



SQL SERVER DAY 2010

SSIS Script component

Doing all the things the SSIS data flow team forgot

Dr. Nico Jacobs

Trainer and consultant @ 

Agenda

- 1 minute intro into SSIS data flow
- Scripting a simple data source:
generate a date dimension
- Dealing with connection managers: RSS source
- Simple synchronous transformation: YTD
- Asynchronous transformation: word breaker
- Multi-stream transformation: Keyword splitter
- Destination: html destination
- Conclusions

Building ETL with SSIS

- SSIS package is an **xml** file with instructions

- A “program”: Package.dtsx

- It needs an interpreter:

- DTEXEC & DTExecUI

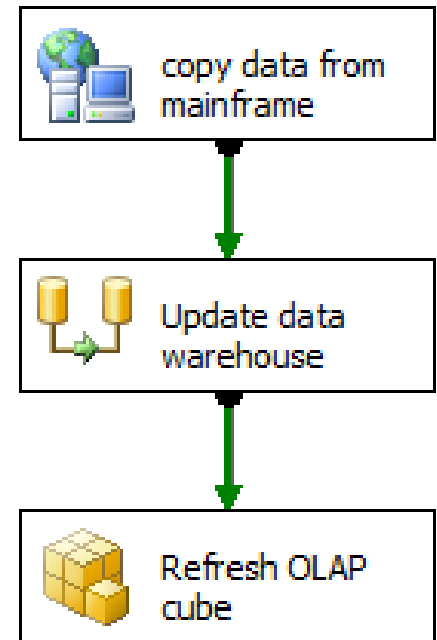
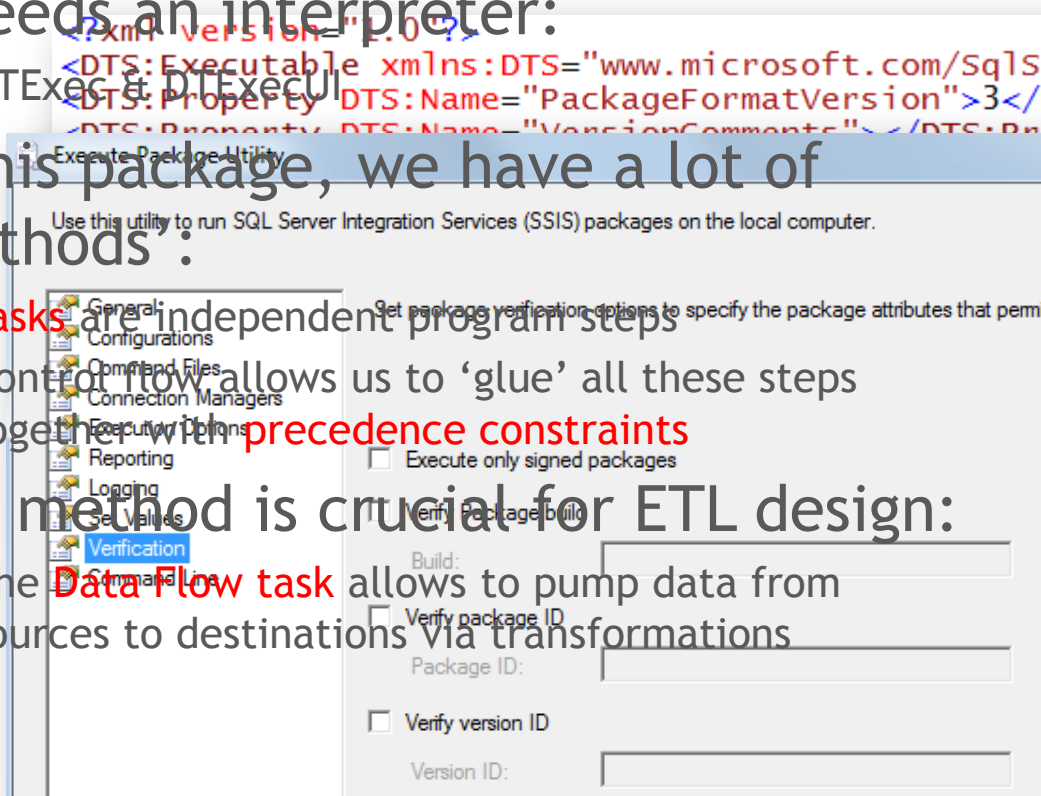
- In this package, we have a lot of ‘methods’:

- **Tasks** are independent program steps

- Control flow allows us to ‘glue’ all these steps together with **precedence constraints**

- One method is crucial for ETL design:

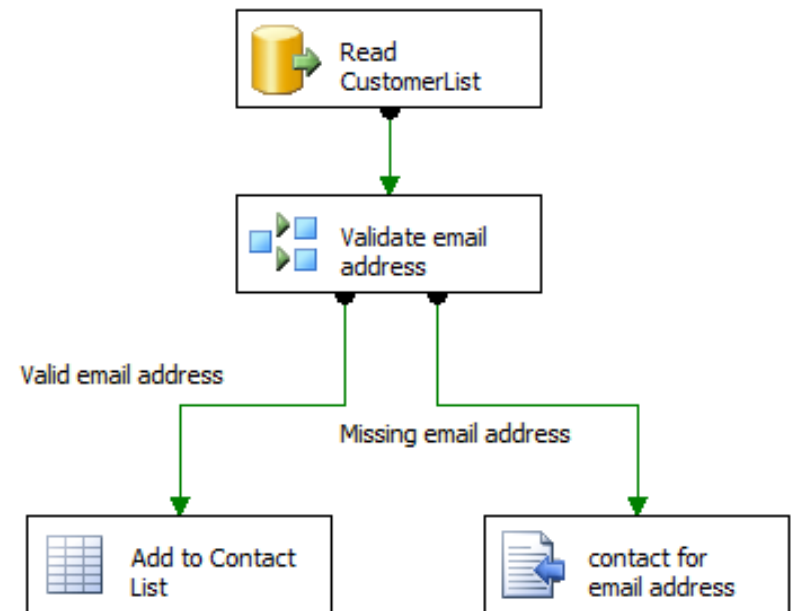
- The **Data Flow task** allows to pump data from sources to destinations via transformations



Data Flow Task



- **Pipeline** metaphor
- Pushes **rows** of data from one or more source components via optional transformations into one or more destinations
- Relational representation
 - Fixed number of columns
 - Named and typed columns



Script component

- Common sources (6), transformations (28) and destinations (12) are pre-programmed components
- But Microsoft left **two DIY options**
 - Using the script component for ad-hoc solutions
 - Building a full-blown data flow component
 - See e.g. <http://www.sqlis.com> or <http://ssisctc.codeplex.com>
- This talk focuses on the script component approach
 - Easier to program
 - Typically only ‘a few lines’
 - No advanced programming skills required
 - Harder to reuse
 - Copy-paste development can be hard to maintain
 - Using DLLs allows us to limit the amount of copy-paste development

Script component

- Three possible uses
 - As **Source**
 - Reading binary files, connecting via non-supported protocols,...
 - As **Transformation**
 - Calculations which require state from previous rows, e.g. year to date, moving average, ...
 - As **Destination**
 - Writing binary files, xml, non-supported protocols, ...

Script a source

- Why would I consume data with an SSIS source script?
- Non-SSIS approach:
 1. Write stand-alone app to convert data to text files or staging tables
 2. Consume data in ETL tool
- Advantages of SSIS approach
 - Performance
 - Less disk storage needed
 - Strong integration (variables, connection strings, events, logging)
 - Only one application to manage and maintain
- Advantage of Non-SSIS approach
 - Wider choice of programming languages

Date source

DEMO TIME!

Operational

	Date	ProdID	SalesAmount
1	2001-07-01 12:30:35.000	310	3578,27
2	2001-07-01 12:34:14.000	346	3399,99
3	2001-07-01 13:52:47.000	336	699,0982
4	2001-07-01 18:03:20.000	346	3399,99
5	2001-07-01 20:29:28.000	346	3399,99
6	2001-07-02 08:05:24.000	344	3399,99
7	2001-07-02 19:06:57.000	351	3374,99
8	2001-07-02 20:29:07.000	310	3578,27

FullDateAlternateKey	TimeKey	FiscalQuarter	FiscalYear
2001-07-01 00:00:00.000	1	1	2002
2001-07-02 00:00:00.000	2	1	2002
2001-07-03 00:00:00.000	3	1	2002
2001-07-04 00:00:00.000	4	1	2002

	DateID	ProdID	SalesAmount
1	1	310	3578,27
2	1	346	3399,99
3	1	336	699,0982
4	1	346	3399,99
5	1	346	3399,99
6	2	344	3399,99
7	2	351	3374,99

RSS source

DEMO TIME!

Source highlights

- VB.Net or C# programming
- Easy access to SSIS variables and connection managers via proxy classes
- Wrapper classes make it easy to insert records in SSIS data flow
- Be careful mapping SSIS data types onto .Net data types

Year To Date transformation

DEMO TIME!

Transformations

- Synchronous transformations are simple
 - Input stream is automatically linked to output stream
 - Transformation loops automatically
 - We can even output a row on multiple output streams
- But what if
 - We need to aggregate data?
 - We need to combine multiple input streams in a single output stream?
 - We need to output multiple rows for a single input row?
- Asynchronous transformations allow us to manually control the reading from input streams and writing to output streams
 - More coding, but more control
 - Increases SSIS memory use

Word breaker transformation

DEMO TIME!

Destinations

- Many useful destinations are not available out of the box:
 - XML
 - HTML
 - SharePoint lists
 - ODBC (2005)

Create HTML destination

DEMO TIME!

A few last remarks

- About .Net
 - SQL Server 2005 uses .Net 2.0
 - SQL Server 2008 and R2 use .Net 2.0 - 3.5
 - We can call custom .Net assemblies as well
 - Signed and uploaded in GAC (on server) or just local (in BIDS)
- About performance
 - Build-in transformations provide better performance
 - Overhead of script component is small (a few %)
- About debugging
 - Breakpoints in data flow script component not supported
 - Though they are supported in the control flow script task
 - Use traditional data viewer approach for debugging
 - Where needed, write events to the log for more detailed information
- About lunch
 - Nearly there, just 3 more slides

Conclusions

- SSIS script component convenient way for ad-hoc data flow extensions
- Little programming effort required
- Transformations can be synchronous or asynchronous
- ExclusionGroup allows us to copy rows to multiple synchronous output streams

Evaluation & slides



Watch your mailbox tomorrow ...

Fill in the evaluation and

win a free ticket for the Techdays 2011

Thanks!



Time is an illusion
Lunchtime doubly so...
Douglas Adams

KOHERA

element  61

 ORDINA

UNISYS

 Global Knowledge™

 avanade®
Results Realized

Microsoft