Configuring switchbox\_v2

Contents

[Purpose 3](#_Toc448391576)

[Locking Mechanism 4](#_Toc448391577)

[Process of Execution 4](#_Toc448391578)

[Interface Activity Logging 5](#_Toc448391579)

[Initial Settings 6](#_Toc448391580)

[Purpose: You will need to setup the Interface to use Switchbox v2 and run interfaces using the TestSwitchBox v2 6](#_Toc448391581)

[Functionality: Test Path 7](#_Toc448391582)

[Purpose: Allows File Path assignment for Test Export Sessions containing "TEST" or "TST" in the ExportCode (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path. 7](#_Toc448391583)

[Functionality: On Demand Path 8](#_Toc448391584)

[Purpose: Allows File Path assignment for “ONDEMAND”, "ON\_DEMAND" Export Session (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path. 8](#_Toc448391585)

[Functionality: OE Path (Open Enrollment) 9](#_Toc448391586)

[Purpose: Allows File Path assignment for Open Enrollment Export Sessions starting with "OE" in the ExportCode OR Export Code is 'OPENENROLL','PASSIVEOE','ACTIVEOE' (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path. 9](#_Toc448391587)

[Functionality: No BCP 10](#_Toc448391588)

[Purpose: Allows switchbox BCP to be disabled to allow for Multiple Files (AfterCollect) 10](#_Toc448391589)

[Functionality: No Empty 11](#_Toc448391590)

[Purpose: Do not create a File if it’s an Empty File 11](#_Toc448391591)

[Functionality: Upper 12](#_Toc448391592)

[Purpose: Convert all characters to Upper Case 12](#_Toc448391593)

[Functionality: Count 13](#_Toc448391594)

[Purpose: Allows the Control of the Record Count 13](#_Toc448391595)

[Functionality: UTF-16 Files 14](#_Toc448391596)

[Purpose: Allows the BCP Statement to generate UTF-16 Files (Use –w for bcp instead of –c) 14](#_Toc448391597)

[Functionality: Sort Fields 15](#_Toc448391598)

[Purpose: Allows the ability to Sort by additional fields (100 characters) 15](#_Toc448391599)

[Functionality: Other Control Characters (Override BCP Parameters) 16](#_Toc448391600)

[Purpose: Allows the ability to Override the BCP Parameters that follow Server Name 16](#_Toc448391601)

[Functionality: CountFilter 17](#_Toc448391602)

[Purpose: Allows the Control of the Record Count 17](#_Toc448391603)

[Functionality: UTF-8 Files 18](#_Toc448391604)

[Purpose: Allows the conversion of files in the UTF8Path Folder to UTF-8 file format 18](#_Toc448391605)

[Frequently Asked Questions (FAQ) 19](#_Toc448391606)

['ZAP' Is Gone! 19](#_Toc448391607)

[“My Interface is stuck in ‘Locking’ Mode…” 19](#_Toc448391608)

[“My Interface Always Fails because it takes over 60 minutes to run on average …” 19](#_Toc448391609)

[Custom Interface run on Web, but getting “ERROR: AHEmpExport.CreateFile ran successfully but no files were created for PayGroup: , PerControl: 201503311” 19](#_Toc448391610)

# Purpose

The switchbox\_v2 has enhanced features and functionality to allow more flexibility and control to the developer by leveraging the Configuration Table (dbo.u\_dsi\_configuration).

Below is a quick summary of the enhancements:

* **Locking Mechanism** – Allows multiple instances of a single custom interface to run in parallel.
* **Interface Activity Logging** – Allows custom interface activity to be logged in dbo.u\_dsi\_InterfaceActivityLog table
* **Test Path** – Ability to generate files to a separate Export Path for the ‘TEST’ Export Session.
* **OnDemand Path** – Ability to generate files to a separate Export Path for the ‘ONDEMAND’ Export Session.
* **No BCP** – Disables the BCP statement in switchbox\_v2, used in conjunction with Multiple File AfterCollects.
* **No Empty** – Do Not Create Export File if File is Empty.
* **Upper** – Convert all alpha characters to Upper Case.
* **Count** – Ability to override of the Record Count outputted.
* **UTF Files** – Ability to generate UTF-16 Files
* **Sort Fields** – Ability to sort up to 100 characters in each SubSort field.
* **Override BCP Parameters** – Ability to override the BCP parameters in the BCP statement of switchbox\_v2

# Locking Mechanism

The Locking Mechanism is a process designed to allow multiple instances of the same interface by FormatCode to run in parallel. The main purpose of this functionality is to allow new UltiPro processes to function, such as Payroll Automation – which runs multiple instances in batches at the same time.

## Process of Execution

* Get SPID# for Current Process.
* Reset Lock If SPID# for Current Process is Re-Ran.
* Load Instance into Interface Activity Log Table (dbo.u\_dsi\_InterfaceActivityLog) if SPID doesn't exist where Locked = 0, Completed = 0, Failed = 0.
* Pause 1 Second before Assigning Locked Session.
* If no processes of same FormatCode are currently Locked, then set the Locked = 1 for the Minimum RunID where Locked = 0.
* If any process of same FormatCode is currently locked, then add to a Wait Queue until SPID/RunID for Current Process is Locked.
  + Set @OverrideWaitMinutes (Defaulted to 3 Minutes).
    - This can be overridden by FormatCode by entering a value in dbo.u\_dsi\_configuration table with CfgName = 'OverrideWaitMinutes'
  + Set @RunTime to track wait minutes for each pass.
  + Set @LockedRunID to the RunID of the Locked process.
  + If @NextRunID becomes the @LockedRunID (queue moved forward), then reset @RunTime.
  + Set @NextRunID to the RunID of the next process in queue (minimum RunID where Locked = 0)
  + Wait Delay 5 seconds
  + If Locked is Over the Wait Time set in @OverrideWaitMinutes, then Flag Failed = 1.
  + If no processes of same FormatCode are currently Locked, then set the Locked = 1 for the @NextRunID.
  + If @LockedRunID has not completed, then repeat steps above.
* Complete the steps in the switchbox\_v2.
* Release the Lock
  + Update Locked = 0
  + Update Completed = 1
  + Update DateTimeCompleted = Current Date/Time
  + Update DurationHHMMss = Comparison of DateTimeStamp and DateTimeCompleted.
  + Update RecordCount with Count in dbo.U\_dsi\_Count\_v2 for FormatCode.
  + Update FileName with @FileName used for BCP
  + Update FilePath with @ExportPath used for BCP

# Interface Activity Logging

The Interface Activity Log table was designed to work in conjunction with the Locking Mechanism. The custom table, dbo.u\_dsi\_InterfaceActivityLog, allows the ability to track interfaces at various stages of the locking mechanism and wait queue. The Interface Activity Log table also functions as a tracking table, which captures information about the parameters used when the instance/session of that interface ran.

Below is a description of the columns in the Interface Activity Log table:

* **FormatCode** – Unique identifier of an Interface.
* **ExportCode** – Export Session Code associated with a SystemID, which uniquely identifies an Export Session.
* **FileName** – Filename of the Export File
* **FilePath** – Export Path of the File Generated, usually in dbo.u\_dsi\_configuration table.
* **RecordCount** – Record Count to output
* **UserID** – ID of the user that initiated the instance/process
* **PerControlStart** – Start Per Control value
* **PerControlEnd** – End Per Control value
* **COIDList** – COIDList values, usually in dbo.AscExp for BackOffice, different for Process Automation/Web
* **SelectField** – SelectByField values, usually in dbo.AscExp for BackOffice, different for Process Automation/Web
* **SelectValues** – SelectByValues values, usually in dbo.AscExp for BackOffice, different for Process Automation
* **DateTimeStamp** – Date/Time stamp when the process started.
* **Locked** – Flag if process is currently Locked and processing (0 – No, 1 – Yes)
* **Completed** – Flag if process completed successfully (0 – No, 1 – Yes)
* **Failed** – Flag if the process failed (0 – No, 1 – Yes)
* **DateTimeCompleted** – Date/Time stamp when the process completed and lock was released.
* **RunID** – Identity Column, in which value is auto-generated upon insert into table. Unique identified used to determine Lock and Next to Process.
* **SPID** – Server Process ID, assigned by the server at runtime.
* **DurationHHMMss** – The duration of the process from start to end within the switchbox – HH:MM:ss format
* **Engine** – Name of Application used to run the custom interface

**NOTE:** The Interface Activity Log table is setup to only hold a Maximum of 20 Records, there is a process to remove records by Oldest to Newest to maintain a 20 record threshold.

# Initial Settings

## Purpose: You will need to setup the Interface to use Switchbox v2 and run interfaces using the TestSwitchBox v2

1. Load the switchbox\_v2 scripts.



1. Change Interface to run the switchbox\_v2 SP.

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200)

SET @FormatCode = 'EKEYSOL'

SET @sCmd = 'UPDATE dbo.AscDefH SET AdhPreProcessSQL = ''dbo.dsi\_sp\_switchbox\_v2'' WHERE AdhFormatCode = ''' + @FormatCode + ''''

EXEC (@sCmd)

1. Change BuildDriverTable to run the testswitchbox\_v2 SP in the comments section.

dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

1. Ensure any Automated Sessions are setup with the Switchbox v2.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'SCHEDULED'

# Functionality: Test Path

## Purpose: Allows File Path assignment for Test Export Sessions containing "TEST" or "TST" in the ExportCode (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path.

1. Create TestPath in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @TestPath VARCHAR(200)

SET @FormatCode = 'EKEYSOL'

SET @TestPath = '\\ez2sup4db01\ultiprodata\Tanya\Exports\Test\'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''TestPath'', ''V'', ''' + @TestPath + '''

'

EXEC (@sCmd)

1. Create Export Session called ‘TEST’ in dbo.AscExp table



1. Run the testswitchbox\_v2 to validate the file is generated to the ‘TestPath’.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: On Demand Path

## Purpose: Allows File Path assignment for “ONDEMAND”, "ON\_DEMAND" Export Session (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path.

1. Create OnDemandPath in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @OnDemandPath VARCHAR(200)

SET @FormatCode = 'EKEYSOL'

SET @OnDemandPath = '\\ez2sup4db01\ultiprodata\Tanya\Exports\Test\'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''OnDemandPath'', ''V'', ''' + @OnDemandPath + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is generated to the ‘TestPath’.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'ONDEMAND'

# Functionality: OE Path (Open Enrollment)

## Purpose: Allows File Path assignment for Open Enrollment Export Sessions starting with "OE" in the ExportCode OR Export Code is 'OPENENROLL','PASSIVEOE','ACTIVEOE' (i.e., I:\Exports), but still allowing other Export Sessions (i.e., SCHEDULED) to use “ExportPath” File Path.

1. Create OnDemandPath in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @OnDemandPath VARCHAR(200)

SET @FormatCode = 'EKEYSOL'

SET @OnDemandPath = '\\ez2sup4db01\ultiprodata\Tanya\Exports\Test\'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''OEPath'', ''V'', ''' + @OnDemandPath + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is generated to the ‘TestPath’.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'OEPath'

# Functionality: No BCP

## Purpose: Allows switchbox BCP to be disabled to allow for Multiple Files (AfterCollect)

1. Create No BCP Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @NoBCP VARCHAR(1)

SET @FormatCode = 'EKEYSOL'

SET @NoBCP = 'Y'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''MultFile'', ''V'', ''' + @NoBCP + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is generated to the Multiple Files (ignore BCP).

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: No Empty

## Purpose: Do not create a File if it’s an Empty File

1. Create No Empty Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @NoEmpty VARCHAR(1)

SET @FormatCode = 'EKEYSOL'

SET @NoEmpty = 'Y'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''NoEmpty'', ''V'', ''' + @NoEmpty + '''

'

EXEC (@sCmd)

1. Please note, Header and Trailer records (RecordType 'H' or 'T') will be ignored in RecordCount to determine if file is truly empty or not.
2. Run the testswitchbox\_v2 to validate NO file is generated, if file is empty.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: Upper

## Purpose: Convert all characters to Upper Case

1. Create Upper Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @Upper VARCHAR(1)

SET @FormatCode = 'EKEYSOL'

SET @Upper = 'Y'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''Upper'', ''V'', ''' + @Upper + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the characters in file are Upper Case.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: Count

## Purpose: Allows the Control of the Record Count

1. Create OverrideCount Parameter in dbo.U\_dsi\_Configuration table. The CfgValue must be an Integer.

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @OverrideCount VARCHAR(10)

SET @FormatCode = 'EKEYSOL'

SET @OverrideCount = 100

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''OverrideCount'', ''V'', ''' + @OverrideCount + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the Count is overridden.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: UTF-16 Files

## Purpose: Allows the BCP Statement to generate UTF-16 Files (Use –w for bcp instead of –c)

1. Create IsUTF Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @IsUTF VARCHAR(1)

SET @FormatCode = 'EKEYSOL'

SET @IsUTF = 'Y'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''IsUTF'', ''V'', ''' + @IsUTF + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is in UTF-16 format.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: Sort Fields

## Purpose: Allows the ability to Sort by additional fields (100 characters)

1. Create SubSort2/SubSort3 Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @SubSort2 VARCHAR(100) , @SubSort3 VARCHAR(100)

SET @FormatCode = 'EKEYSOL'

SET @SubSort2 = 'EepSSN'

SET @SubSort3 = 'ConSystemID'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''SubSort2'', ''C'', ''' + @SubSort2 + '''

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''SubSort3'', ''C'', ''' + @SubSort3 + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is Sorted as specified.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: Other Control Characters (Override BCP Parameters)

## Purpose: Allows the ability to Override the BCP Parameters that follow Server Name

1. Create BCPParms Parameter in dbo.U\_dsi\_Configuration table

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @BCPParms VARCHAR(50)

SET @FormatCode = 'EKEYSOL'

SET @BCPParms = '-c -t -r "0x0A"'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''BCPParms'', ''V'', ''' + @BCPParms + '''

'

EXEC (@sCmd)

1. Run the testswitchbox\_v2 to validate the file is outputted using BCP Parameters specified.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: CountFilter

## Purpose: Allows the Control of the Record Count

1. Create CountFilter Parameter in dbo.U\_dsi\_Configuration table. The CfgValue is the Filter in the WHERE Clause of the dbo.U\_dsi\_Count\_v2 table.

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(200), @CountFilter VARCHAR(1000)

SET @FormatCode = 'EKEYSOL'

SET @CountFilter = 'LEFT(RecordSet,1) NOT IN (''''H'''')'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''CountFilter'', ''C'', ''' + @CountFilter + '''

'

EXEC (@sCmd)

1. The Example above was given to exclude Header Records (i.e., Record Set starts with ‘H’) from Record Count.
2. Run the testswitchbox\_v2 to validate the Count is overridden.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Functionality: UTF-8 Files

## Purpose: Allows the conversion of files in the UTF8Path Folder to UTF-8 file format

1. Create IsUTF8 and UTF8Path Parameters in dbo.U\_dsi\_Configuration table. The UTF8Path value is required and used to temporarily store the file for conversion, then moves the file to the ExportPath folder.

DECLARE @FormatCode VARCHAR(10), @sCmd VARCHAR(1000), @IsUTF8 VARCHAR(1), @UTF8Path VARCHAR(1000)

SET @FormatCode = 'EKEYSOL'

SET @IsUTF8 = 'Y'

SET @UTF8Path = '\\ez2sup4db01\ultiprodata\Tanya\Exports\UTF8\'

SET @sCmd = '

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''IsUTF8'', ''V'', ''' + @IsUTF8 + '''

INSERT INTO dbo.U\_Dsi\_Configuration

SELECT ''' + @FormatCode + ''', ''UTF8Path'', ''V'', ''' + @UTF8Path + '''

'

EXEC (@sCmd)

1. The UTF-8 Converter utilizes Powershell v2.0 or higher, which is a built-in Windows application, to encode the file in UTF-8 format. The use of Powershell within switchbox\_v2 was approved by SaaS Security and DBA Team.
2. The UTF-8 conversion process will pull all files in the UTF8Path folder and convert them, then move the converted files into the ExportPath folder. This will allow for UTF-8 conversion of multiple files generated via AfterCollect, as long as the files are generated to the UTF8Path folder.
   1. If you are using an after collect, include -w for widechar in your BCP statement.

SET @BCPStr = 'bcp "SELECT Data FROM ' + DB\_NAME() + '..' + @SourceTableName + '" QUERYOUT "' + @FileName + '" -S' + @@SERVERNAME + ' -T -w -t'

1. Run the testswitchbox\_v2 to validate the UTF-8Files are generated.

EXEC dbo.dsi\_sp\_testswitchbox\_v2 'EKEYSOL', 'TEST'

# Frequently Asked Questions (FAQ)

## 'ZAP' Is Gone!

* You may notice the absence of the ‘ZAP’ Export Session.
* The ZAP is no longer needed, as it’s been replaced with a locking mechanism that is tracked in the Custom Table: dbo.u\_dsi\_InterfaceActivityLog.
* If you do run ‘ZAP’, then you may notice that nothing happens.

## “My Interface is stuck in ‘Locking’ Mode…”

* You may encounter an issue with the Locking Mechanism is holding an interface in a “wait queue” until a previous session completes for the same Interface/FormatCode.
* There is a “Wait Queue” timer that is defaulted to the Average Run time + 3 minutes.
* If you are testing and run into the scenario where a session is stuck in the ‘Locking’ mode and you don’t want to wait an hour, then you can manually Fail the Locked process to clear the queue so your process will run.
* Use the script below to manually Fail a Locked Process:

DECLARE @FormatCode VARCHAR(10)

SET @FormatCode = 'EFORMATCODE';

UPDATE dbo.u\_dsi\_InterfaceActivityLog

SET Failed = 1, Locked = 0

WHERE FormatCode = @FormatCode AND Locked = 1 AND Completed = 0 AND Failed = 0;

## “My Interface Always Fails because it takes over 60 minutes to run on average …”

* The Locking Mechanism was setup with a “Wait/Hold Queue” so that interfaces can run in batches without causing issues for automations such as Payroll Automation.
* There is a “Wait Queue” timer that is defaulted to 60 minutes (1 hour), after an hour, the current Locked process will be setup to Failed to allow the next process to run.
* An optional “OverrideWaitMinutes” parameter was provided to allow interfaces with longer average run times.
  + You can add an “OverrideWaitMinutes” by FormatCode to U\_dsi\_Configuration table see script below:

DECLARE @FormatCode VARCHAR(10), @WaitInMinutes VARCHAR(2)

SET @FormatCode = 'EFORMATCODE';

SET @WaitInMinutes = '90';

INSERT INTO dbo.U\_dsi\_Configuration (FormatCode, CfgName, CfgType, CfgValue)

SELECT @FormatCode, 'OverrideWaitMinutes', 'V', @WaitInMinutes;

* + The “OverrideWaitMinutes” is setup in Minutes and will only override the wait queue by FormatCode.
  + It’s recommended to use this for Interfaces that has an average runtime over the 60 minute default.

## Custom Interface run on Web, but getting “ERROR: AHEmpExport.CreateFile ran successfully but no files were created for PayGroup: , PerControl: 201503311”

* This will occur when the ‘UseFileName’ is set to ‘N’, thus the Web doesn’t know where to create the file.
* Ensure the ‘UseFileName’ = ‘Y’ in U\_dsi\_Configuration table.