([out, retval] BSTR *pVal);
[propget, id(20)] HRESULT LogoPosition([out, retval] short *pVal);

// Header and footer
[propget, id(7)] HRESULT Header([out, retval] BSTR *pVal);
[propput, id(7)] HRESULT Header([in] BSTR newVal);
[propget, id(8)] HRESULT Footer([out, retval] BSTR *pVal);
[propput, id(8)] HRESULT Footer([in] BSTR newVal);

// Printer
[propget, id(9)] HRESULT Printer([out, retval] BSTR *pVal);
[propput, id(9)] HRESULT Printer([in] BSTR newVal);

// Page orientation
[propget, id(11)] HRESULT Orientation([out, retval] short *pVal);
[propput, id(11)] HRESULT Orientation([in] short newVal);

// Page numbering
[propget, id(12)] HRESULT NextPageNumber([out, retval] long *pVal);
[propput, id(12)] HRESULT NextPageNumber([in] long newVal);

// Time-out
[propget, id(13)] HRESULT TimeOut([out, retval] long *pVal);
[propput, id(13)] HRESULT TimeOut([in] long newVal);

// Clipping behaviour
[propget, id(14)] HRESULT Clipping([out, retval] BOOL *pVal);
[propput, id(14)] HRESULT Clipping([in] BOOL newVal);
```

```
// Page sizes
[propget, id(15)] HRESULT HorzSize([out, retval] float *pVal);
[propget, id(16)] HRESULT VertSize([out, retval] float *pVal);
[propget, id(17)] HRESULT HorzRes([out, retval] long *pVal);
[propget, id(18)] HRESULT VertRes([out, retval] long *pVal);

// User defined escape characters
[id(19)] HRESULT EnableEscape([in] char Escape, [in] BSTR Description);
[id(20)] HRESULT DisableEscape([in] char Escape);
[propget, id(21)] HRESULT EscapeValue([in] char Escape, [out, retval] BSTR
*pVal);
[propget, id(21)] HRESULT EscapeValue([in] char Escape, [in] BSTR newVal);

// Interactive access via dialogues
[propget, id(22)] HRESULT Interactive([out, retval] LPDISPATCH *pVal);
```

## 2.2.5 Methods

2.2.5.1 The method ***Print*** shall print the text contained in *Text*. The string may consist of an arbitrary number of lines separated by control characters.

2.2.5.2 The method ***Flush*** shall print the content of the buffer, if it is not empty.

2.2.5.3 The method ***PrintGraphicFile*** shall print the graphic contained in the enhanced meta file named by *FileName*, placing it on a separate page. If *FileName* is not valid, an error status shall be returned.

2.2.5.4 The method ***ChangeFont*** shall change the font using the given *FaceName*, *Height* in points, *Weight* as usual in Windows (400 = normal, 700 = bold) and *Italic* indicating whether it shall be an italic font. This method will enable the client to set several properties simultaneously.

2.2.5.5 The method ***DefineLogo*** shall define the bitmap file named *FileName*, which contains the logo, and the logo position *Position* with the following valid values:

0 = top right

1 = top left

2 = bottom left

3 = bottom right

The logo shall be printed with a height of 1.5 cm on each page. This method will enable the client to set several properties simultaneously.



2.2.5.6 The method ***PrintProtocolPage*** shall print the page with number *PageNumber* contained in the protocol file *FileName*. The file will have to be generated by the print server. If *FileName* is not valid an error status shall be returned. If a page with the required number is not contained in the file nothing shall be printed. If several pages with the desired number are contained only the first one shall be printed.

2.2.5.7 The method ***EnableEscape*** shall define *Escape* as escape character, i.e. if *Escape* = 'c' then the sequence %c will be replaced in the header and footer by the property *EscapeValue* ('c') (cf. 2.2.6.18). *Description* will be used for display in the configuration dialogue (cf. 2.3.5.1).

2.2.5.8 The method ***DisableEscape*** shall disable the escape character *Escape*.

## 2.2.6 Properties

2.2.6.1 All properties shall be global, i.e. the properties apply to all clients.

2.2.6.2 The get/put property ***Header*** shall represent the header as string containing control characters and escape sequences for inserting additional information. Lines that are too long are clipped. The height of the header must not exceed the height of half a page minus 1 line; otherwise it is clipped. Arbitrary strings are possible with an empty string as default.

2.2.6.3 The get/put property ***Footer*** shall represent the footer as string containing control characters and escape sequences for inserting additional information. Lines that are too long are clipped. The height of the footer must not exceed the height of half a page minus 1 line; otherwise it is clipped. Arbitrary strings are possible with an empty string as default.

2.2.6.4 The get/put property ***Printer*** shall denote the name of the printer to be used. For a network printer abc on server xyz this would be \\xyz\abc. For a local printer it is the name as given in the installation. Possible values are the names of the installed printers and the empty string representing the default printer. The default shall be the empty string.

2.2.6.5 The get/put property ***Protocol*** shall denote the name of the protocol file. An empty string shall mean that no protocol is written. Possible values are valid file names and the empty with the latter as default.

2.2.6.6 The get/put property ***Orientation*** shall denote the page orientation with the following valid values:

1 = Portrait

2 = Landscape (default)

- 2.2.6.7 The get/put property ***NextPageNumber*** shall denote the number of the next page printed. Initially, it shall have the value 1.
- 2.2.6.8 The get/put property ***TimeOut*** shall be the time in milliseconds after which the Print Server terminates by flushing its buffer. Possible values shall be non-negative integers with 5000 as default.
- 2.2.6.9 The get/put property ***Clipping*** shall control how lines which are too long are treated. If clipping is set these lines are clipped such that the end of the line is lost. If it is not set, which is the default, the line is wrapped always writing as many letters in a line as possible. Possible values shall be TRUE and FALSE with TRUE as default.
- 2.2.6.10 The get property ***HorzSize*** shall return the width of the rectangle where the text, excluding header and footer is printed on the page, in mm.
- 2.2.6.11 The get property ***VertSize*** shall return the height of the rectangle where the text, excluding header and footer is printed on the page, in mm.
- 2.2.6.12 The get property ***HorzRes*** shall return the width of the rectangle where the text, excluding header and footer is printed on the page, in printer pixels.
- 2.2.6.13 The get property ***VertRes*** shall return the height of the rectangle where the text, excluding header and footer is printed on the page, in printer pixels.
- 2.2.6.14 The get property ***FontFaceName*** shall return the face name of the font set in *ChangeFont* (cf. 2.2.5.4). Possible values shall be the face names of the installed fonts with “Courier New” as default.
- 2.2.6.15 The get property ***FontHeight*** shall return the height of the font set in *ChangeFont* (cf. 2.2.5.4). Possible values shall be positive integers with 12 as default.
- 2.2.6.16 The get property ***FontWeight*** shall return the weight of the font set in *ChangeFont* (cf. 2.2.5.4). Possible values shall be positive integers with 400 as default.
- 2.2.6.17 The get property ***FontItalic*** shall return the italic flag of the font set in *ChangeFont* (cf. 2.2.5.4). Possible values shall be TRUE and FALSE where FALSE shall be the default.
- 2.2.6.18 The get/put property ***EscapeValue*** shall return the string which replaces the escape sequence %c in the header and footer, where *Escape* = ‘c’. The default shall be an empty string.
- 2.2.6.19 The get property ***Interactive*** shall return an *Interactive* interface (cf. 2.3).

## 2.2.7 Events Fired

2.2.7.1 There will be no events.

## 2.2.8 Usage Conditions and Restrictions

2.2.8.1 All properties will have to be set before they apply to the calls of *Print*, *Flush*, *PrintGraphicFile* or *PrintProtocolPage*. This will also apply to the properties set by the methods *DefineLogo* and *ChangeFont*. The properties will also apply to the buffer content, i.e. the properties will always apply to the whole content of a page.

2.2.8.2 File names in the *PrintGraphicFile*, *PrintProtocolPage* and *DefineLogo* method and in the *Protocol* and *LogoFile* properties will be interpreted as file names on the computer where the Print Server is running, i.e. drive names will have to be defined on this computer.

2.2.8.3 The properties *HorzRes*, *VertRes*, *HorzSize* and *VertSize* are only defined if the *Printer* property is set to a valid printer, including an empty string for the default printer.

## 2.2.9 Persistent Data

2.2.9.1 All properties shall be persistently stored in an initialisation file.

## 2.2.10 Example

2.2.10.1 The following is an example of a VBS client.

```
' Directory with Print Server files.
Dir = "D:\PrintServer\"

' Establish the connection to the server.
Set app = CreateObject ("PrintServer.Output")

' Set the time-out of 1 second.
app.TimeOut = 1000

' Set the printer.
app.Printer = "\\MPRT01\DD1654"

' Define the layout of the page.
DefineLayout

' Print some text.
PrintText

' Print an enhanced meta file.
app.PrintGraphicFile Dir & "MyDiagram.emf"
```

```
' Notify the user.
MsgBox "Done"
```

```
' -----

sub DefineLayout
    ' Set an additional escape character.
    app.EnableEscape ('q', "My escape")

    ' Set header and footer.
    app.Header = "Header of page %P, my escape %q" & vbCrLf & _
        "second header line:" & String (100, "W") & String (100, "X") & _
        String (100, "A")
    app.Footer = "Footer of page %P date %d.%m.%Y time %H:%M:%S"

    ' Set the protocol file.
    app.Protocol = Dir & "Protocol.txt"

    ' Set the page orientation.
    const oPortrait = 1
    const oLandscape = 2
    app.Orientation = oPortrait

    ' Set the font.
    app.ChangeFont "Arial", 12, 300, true

    ' Set the logo.
    app.DefineLogo Dir & "Logo.bmp", 0

    ' Set the clipping behaviour.
    app.Clipping = false
end sub
```

```
' -----

sub PrintText
    ' Define the user defined escape value.
    app.EscapeValue ('q') = "MyEscapeValue"

    ' Print 10 lines of text.
    fill = String (120, "*")
    dim text
    for i = 1 to 10
        text = text & fill & " Line " & i & vbCrLf
    next
    app.Print text
end sub
```

## 2.3 Interface "Interactive"

2.3.1.1 This interface shall not be directly created. It shall be created by the *Interactive* property of the interface *Output*.

2.3.1.2 Screenshots of the generated dialogues can be found in the Engineering Specification.

## 2.3.2 Design

2.3.2.1 This interface shall be a dispatch interface.

2.3.2.2 This interface shall be an automation interface.

## 2.3.3 Use Cases

2.3.3.1 The use cases are defined in the Engineering Specification for the Continuous Printing.

## 2.3.4 Definition in IDL

2.3.4.1 The definition of the interface in IDL shall be as follows.

```
// Edit configuration data
[id(1)] HRESULT EditConfig ();

// Print pages from a protocol file
[id(2)] HRESULT PrintProtocol([in] BSTR FileName);
```

## 2.3.5 Methods

2.3.5.1 The method *EditConfig* shall display a dialogue which enables the user to change all properties of the interface "Output".

2.3.5.2 The method *PrintProtocol* shall display a dialogue which enables the user to reprint pages from the protocol file *FileName*. If *FileName* is not valid an error status shall be returned.

## 2.3.6 Properties

2.3.6.1 There will be no properties.

## 2.3.7 Events Fired

2.3.7.1 There will be no events.

**2.3.8 Usage Conditions and Restrictions**

2.3.8.1 The interface can only be used on the computer where the Print Server is running.

**2.3.9 Persistent Data**

2.3.9.1 There will be no persistent data specific for this interface.

**2.3.10 Example**

2.3.10.1 The following is an example of a VBS client.

```
' Establish the connection to the server.
Set app = CreateObject ("PrintServer.Output")

' Create an Interactive interface
Set interact = app.Interactive

'Edit the properties of interface output.
interact.EditConfig

' Let the user reprint pages.
interact.PrintProtocol "D:\PrintServer\MyProtocol.txt"
```