Tech Note 506
Alarm DB Logger Status Functions

Topic#: 002232
Created: October 2007

This *Tech Note* explains using several functions that can be configured from InTouch® to monitor the current status of the Alarm DB Logger. The document includes the following information:

Application Versions

The Alarm DB Logger Status functions are supported in the following InTouch versions:

- 8.0 SP2 P04
- 9.0 P03
- 9.5 all patches
- 10.0

Functions List

GetAlarmLogConnStatus( )

Returns

Returns the following database connection status values:

- **-1** if the alarm logger is closing with no connection
- **0** if the connection status is not available
- **1** if connection is active
- **2** if the connection is not available
- **4** if alarm logger is closing with valid database connection

Example

```
AlarmLogConnStatus = GetAlarmLogConnStatus( ); {Where AlarmLogConnStatus is a Memory Integer type tag}
```

GetAlarmLogCacheCount( )

Returns

Returns the integer value of current number of alarms in the Alarm DB Logger cache.

Example

```
AlarmLogCacheCount = GetAlarmLogCacheCount( ); {Where AlarmLogCacheCount is a Memory Integer type tag}
```

GetAlarmLogMaxCacheCount( )

Returns

Returns the integer value of the maximum possible number of alarms in the Alarm DB Logger cache.

Example

```
AlarmLogMaxCacheCount = GetAlarmLogMaxCacheCount( ); {Where AlarmLogMaxCacheCount is a Memory Integer
type tag}
```

Exposing the Functions for Use

These functions are not exposed by default. In order to see and use the functions in InTouch, you must create a set of registry entries using the following steps:

1. Open the Registry by clicking **Start/Run**, typing **Regedit** in the field and clicking **OK**.
2. Navigate to the section **MyComuter\HKEY_LOCAL_MACHINE\Software\Wonderware**.
3. Create a Key under **Wonderware** called **AlarmDBLogStatus**.
4. In the key, create a **DWORD EnableAlarmDBLogStatus** entry and set it to **1**.
5. **(OPTIONAL)** Create another key called **SmartCacheStatusLoggingRate** and set it to **10,000**, decimal.

- **EnableAlarmDBLogStatus** is the setting that enables Alarm DB Logger diagnostics.
- **SmartCacheStatusLoggingRate** is time interval at which the Alarm DB Logger provides the diagnostics to WindowViewer. If this value is not configured, the default is 1 minute.

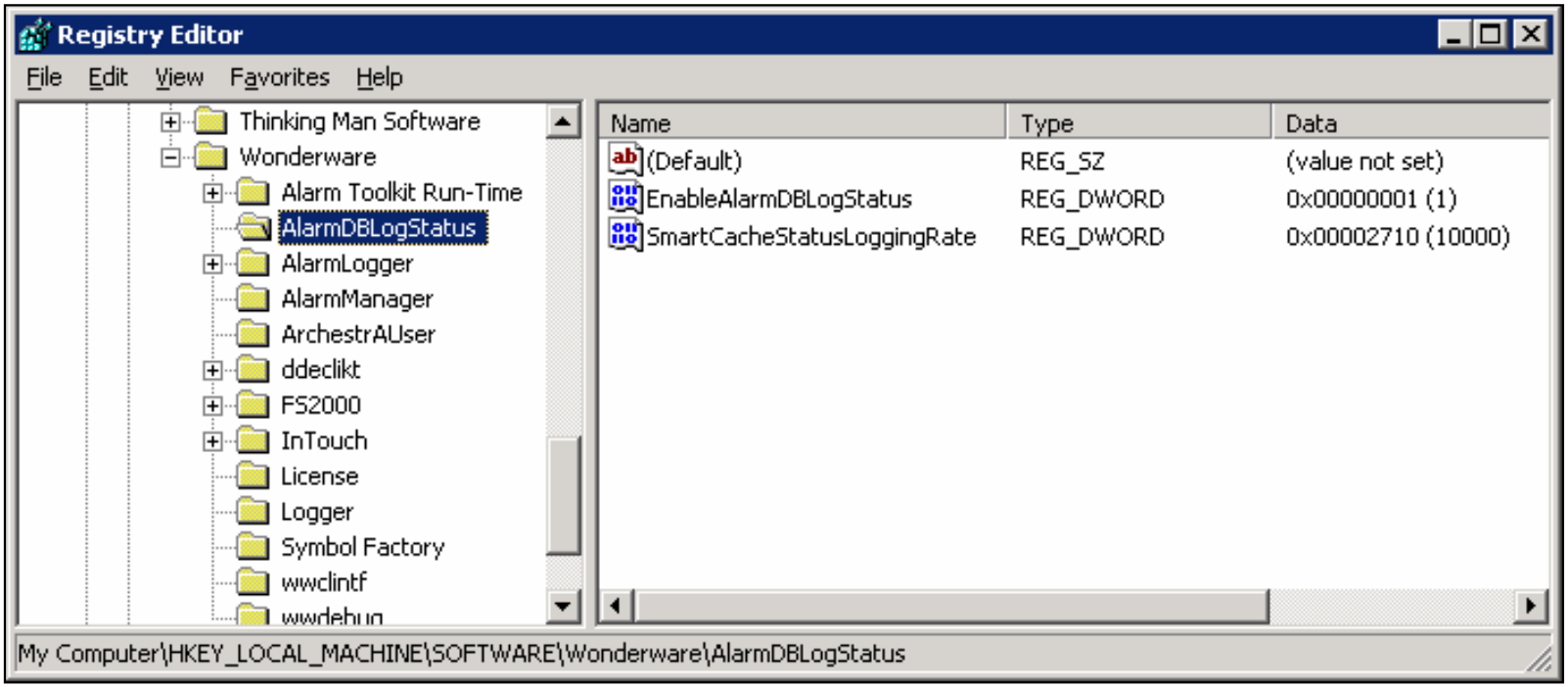

Figure 1: New Registry Keys

6. Close the Registry Editor.

Configuring the Functions from InTouch

Once the registry entries are created and enabled, the functions are visible from InTouch.

1. From an Action script in WindowMaker, select **All** from the **Functions** section.

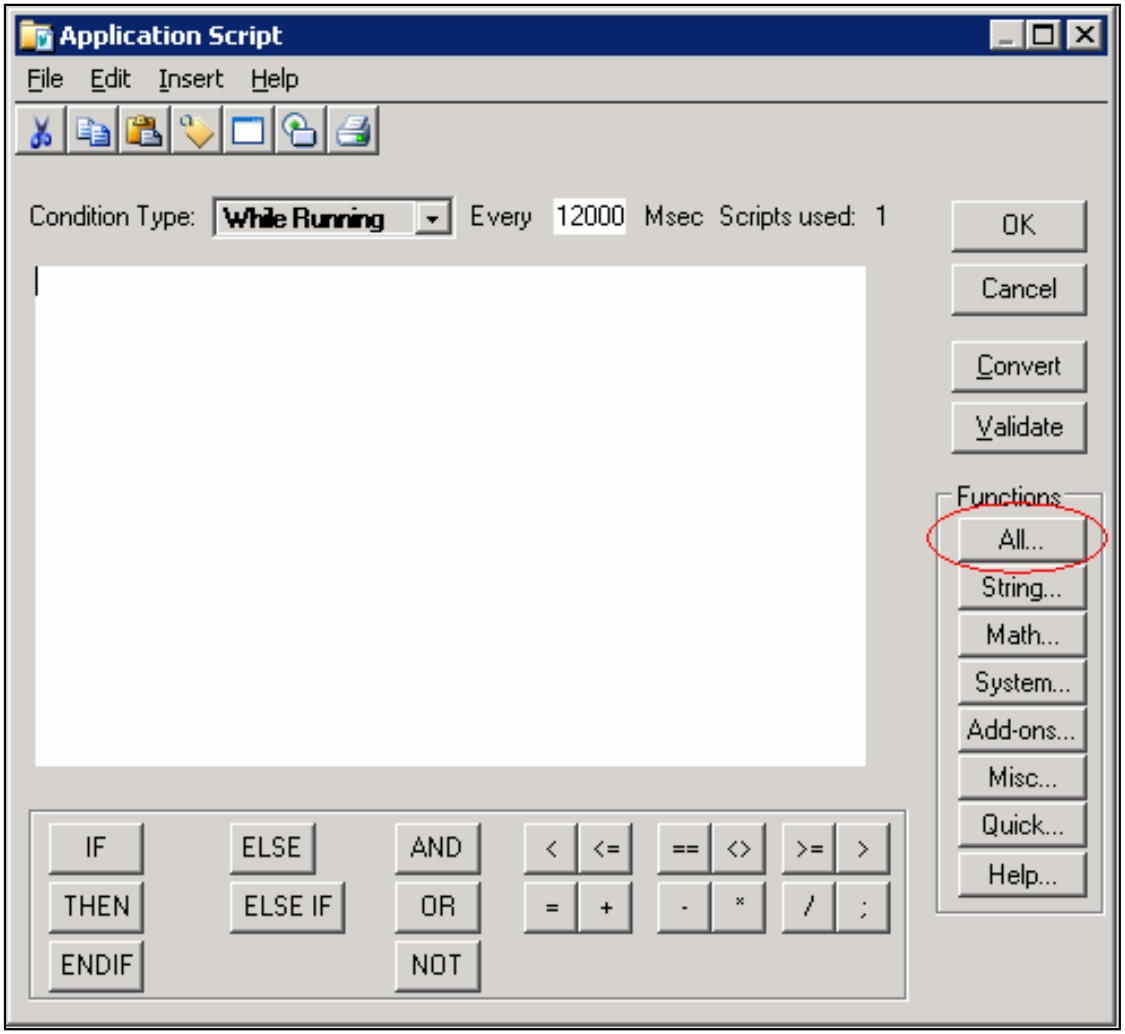

Figure 2: Click the All Button

You can now see the three AlarmLog status functions.

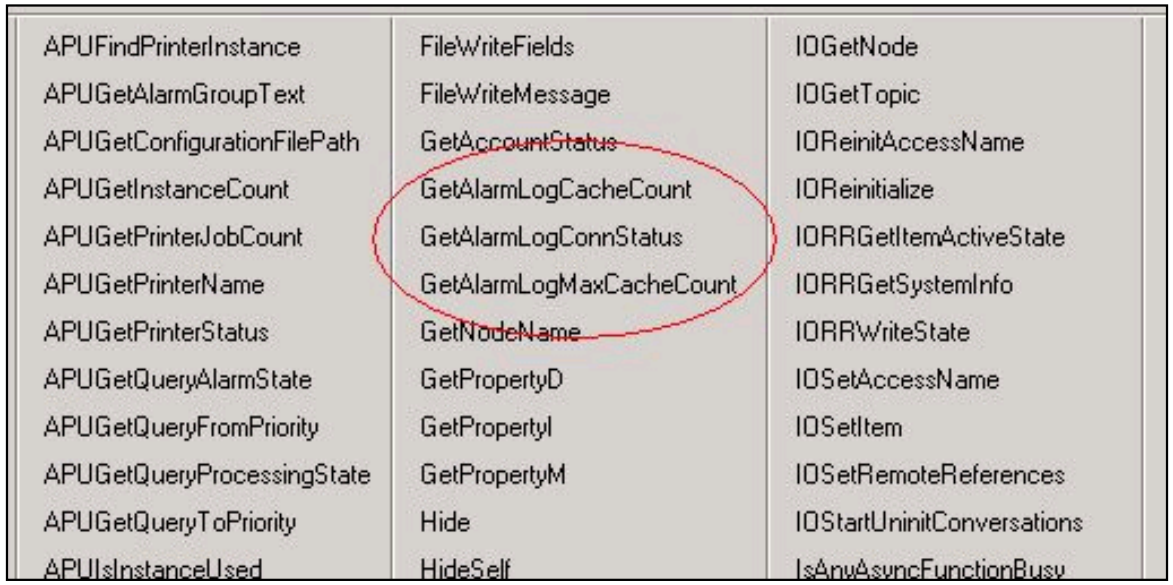

Figure 3: GetAlarmLog Functions

2. Add the functions to the script and link them to the appropriate Memory Integer tags as shown below.

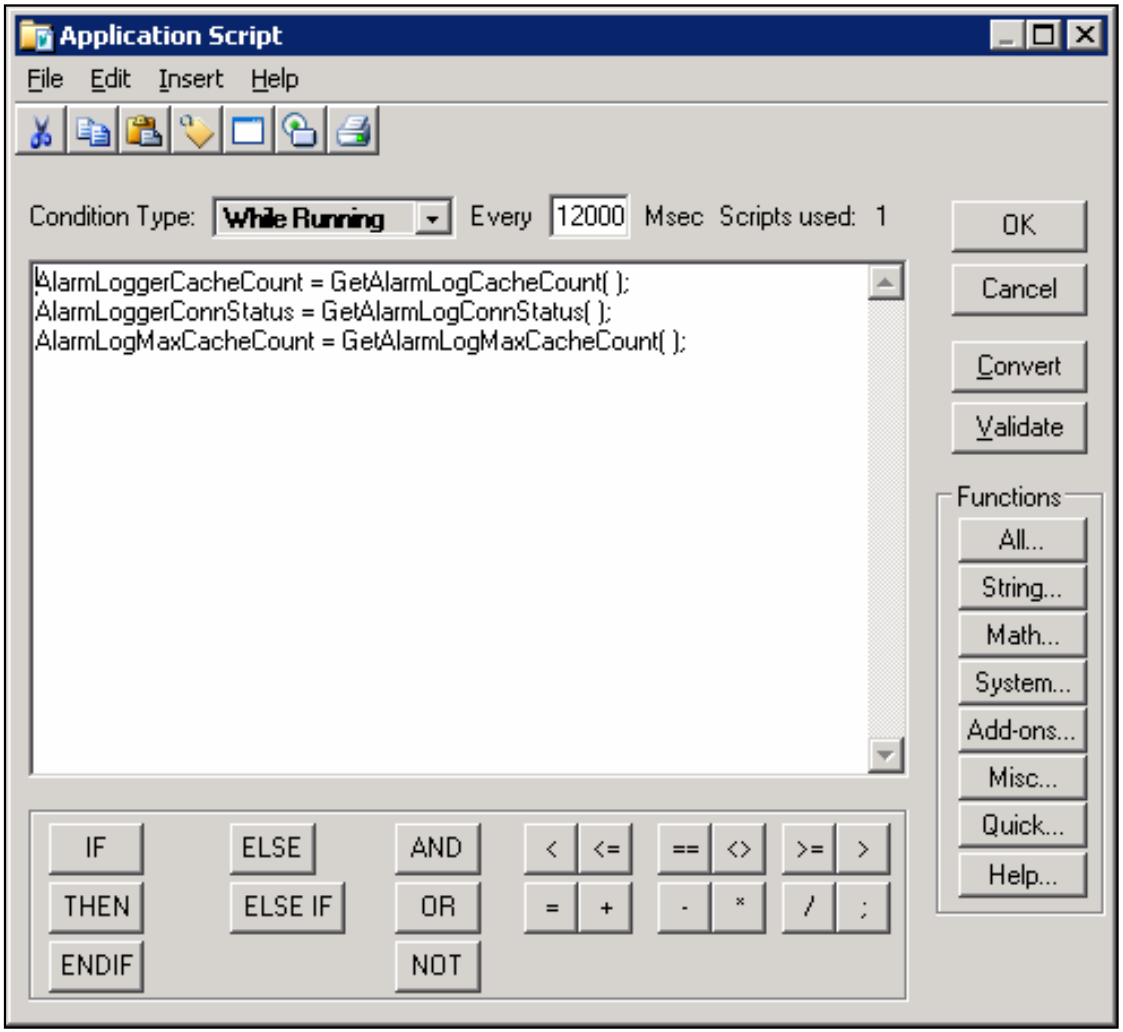

Figure 4: Application Script and Functions

```
AlarmLoggerCacheCount = GetAlarmLogCacheCount( );
AlarmLoggerConnStatus = GetAlarmLogConnStatus( );
AlarmLogMaxCacheCount = GetAlarmLogMaxCacheCount( );
```

Note: Keep in mind that the function returns a value based on the logging rate refresh value configured in the registry.

Expected Results

In runtime, you can expect to see an example of the data returned according to the following conditions:

- The Alarm DB Logger is not started.

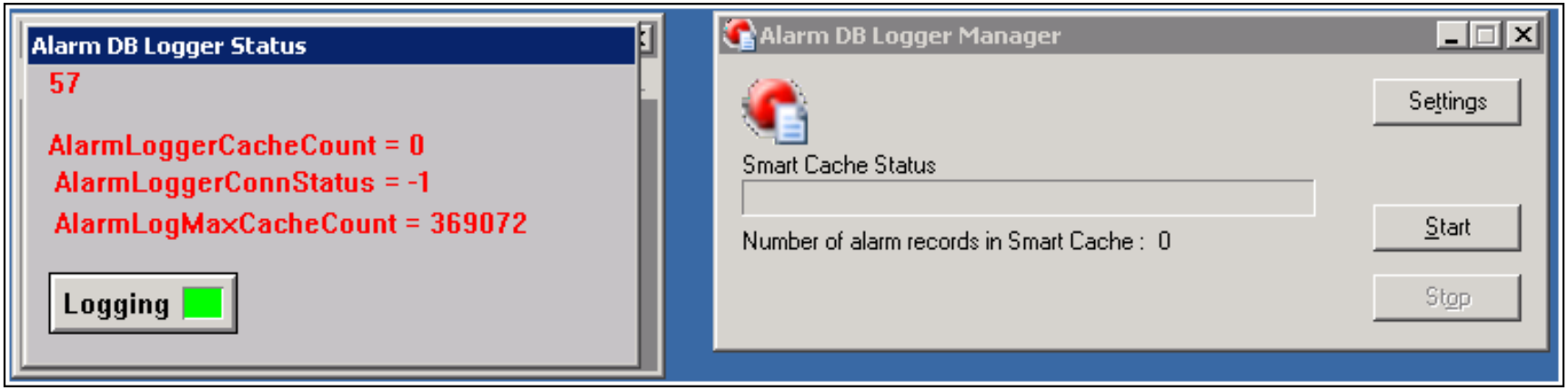

Figure 5: Alarm DB Logger Not Started

- The Alarm DB Logger posts bursts of alarms at a higher frequency than can be posted to the database.

  Notice how the **AlarmLogMaxCacheCount** value is higher than the **Smart Cache** value shown in the Alarm DB Logger Manager status window (Figure 6 below).

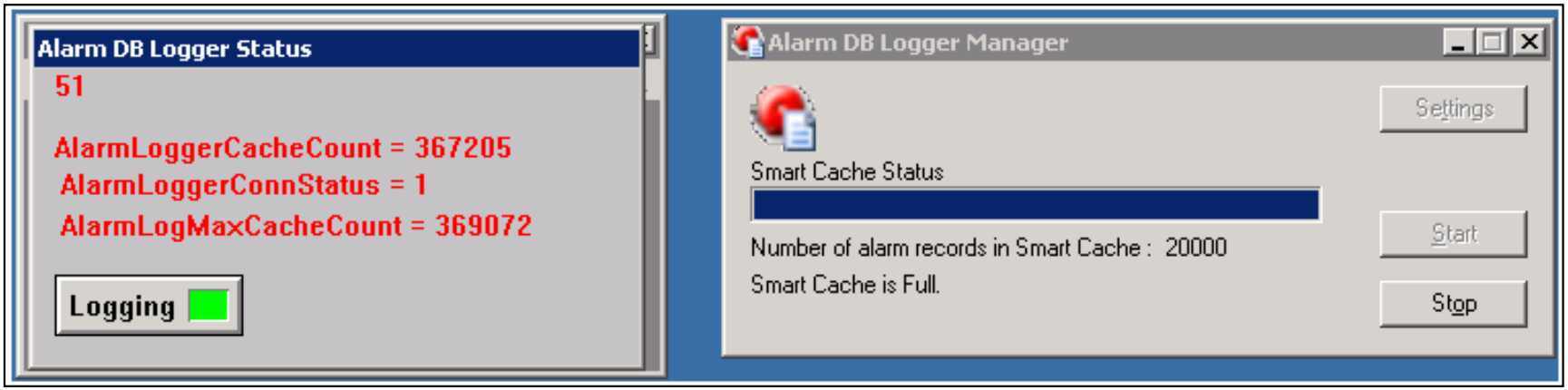

Figure 6: Cached Alarms

- The Alarm DB Logger is configured to store the data on a different node.

  The SQL Server node that stores the data is in the process of being rebooted.

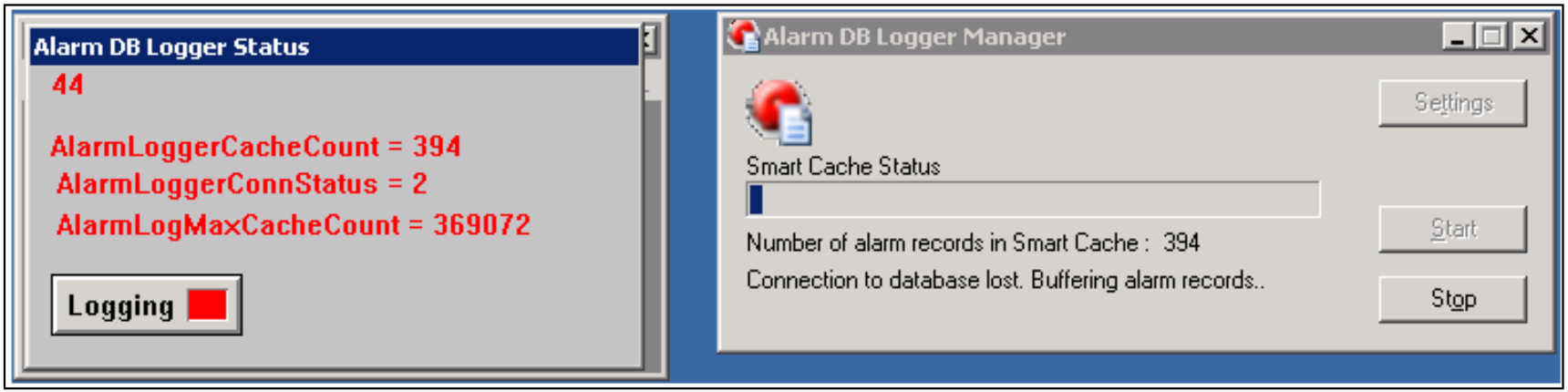

Figure 7: Remote SQL Server Alarm Storage Node

- The Alarm DB Logger is in the process of stopping. It is flushing the alarms that are currently in the Smart Cache.

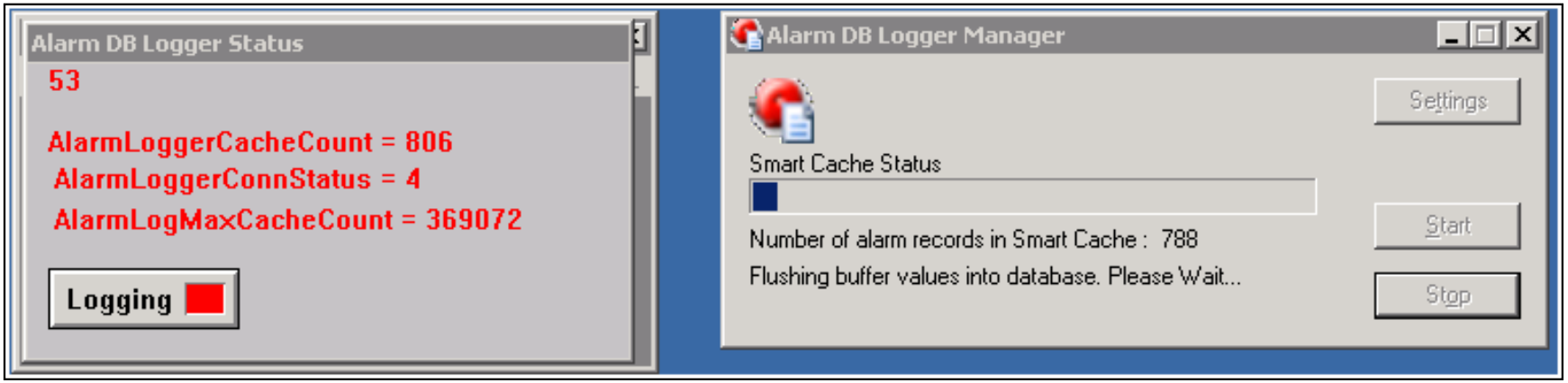

Figure 8: Alarm DB Logger Stopping and Flushing the Cache

Function Limitations

The current limitations for these functions are as follows:

- The functions can only access a local node. Both InTouch and the Alarm DB Logger need to be running in the same node.
- In a Terminal Services Environment, only the InTouch application running in the console can get the status from the Alarm DB Logger running in the console either as an application or a service. WindowViewer running in a terminal session cannot access the status of the Alarm DB Logger running in the console.
- The functions cannot execute from an Application Server script.

Note: The tests performed to validate the functions were completed using **Administrator**, **Power User**, and **Regular User** accounts in a Windows 2003 Server environment. All three user types successfully returned status from the Alarm

DB Logger. Keep in mind that Wonderware typically recommends running InTouch with Power User or higher privileges.

Please contact **product.marketing@wonderware.com** for the status of the change request L00070445 which requests support for these limitations.

A. Martinez

Tech Notes are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at **www.wonderware.com/support/mmi**

For technical support questions, send an e-mail to **support@wonderware.com**.

 **Back to top**