

Windows Installer ログファイルの見方

概要

MSI 形式インストーラは実行時にインストールの情報をログへ出力させることが可能です。この資料では、生成されたログより、アクションの実行状態やファイルの転送状態等を確認する方法について説明いたします。

A. ログの取得

ログの取得方法につきましては、以下の FAQ をご参照ください。

ID:00000214 【Tips】インストールログを作成する

http://hds.networld.co.jp/helpdesk/support/faq_info.jsp?KBNO=00000214

B. アクションの実行状態

ログにはインストーラがインストール時に実行する各種アクションの情報に関しても、実行の有無や実行結果が記録されます。

出力例:

```

MSI (c) (E0:E8): Skipping action: LaunchConditions (condition is false)

MSI (s) (B8:F8): Doing action: WriteRegistryValues
Action start 25:00:05: WriteRegistryValues.
Action ended 25:00:05: WriteRegistryValues. Return value 1.
  
```

- ① … (C) と表示される行はクライアントサイド(ユーザーインターフェイスシーケンス)で実行されたことを表しています。また (S) と表示される行は、サーバーサイド(実行シーケンス)で実行されたことを表しています。
- ② … アクションの処理を表します。 **Doing action** と表示された場合、アクションは実行されます。**Skipping action** と表示された場合、アクションは実行されません。(アクションが実行されない理由としては、アクション自体に実行条件が設定されており、その条件が偽に評価された場合等があります)
- ③ … 実行されるアクション名です。標準アクション名およびカスタムアクション名の両方が記録されます。
- ④ … アクションの戻り値を表します。各戻り値の意味は以下の表の通りです。

返却値	説明
0	アクションが実行されなかった
1	成功時
2	ユーザキャンセル
3	異常終了
4	再起動を待機

C. 機能・コンポーネントの転送状態

MSI 形式インストーラでは、ファイル・レジストリエントリ等の情報は「コンポーネント」に関連付けられていますが、コンポーネントや機能の転送状態もログより確認することができます。

出力例:

① ② ③

```
Feature: ProgramFiles; Installed: Absent; Request: Local; Action: Local
Component: main_exe; Installed: Absent; Request: Local; Action: Local
```

- ① ... 該当の機能・コンポーネントがローカル環境に存在しているかどうかの状態を表します。
 - ② ... 該当の機能・コンポーネントに対して、インストール時に判定された要求を表します。
 - ③ ... 該当の機能・コンポーネントに対して、実際に行われる動作を表します。
- ※ **Request** と **Action** は、ほとんどの場合同じ値となります。

各項目が表す状態は以下の表の通りです。

Installed 値

表示名	説明
Absent	機能・コンポーネントがターゲットシステム上に存在していない
Local	機能・コンポーネントはインストール済み
Source	機能・コンポーネントはインストールソースから実行する方法で、インストール済み
Advertise	機能はアドバタイズされている

Request / Action の値

表示名	説明
Absent	機能・コンポーネントはアンインストールされる
Local	機能・コンポーネントはインストールされる
Source	機能・コンポーネントはインストールソースから実行する方法で、インストールされる
Advertise	機能はアドバタイズされる
Reinstall	機能はメンテナンスモード・マイナーアップグレードとして再インストールされる
Null	どのアクションも機能やコンポーネントに対して行わない

前述しました出力例は、「ターゲット環境に該当の機能・コンポーネントはまだインストールされていないためインストールの要求が行われ、実際にインストールが行われる」ことを表しています。

```
Feature: ProgramFiles; Installed: Absent; Request: Local; Action: Local
Component: main_exe; Installed: Absent; Request: Local; Action: Local
```

● ビルド時に自動追加されるコンポーネントについて

インストーラに追加した覚えの無いコンポーネントがログに記録されている場合、ダイナミックファイルリンク機能によって自動生成されたコンポーネントか、もしくはマージモジュールとしてビルド時にマージされたコンポーネントである可能性があります。

マージモジュールによって追加されたコンポーネントの表示例：

component_name.12341234_5678_5678_9ABC_ABCDEFABCDEF

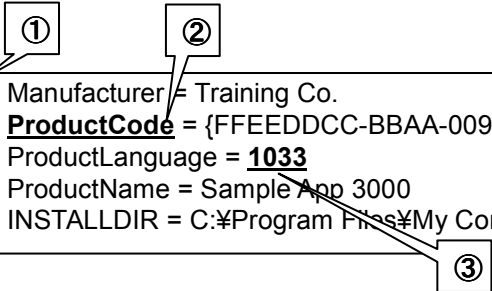
ダイナミックファイルリンクによって自動生成されたコンポーネントの表示例：

_88DD0833EE9EC62582D39C42A9AE8628

D. プロパティの最終値

MSI 形式インストーラがインストール時に使用したプロパティは、その最終値が実行シーケンスの終了時、およびユーザーインターフェイスシーケンスの終了時にログへ記録されます。(通常はログの最下部に合わせて記録されます)

出力例：



```
Property(S): Manufacturer = Training Co.
Property(S): ProductCode = {FFEEDDCC-BBAA-0099-8877-665544332211}
Property(S): ProductLanguage = 1033
Property(S): ProductName = Sample App 3000
Property(S): INSTALLDIR = C:¥Program Files¥My Company Name¥My Product Name¥
```

- ① … (C) と表示される行はクライアントサイド(ユーザーインターフェイスシーケンス)で使用されたプロパティです。(S) と表示される行は、サーバーサイド(実行シーケンス)で使用されたプロパティです。
- ② … プロパティ名を表します。すべて大文字で構成されるプロパティはパブリックプロパティです。小文字が含まれるプロパティはプライベートプロパティとなります。
- ③ … プロパティが格納する値です

● パブリックプロパティとプライベートプロパティ

MSI 形式インストーラが使用するプロパティには、パブリックプロパティとプライベートプロパティがあります。

パブリックプロパティ:すべて大文字で構成される。例:INSTALLDIR

プライベートプロパティ:プロパティ名に小文字が含まれる。例: ProductName

プライベートプロパティは、ユーザーインターフェイスシーケンスから実行シーケンスに移行する際に初期化されてしまいます。ダイアログの入力値等を実行シーケンスで使用する場合、予めプロパティをパブリックプロパティとして作成しておく必要があります。

なお、実行シーケンスで変更されたプロパティ値をユーザーインターフェイスシーケンスに戻すことはできません。例えば、インストーラの最終ダイアログ(SetupCompleteSuccess) で、実行シーケンスで変更された値を表示することはできません。

E. ファイルの転送状態

各コンポーネントによって転送されるファイルの状態についてもログにて確認が可能です。

以下は、初回インストール時、ファイルの転送が行われると判定された場合のログです。

```
File: C:\Program Files\Training\Sample.exe; To be installed; No patch; No existing file
```

以下は、アップグレードインストール時にファイルバージョンが同一のため上書きされないと判定された場合のログです。

```
File: C:\Program Files\MinorUpgrade\UpdateMe.exe; Won't Overwrite; No patch; Existing file is of an equal version
```

なお、MSI 形式インストーラはファイルの上書き時「 File Versioning Rules 」に従って判定を行います。ファイル上書き動作の詳細につきましては、以下の FAQ をご参照ください。

ID: 00000305 Windows Installer サービスはどのようにして上書きを行いますか？
http://hds.networld.co.jp/helpdesk/support/faq_info.jsp?KBNO=00000305

補足： イベントビューアでの確認

MSI 形式インストーラは Windows OSのコンポーネントの一つである Windows Installer より実行されますので、成功・失敗時、一部のステータスに関してはイベントビューアの[アプリケーションログ]に記録されます。

(イベントビューアは、[スタートメニュー]–[コントロールパネル]–[管理ツール] より起動可能です。)

以下は、イベントビューアに記録される情報の出力例です。

The screenshot shows the Windows Event Viewer window titled 'イベント ビューア'. The left pane shows the tree view with 'アプリケーション' selected. The right pane displays a table of 1,658 events. The table has columns for '種類', '日付', '時刻', 'ソース', '分類', 'イベント', 'ユーザー', and 'コンピューター'.

種類	日付	時刻	ソース	分類	イベント	ユーザー	コンピューター
情報	2011/07/20	11:33:15	MsInstaller	なし	11707	Administrator	ISTEAM-
情報	2011/07/20	11:11:19	HHCTRL	なし	1904	N/A	ISTEAM-
情報	2011/07/20	11:11:19	HHCTRL	なし	1904	N/A	ISTEAM-
情報	2011/07/20	11:11:12	HHCTRL	なし	1904	N/A	ISTEAM-
情報	2011/07/20	11:11:12	HHCTRL	なし	1904	N/A	ISTEAM-
情報	2011/07/20	9:01:52	LoadPerf	なし	1000	N/A	ISTEAM-
情報	2011/07/20	9:01:52	LoadPerf	なし	1001	N/A	ISTEAM-
情報	2011/07/20	8:57:49	SecurityCenter	なし	1800	N/A	ISTEAM-
情報	2011/07/20	8:57:46	VMware Virtual Mount...	なし	1	N/A	ISTEAM-
情報	2011/07/20	8:57:45	VMware NAT Service	なし	1000	N/A	ISTEAM-

自己修復時の原因ファイルの特定

イベントビューアの情報が有用なケースとしては、MSI 形式インストーラが意図せずに自動修復を行ってしまうケースが考えられます。MSI 形式インストーラは、コンポーネントのキーファイルが欠損している場合、アプリケーションの起動時に自動的に欠損ファイルの復元を行う自動修復をサポートしていますが、実際に欠損しているファイルの情報はイベントビューアにて簡単に調べることが可能です。

自己修復が行われた場合、イベントビューアには以下の二つの行が記録され、片方の行に欠損ファイルの情報が表示されます。

Detection of product '{11111111-1111-1111-1111-111111111111}', feature 'Main_Program' failed during request for component '{22222222-2222-2222-2222-222222222222}'.

Detection of product '{11111111-1111-1111-1111-111111111111}', feature 'Main_Program', component '{22222222-2222-2222-2222-222222222222}' failed. The resource 'C:¥Program Files¥Training¥sample.exe' does not exist..