
**BASE64: Thistle Type A Interface to the Code
Magus Base64 Codec Functions**

CML00004-01

Code Magus Limited (England reg. no. 4024745)
Number 6, 69 Woodstock Road
Oxford, OX2 6EY, United Kingdom
www.codemagus.com
Copyright © 2014 by Code Magus Limited
All rights reserved

Contents

1	Thistle Interface	2
1.1	Introduction	2
1.2	<i>Thistle</i> Interface Definition for BASE64	2
2	BASE64 Type A Interface reference	3
2.1	BASE64 Type A Interface Methods	3
3	Example <i>Thistle</i> Script Using BASE64	4

1 Thistle Interface

1.1 Introduction

The Code Magus Limited Base64 Codec [1] provides functions to encode a buffer as base64 encoded text. This is an encoding scheme that encodes binary data by treating it numerically and translating it into a base 64 representation. The particular choice of base is due to the history of character set encoding: one can choose 64 characters that are both part of the subset common to most encodings, and also printable. The *Thistle* Type A Interface BASE64 is the interface to the Code Magus Base64 Codec Functions and provides the means by which *Thistle* scripts (packages and use-cases) can encode and decode data from BASE64 encoding.

This document shows how the BASE64 functions can be used by the *Thistle* scripts.

1.2 *Thistle* Interface Definition for BASE64

The *Thistle* Interface Definition for BASE64 is a normal Type A Interface to *Thistle*.

```
interface BASE64;
{
  --
  -- $Author: patrick $
  -- $Date: 2008/05/20 07:28:09 $
  -- $Id: BASE64.tid,v 1.1 2008/05/20 07:28:09 patrick Exp $
  -- $Name: $
  -- $Revision: 1.1 $
  -- $State: Exp $
  --
  -- $Log: BASE64.tid,v $
  -- Revision 1.1 2008/05/20 07:28:09 patrick
  -- Initial add
  --
  --
}
type : typea;
module : "c:\CodeMagus\CodeMagus\bin\base64tai.dll";
init : base64tai_init;
end.
```

2 BASE64 Type A Interface reference

2.1 BASE64 Type A Interface Methods

The following methods are exposed by the BASE64 Type A Interface. These are shown here together with the names of their parameters.

`Encode ()` Function `Encode ()` is the Type A Interface wrapper function for the Base64 function `SRDB64EC ()`. This function encodes a buffer which is passed to the function.

Function `Encode ()` returns a base64 encoded buffer which is an ASCII encoded character array but will be returned as a BLOB.

Parameters for Function `Encode ()`

`buffer` The buffer parameter contains the data buffer to be encoded which must be a BLOB.

`Decode ()` Function `Decode ()` is Type A Interface wrapper function for the Base64 codec function `SRDB64DC ()`.

This function maps between the *Thistle* called function and the Base64 codec function. If the Base64 codec operation fails this reports the failure to the executing script.

If function `Decode ()` successfully Decodes a buffer, then the function returns blob . **Parameters for Function `Decode ()`**

`buffer` The buffer supplied to the `Decode ()` function is an ASCII byte array which must be a BLOB.

3 Example *Thistle* Script Using **BASE64**

The module `base64tai` wraps the `Base64` library interface and exposes it to the *Thistle* environment as a Type A Interface. The following *Thistle* Interface Definition defines this interface to the *Thistle* environment:

```
interface BASE64;
{
  --
  -- $Author: patrick $
  -- $Date: 2008/05/20 07:28:09 $
  -- $Id: BASE64.tid,v 1.1 2008/05/20 07:28:09 patrick Exp $
  -- $Name: $
  -- $Revision: 1.1 $
  -- $State: Exp $
  --
  -- $Log: BASE64.tid,v $
  -- Revision 1.1 2008/05/20 07:28:09 patrick
  -- Initial add
  --
  --
}
  type : typea;
  module : "c:\CodeMagus\CodeMagus\bin\base64tai.dll";
  init : base64tai_init;
end.
```

The *Thistle* run-time locates the named interface definition using the standard *Thistle* external pathing convention. For example, the following in the preamble of a script

```
interface base64 : CodeMagusExtras.BASE64
```

introduces `base64` as an internal local name of the interface. This expects that the variable `CodeMagusExtras` is defined suitably in a *Thistle* configuration file. For example in the directory containing a script with the above `interface` definition in its preamble, a configuration file might include:

```
DefaultRootDirectory=C:\
CodeMagusExtras=Codemagus\CodeMagus\bin\
```

and in which case, the *Thistle* interface definition file shown above would be expected to have the fully qualified name of

```
C:\CodeMagus\CodeMagus\bin\BASE64.tid
```

The following scripts shows how the `BASE64` interface could be used for processing record streams:

```
package BASE64SamplePackage;

  created by 'Patrick Power';
  description 'Test BASE64 TypeA Interface';
  date      2008-05-20T13:13:13;
```

3 EXAMPLE THISTLE SCRIPT USING BASE64

```
target          'BASE 64';

interface Base64          : CodeMagusExtras.BASE64;
begin
  codec := Base64.Connect();
  msg.text := "introduces \\texttt{base64} as an internal local name of the interface. T'
              \" expects that the variable \\texttt{CodeMagusExtras} is defined suitable
              \" \\emph{Thistle} configuration file. For example in the directory containing
              \" script with the above \\texttt{interface} definition in its preamble.
              \" configuration file might include:";

  msg.allowchars := <<0d0a>>;
  msg.encoded := codec.Encode(System.TranslateFromStringToASCII(msg.text));
  msg.encodedString := System.TranslateFromASCIIToString(msg.encoded,
                                                         ALLOWCHARS := msg.allowchars);
  msg.decoded := System.TranslateFromASCIIToString(codec.Decode(msg.encoded),
                                                    ALLOWCHARS := msg.allowchars);
  System.DumpScope(msg);
end.
```

Running this script produces the expected (but truncated here) output:

```
Initializing run...
CodeMagusExtras.BASE64
Interface BASE64 Codec initialised.
$Id: output.txt,v 1.2 2008/05/26 15:21:28 patrick Exp $
msg
  allowchars <property:blob len 2> =
    00..___05..___10..___15..___20..___25..___30..
0000: 0D0A
0000: ....

  decoded <property:string len 336> =
    'introduces \texttt{base64} as an internal local name of the interface. T'
    'his expects that the variable \texttt{CodeMagusExtras} is defined suitable'
    'ly in a \emph{Thistle} configuration file. For example in the directory '
    'containing a script with the above \texttt{interface} definition in its p'
    'reamble, a configuration file might include:'
  encoded <property:blob len 462> =
    00..___05..___10..___15..___20..___25..___30..
0000: 61573530636D396B64574E6C63794263644756346448523065324A6863325532
0000: a/W.5.0.c.m_9.k,d.W.N+l%c.y`B.c.d.G.V.4.d.H.R.0.e.2.J.h.c.2.U.2.
0032: 4E48306759584D6759573467615735305A584A75595777676247396A59577767
0032: N+H.0.g.Y.X.M(g.Y.W.4.g.a/W.5.0.Z!X.J.u.Y.W.w.g.b.G.9.j.Y.W.w.g.
0064: 626D46745A5342760D0A5A69423061475567615735305A584A6D59574E6C4C69
0064: b.m_F.t.Z!S.B.v.....Z!i.B.0.a/G.U.g.a/W.5.0.Z!X.J.m_Y.W.N+l%L<i.
0096: 425561476C7A494756346347566A64484D67644768686443423061475567646D
0096: B.U.a/G.l%z:I.G.V.4.c.G.V.j.d.H.M(g.d.G.h.h.d.C.B.0.a/G.U.g.d.m_
0128: 467961574669624755675848526C654852300D0A644874446232526C5457466E
0128: F.y`a/W.F.i.b.G.U.g.X.H.R.l%e.H.R.0.....d.H.t.D.b.2.R.l%T.W.F.n>
0160: 64584E466548527959584E3949476C7A4947526C5A6D6C755A57516763335670
0160: d.X.N+F.e.H.R.y`Y.X.N+9.I.G.l%z:I.G.R.l%Z!m_l%u.Z!W.Q.g.c.3.V.p.
```

3 EXAMPLE THISTLE SCRIPT USING BASE64

```
0192: 6447466962486B6761573467595342635A5731776148745561476C7A0D0A6447
0192: d.G.F.i.b.H.k,g.a/W.4.g.Y.S.B.c.Z!W.1.w.a/H.t.U.a/G.l%z:....d.G.
0224: 786C6653426A6232356D61576431636D46306157397549475A70624755754945
0224: x.l%f.S.B.j.b.2.5.m_a/W.d.l.c.m_F.0.a/W.9.u.I.G.Z!p.b.G.U.u.I.E.
0256: 5A766369426C654746746347786C49476C754948526F5A53426B61584A6C5933
0256: Z!v.c.i.B.l%e.G.F.t.c.G.x.l%I.G.l%u.I.H.R.o?Z!S.B.k,a/X.J.l%Y.3.
0288: 5276636E6B670D0A5932397564474670626D6C755A79426849484E6A636D6C77
0288: R.v.c.n>k,g.....Y.2.9.u.d.G.F.p.b.m_l%u.Z!y`B.h.I.H.N+j.c.m_l%w.
0320: 644342336158526F4948526F5A534268596D39325A5342636447563464485230
0320: d.C.B.3.a/X.R.o?I.H.R.o?Z!S.B.h.Y.m_9.2.Z!S.B.c.d.G.V.4.d.H.R.0.
0352: 65326C75644756795A6D466A5A5830670D0A5A47566D6157357064476C766269
0352: e.2.l%u.d.G.V.y`Z!m_F.j.Z!X.0.g.....Z!G.V.m_a/W.5.p.d.G.l%v.b.i.
0384: 4270626942706269427064484D6763484A6C5957316962475573494745675932
0384: B.p.b.i.B.p.b.i.B.p.d.H.M(g.c.H.J.l%Y.W.l.i.b.G.U.s.I.G.E.g.Y.2.
0416: 39755A6D6C6E64584A6864476C766269426D6157786C494731700D0A5A326830
0416: 9.u.Z!m_l%n>d.X.J.h.d.G.l%v.b.i.B.m_a/W.x.l%I.G.l.p.....Z!2.h.0.
0448: 49476C75593278315A4755360D0A
0448: I.G.l%u.Y.2.x.l.Z!G.U.6.....
```

```
endcodedString <property:string len 462> =
```

```
'aW50cm9kdWNlcyBcdGV4dHR0e2Jhc2U2NH0gYXMGYmY4gaW50ZXJuYWwgG9jYWwgYmFtZSBv'
',
ZiB0aGUgaW50ZXJmYWwNlLiBUaGlzIGV4cGVjdHMgdGhhdCB0aGUgdmFyaWFibGUgXHRleH'
'R0
dHtDb2RlTWFndXNFeHRyYXN9IGlzIGRlZmluZWQgc3VpdGFibHkgaW4gYSBcZWlwaHtU'
'aGlz
dGxlfSBjb25maWdlcmF0aW9uIGZpbGUuIEZvcjBleGFtcGxlIGluIHRoZSBkaXJlY3'
'Rvcnkq
Y29udGFpbmluZyBhIHNjcmlwdCB3aXRoIHRoZSBhYm92ZSBcdGV4dHR0e2ludGVy'
'ZmFjZX0g
ZGVmaW5pdGlvb25maWdlcmF0aW9uIGZpbGUuIEZvcjBleGFtcGxlIGluIHRoZSBkaXJlY3'
'BmaWxlIG1p
Z2h0IGluY2x1ZGU6
'
```

```
text <property:string len 336> =
```

```
'introduces \texttt{base64} as an internal local name of the interface. T'
'his expects that the variable \texttt{CodeMagusExtras} is defined suitab'
'ly in a \emph{Thistle} configuration file. For example in the directory '
'containing a script with the above \texttt{interface} definition in in i'
'ts preamble, a configuration file might include:'
Interface BASE64 Codec cleanup complete.
$Id: output.txt,v 1.2 2008/05/26 15:21:28 patrick Exp $
Run completed...
```

References

- [1] BASE64: Thistle Type A Interface to the Code Magus Base64 Codec Functions. CML Document CML00004-01, Code Magus Limited, July 2008. [PDF](#).