
**applparms: Application Parameters Type A
Interface Guide and Reference Version 1**

CML00057-01

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	Interface Definition	2
2	Interface reference	3
2.1	Methods	3
2.1.1	open ()	3
2.2	Example <i>Thistle</i> Script	4

1 Thistle Interface

1.1 Introduction

The *Thistle* Type A Interface `applparmtai` is the interface to the Code Magus `applparms` Library and provides the means by which *Thistle* scripts (packages and usecases) can open and make use of the parameters defined in a given application parameter file.

It is not the responsibility of the *Thistle* Type A Interface `applparmtai` to define how a particular user interface should look or how it should interact with the current application parameter file but rather to provide a means of accessing the application parameter file and making the parameters defined in that file available to the *Thistle* runtime environment.

This document shows how the parameters defined in an application parameter file can be used by the *Thistle* scripts. For instruction on how a particular application is used to modify application parameter files and for more information on the `applparms` library, the appropriate documentation should be consulted.

This document uses the `aptabgui` user interface to illustrate how to open a given application parameter file as well as accessing it's parameters in a *Thistle* script.

1.2 Interface Definition

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

```
interface ApplParms;
{
  -- $Author: francois $
  -- $Date: 2018/06/20 07:49:37 $
  -- $Id: ApplParms.tid,v 1.4 2018/06/20 07:49:37 francois Exp $
  -- $Name: $
  -- $Revision: 1.4 $
  -- $State: Exp $
  --
  -- $Log: ApplParms.tid,v $
  -- Revision 1.4 2018/06/20 07:49:37 francois
  -- Initial commit or hard path removed.
  --
  -- Revision 1.2 2018/06/10 10:23:32 Francois
  -- *** empty log message ***
  --
  -- Revision 1.1 2018/05/01 19:50:49 Francois
  -- *** empty log message ***
  --
  -- Revision 1.1 2018/05/01 08:41:23 Francois
```

```
-- *** empty log message ***
--
-- Revision 1.1  2018/04/06 14:33:34  Francois
-- *** empty log message ***
--
-- Revision 1.1  2018/03/18 07:10:16  Francois
-- *** empty log message ***
--
-- Revision 1.3  2010/01/08 08:59:47  justin
-- updated path to appropriate installation directory.
--
-- Revision 1.2  2010/01/05 09:58:53  justin
-- fixed syntax error.
--
-- Revision 1.1  2010/01/05 09:56:34  justin
-- take on.
--
--
--
--
}

Type : typea;
Module : "${THISTLEPORTALBINPATH}applparmstai${THISTLEPORTALSUFDL}";
Init : applparmstai_init;
end.
```

2 Interface reference

2.1 Methods

The `open()` method is the only method exposed by the `applparmtai` Type A Interface. This is shown here together with the name of the parameter and with the associate `applparms` Library function name. For a description of the method, refer to the corresponding `applparms` Library documentation.

2.1.1 `open()`

`open()` Function `open()` is the Type A Interface wrapper function for the `applparms` library function `applparms_open()`.

This function maps between the *Thistle* called function and the `applparms` library function. If the `applparms` library operation fails then this writes the error message to the log and reports the failure to the executing script.

If function `open()` successfully opens an application parameter file, then the function returns all the parameters contained in the application parameter file to *Thistle* so that the parameter values may be used as required.

Parameters for Function `open()`

`filename` The file name string supplied to the `open()` function is a text string in the format as required by the first parameter of the `applparms_open()` function. This string names the path and file name of the application parameter file to be opened.

`flags` The flags parameter is required and supplies, in operator-string form (see `flagopts.pdf`), the values of the desired flags. The following flags are supported:

Flag Value	Flag String	Option
APPLPARMS_OPT_OFFLINE	OFFLINE	process in non-interactive mode, no UI
APPLPARMS_OPT_VERBOSE	VERBOSE	produce verbose output when processing

Examples

```
flags:="/VERBOSE"
flags:="/OFFLINE"
```

2.2 Example *Thistle* Script

The module `ApplParmtai` wraps the `applparms` 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 ApplParms;
{
  -- $Author: francois $
  -- $Date: 2018/06/20 07:49:37 $
  -- $Id: ApplParms.tid,v 1.4 2018/06/20 07:49:37 francois Exp $
  -- $Name: $
  -- $Revision: 1.4 $
  -- $State: Exp $
  --
  -- $Log: ApplParms.tid,v $
  -- Revision 1.4 2018/06/20 07:49:37 francois
  -- Initial commit or hard path removed.
  --
  -- Revision 1.2 2018/06/10 10:23:32 Francois
  -- *** empty log message ***
  --
  -- Revision 1.1 2018/05/01 19:50:49 Francois
```

```

-- *** empty log message ***
--
-- Revision 1.1  2018/05/01 08:41:23  Francois
-- *** empty log message ***
--
-- Revision 1.1  2018/04/06 14:33:34  Francois
-- *** empty log message ***
--
-- Revision 1.1  2018/03/18 07:10:16  Francois
-- *** empty log message ***
--
-- Revision 1.3  2010/01/08 08:59:47  justin
-- updated path to appropriate installation directory.
--
-- Revision 1.2  2010/01/05 09:58:53  justin
-- fixed syntax error.
--
-- Revision 1.1  2010/01/05 09:56:34  justin
--take on.
--
--
--
--
}

Type : typea;
Module : "${THISTLEPORTALBINPATH}applparmtai${THISTLEPORTALSUFDL}";
Init : applparmtai_init;
end.

```

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

```
interface ApplParms : CodeMagus.ApplParms;
```

introduces `applparmtai` as an internal local name of the interface.

The *Thistle* interface definition file shown above is installed with the Eresia Visual Test Environment to the directory listed below.

```
C:\Program Files\CodeMagusLimited\eresiavte\components\ApplParms.tid
```

The following script shows how the `applparmtai` interface could be used for opening an application parameter file and using the parameters contained within the file.

```

package ApplParmsExample;
{ preamble }

    created by 'Justin Albertyn';
    description 'Access parameters defined in an application parameter file';
    date 2005-08-02T09:31:28;
    target 'Eresia File Portal';
    interface ApplParms : CodeMagus.ApplParms;

```

```
begin
  APInstance := ApplParms.Connect();

  APInstance.open("C:/CodeMagus/Stl/GenerateDailySettlementFile.apd", "/VERBOSE");

  OutputFile := APInstance.SettlementFileName;
  InputFile   := APInstance.INFILE;

  System.WriteLine("Output file name: " # OutputFile);
  System.WriteLine("Input file name: "  # InputFile);

  return 0;
end.
```

When the running *Thistle* script reaches the call to open the application parameter file i.e.

```
APInstance.open("C:/CodeMagus/Settlement/GenerateDailySettlementFile.apd");
```

the interface described in the application parameter definition will be made available to the user. The `aptabgui.dll` is a GUI version used with *Thistle* scripts, this is shown in [Figure 1](#).

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

```

ApplParms File: C:/CodeMagus/Examples/MyConfig.apd Opened Successfully.
Output file name: SampeOutputFile.otp
Input file name: SampleInputFile.inp
20100111-103726 codemagus@SETEBOS Package ApplParmsExample has completed with
highest return code 0.
Interface applparms cleanup complete.
$Id: output.txt,v 1.1 2010/01/11 12:25:31 justin Exp $
20100111-103726 codemagus@SETEBOS Job J0002802_D20100111_T103724_ApplParmsExample
completed.

```


The application parameter file used in the script above is shown below.

```

application MyConfig;
-- A sample APD file used by the testappl program for running unit tests. If
-- changes are made to this file make sure that the test program is checked to
-- still work correctly as it assumes various attributes set in this APD in
-- order to test all the functionality of the AP library.
--
-- $Author: justin $
-- $Date: 2010/02/05 10:40:45 $
-- $Id: MyConfig.apd,v 1.4 2010/02/05 10:40:45 justin Exp $
-- $Name: $
-- $Revision: 1.4 $
-- $State: Exp $
--
-- $Log: MyConfig.apd,v $
-- Revision 1.4 2010/02/05 10:40:45 justin
-- aligned text.
--
-- Revision 1.3 2010/01/15 11:34:10 justin
-- updated for documentation purposes.
--
-- Revision 1.2 2010/01/11 12:39:53 justin
-- added apd config.
--
-- Revision 1.1 2010/01/11 12:31:55 justin
-- take on.
--
-- Revision 1.1 2010/01/06 09:54:37 darren
-- Take on
--
-- Revision 1.2 2009/12/31 09:37:04 hayward
-- Split program name into path,name and suffix
--
-- Revision 1.1.1.1 2009/12/29 09:20:52 hayward
-- Import appcmdui into CVS.
--
title "Example Usage of the Thistle Type A Interface to the applparms Library";
description "The Thistle Type A Interface applparmtai is the interface to the "
            "Code Magus applparms Library and provides the means by which "

```


Daily Test Settlement File Generation

 **CODE MAGUS**

www.codemagus.com

Generates the daily test settlement file in the format required by the target system and is ready for uploading to MVS for the nightly batch. Done
Abort

Run Date | System Name | Input File | Settlement File to be Written out | Password < >

Set the run date to produce a report with the chosen effective date. The format of the supplied date must be CCYYMMDD. If it is not supplied by the user then today's date is used.

Name: RUNDATE
Constraint: ^\ (19\ |20\) [0-9] [0-9] [0-1] [0-9] [0-3] [0-9] \$
Value: 3/16/2010 [Set] History: [dropdown]

Set Values Back to Mark Point [dropdown] [Apply]

Set Values Back to Date and Time: 3/16/2010 [dropdown] 7:58:29 PM [dropdown] [Apply]

New Mark Point: Name [input] [Set] Description [input]

```

Name:      GenerateDailySettlementFile
Title:     Daily Test Settlement File Generation
Description: Generates the daily test settlement file in the format
Store:     .
Interface: C:\Program Files\CodeMagusLimited\eresiavte\lib\aptabgui
Entry point: aptabgui_init
Log file:  .\codemagus_GenerateDailySettlementFile.apl
Log Start: 2010-03-16T19:58:29
Log End:   2010-03-16T19:58:29

```

Figure 1: Opening aptabgui from a *Thistle* script.

```

        "Thistle scripts can open and access a given application "
        "parameter file."
        ;
store "c:/logs";
set TODAY = ${DATE_YYYYMMDD};

-- The interface defines the shared object or DLL program that will interact
-- with the user to ensure that all parameters have a value. The value
-- for interface is string in quotes naming the DLL program or the word default
-- in which case the UI is the command line UI.

-- set CMDSUFF = ".so";
-- set CMDPATH = "./";
set CMDSUFF = ".dll";
set CMDPATH = "c:/Program Files/CodeMagusLimited/eresiavte/lib/";
interface "c:/Program Files/CodeMagusLimited/eresiavte/lib/aptabgui.dll";
entry aptabgui_init;

parameter RUNDATE
    title "Run Date";
    default ${TODAY};
    options date;    -- indicates system can use calendar input
    description "Set the run date to produce a report with the chosen "
        "effective date. The format of the supplied date must be CCYYMMDD";
    constraint "^\(19\|20\) [0-9] [0-9] [0-1] [0-9] [0-3] [0-9] $";
end

parameter SYSID
    title "System Name";
    default "DARRENDEVXP";
    options alphanumeric;    -- indicates a alphanumeric value
    description "The current name of the system this process is running "
        "on.";
    constraint "^.*$";
end

parameter INFILE
    title "Input File";
    default NULL; -- forces the user to insert a value if one is not found
    options filename;
        -- allows system to file system navigate to choose name
    description "Name of the raw input file to process in order to produce "
        "the report.";
    constraint "^[^ ]\+$"; --- at least one character, no spaces.
end

parameter SettlementFileName
    title "Settlment File to to be Written out";
    default NULL; -- forces the user to insert a value if one is not found
    options filename;
        -- allows system to file system navigate to choose name
    description

```

```
    "This is the name of the settlement test file that will be "  
    "created as the result of a success execution of this script. "  
    "The file will be in a format suitable for directly copying "  
    "into the concatenation of your settlement process. ";  
    constraint "^[^ ]\+$"; --- at least one character, no spaces.  
end  
end.
```