LANSA

Newsletter
March 2012

LANSA Composer 4.0

LANSA Composer version 4.0 introduces many new features and capabilities.

Some of the highlights are described in the following sections:

1. Installation.
2. Processing Sequences Capabilities and Editing Enhanced.
3. Activities – New and Revised.
4. Transformation Maps New and Enhanced Features.
5. Transaction Document Processing.
6. LANSA Composer Request Server.
7. LANSA Composer Client New and Enhanced Features.
8. Other New and Enhanced Features.

1. Installation

The option of installing LANSA Composer into an existing LANSA system is no longer offered for new installations. (Existing systems can be upgraded provided they are at the prerequisite levels.)

The LANSA Composer Windows client installation is considerably faster and more robust for this version.

LANSA Composer version 4.0 is built with LANSA version 12SP1 at EPC level 871 and requires a run-time environment at least at that level. In addition it requires LANSA Integrator at EPC level 869. This is of concern only to existing users who have installed LANSA Composer in an existing LANSA system. The upgrade of standard LANSA Composer installations (in their own LANSA run-time system) will upgrade all the necessary components.

Altova MapForce version 2012SP1 is provided on the media. It is strongly recommended that MapForce is upgraded to this level on all LANSA Composer clients.

# 2. Processing Sequence Capabilities and Editing Enhanced

Processing Sequences have significant new capabilities and the Processing Sequence Editor has been enhanced to be easier to use. Refer to the following for more information:

2a. Processing Sequence Editing Improved

A series of improvements have been made to the Processing Sequence Editor to make it faster and easier to use.

Resources tabs improved

- The Activities, Transformations, Processing Sequences and Configurations resources tabs have been significantly reworked to make it quicker and easier to locate and select a desired item. This includes a new text search item that will search the definition names and descriptions for matching items.
- The Activities, Transformations, Processing Sequences and Configurations resources tabs now offer the capability to print/preview the item definition (using existing printing support). This makes the full definition of an item of interest available in the Processing Sequence Editor.
- The Activities, Transformations and Processing Sequences resources tabs now include a dynamic list of the items presently used in the Processing Sequence.
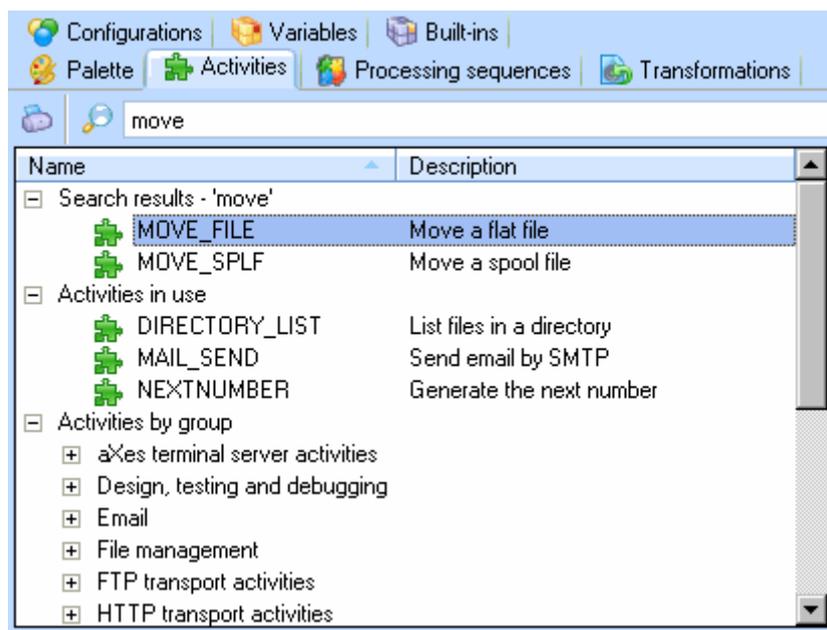

- It is now possible to drag and drop configurations from the Configurations tab directly onto the Processing Sequence Editor. When this is done, LANSA Composer will insert a default corresponding activity, referencing the

configuration where possible, in the appropriate activity parameter. This feature is more useful for some configuration types than for others. For example, dropping an FTP inbound configuration will insert the FTP_INBOUND activity, specifying the selected configuration as the value of the FTPCONFIG parameter. Other configuration types such as database configurations yield a less-useful result.

Processing Sequence Editor view options

A new Processing Sequence Editor Options dialogue permits the view in the editor to be customised, including the ability to show or hide lines connecting the items, to show or hide the sequence number column and to colour-code the Processing Sequence items according to their nesting level.
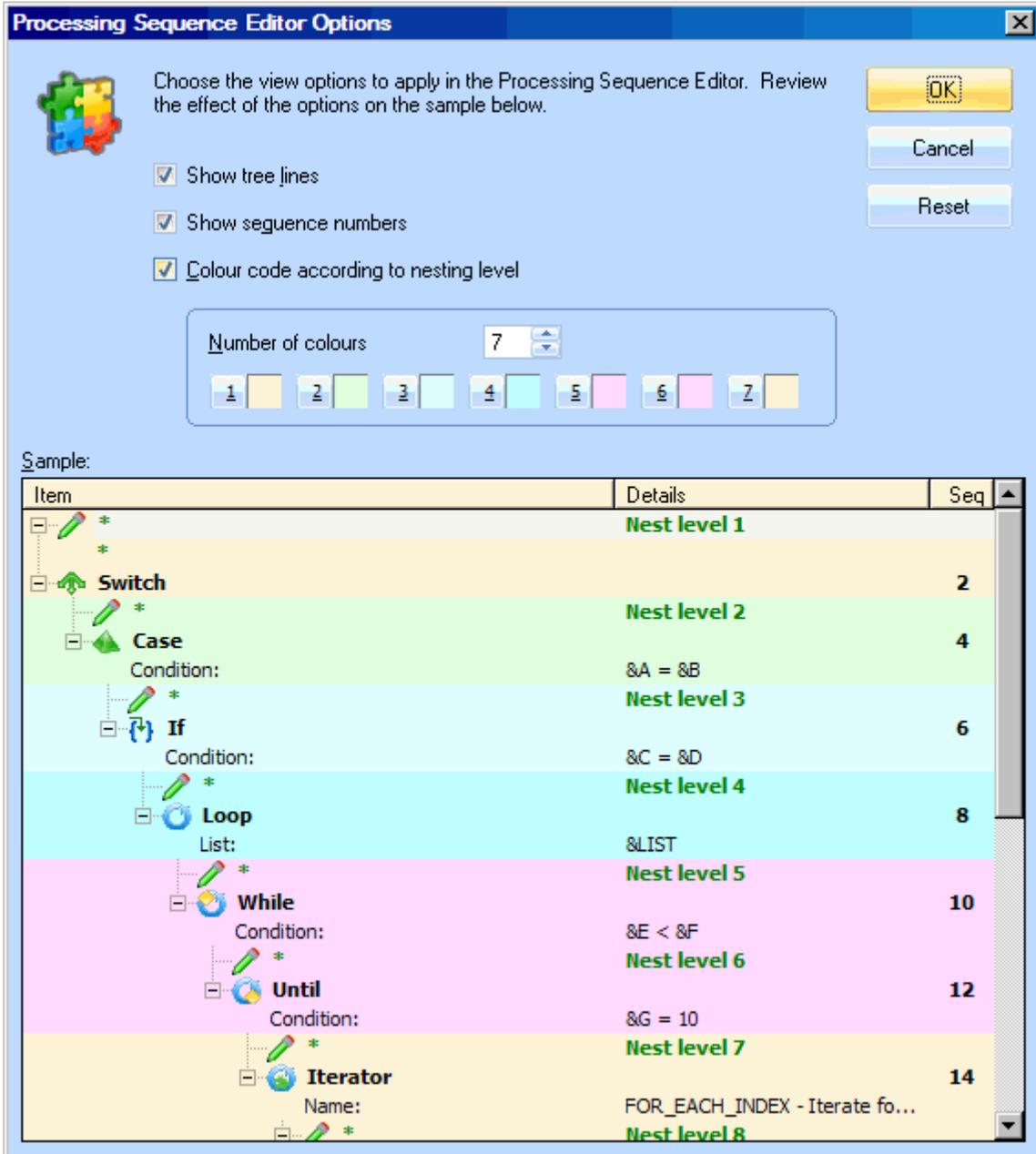
Assistant tab shows formatted help from the LANSA Composer Guide when applicable

The existing Notes tab has been renamed Assistant. It continues to display notes associated with selected activities, transformation maps, processing sequences and configurations. When processing sequence directives are selected on the Palette tab or for activities supplied with LANSA Composer, the relevant topic in the LANSA Composer guide is now shown.

Edit Processing Sequence notes in the Editor

A new Notes tab is added to the main editor pane that allows entry or modification of the notes for the Processing Sequence being edited. Previously Processing Sequence notes could only be entered or modified in the main LANSA Composer client application, outside the Processing Sequence Editor.

Multiple items copy and delete

It is now possible to delete and/or to cut, copy and paste multiple selected items at once. Any shape selection is supported for this, including discontinuous selections and selections spanning multiple nesting levels. When pasting such a selection, LANSA Composer will preserve as much of the structure (parent-child relations) as is included in the selection.

Processing Sequence Editor performance improvements

The improvements will reduce load times and improve general responsiveness during editing.

Other improvements

- Context menus (popup menus or right-click menus) are provided for the major areas of the Processing Sequence Editor window.
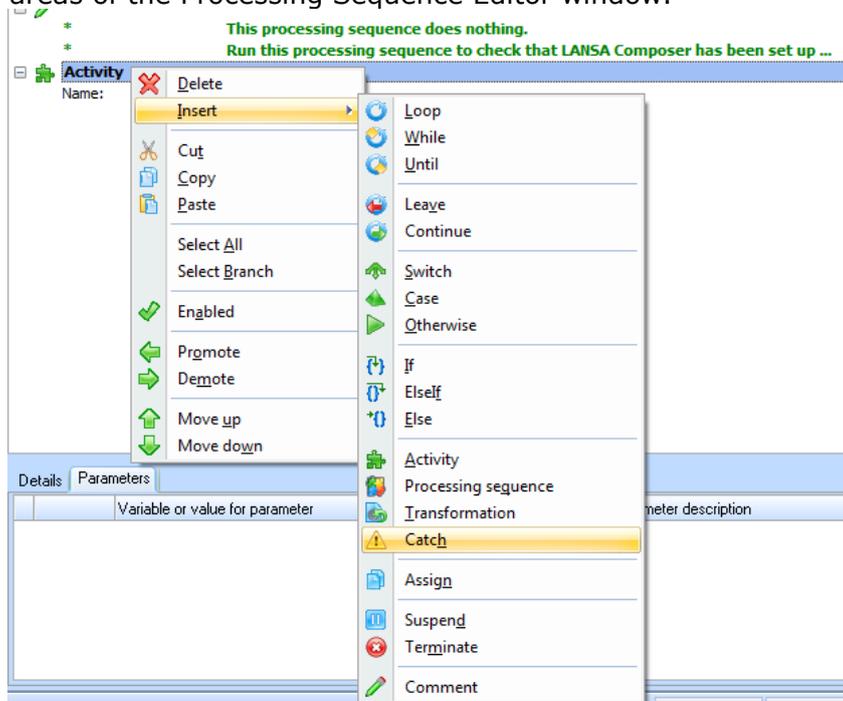
- Some keyboard shortcuts in the Processing Sequence Editor have been revised for better consistency with the main LANSA Composer client application. For example, Ctrl+E selects Activities in both, Ctrl+M selects Transformation Maps and so on.
- New Edit menu commands (Select All and Select Branch) for more convenient selection operations.
- The sequence number and nesting level for the currently selected Processing Sequence item are now shown in the status bar. (The sequence numbers are also now printed when the Processing Sequence definition is printed.)
- Additional information, including the Processing Sequence description, status and event identifier are saved in (and can be restored from) the version history.

2b. Processing Sequence Capabilities Enhanced

Important new process orchestration capabilities have been added to Processing Sequences in this version.

Support for Notification Events by Processing Sequence

LANSA Composer has always included a powerful event mechanism that could be used to generate e-mail or SMS notifications in the event of errors occurring in certain supplied Activities and/or explicitly using the NOTIFYEVENT activity. In this release, you can associate an event definition with an individual Processing Sequence, thereby allowing the use of notification events selectively by Processing Sequence and irrespective of the Activity that caused the error.

New Built-in Variables Permit Enhanced Exception Handling
This release provides a new set of built-in variables that provide additional information about the last error when an exception occurs:

- *lasterror.id: provides the message id of the error message following the last error.
- *lasterror.tx1: provides the first level message text of the error message following the last error.
- *lasterror.tx2: provides the second level message text of the error message following the last error.

These are intended for use in a CATCH block to support further interrogation of the error condition. The "last" error message is defined as the FIRST error message logged after an activity sets the error (ER) result status. It is not necessarily the "last" error message logged because an exception frequently logs more than one error message and the later messages tend to become more generic. The most specific error condition information is normally contained in the FIRST error message logged as defined here.

In addition, the new *process_jsmtrace built-in variable (which is writeable) allows a processing sequence to interrogate and/or override the system setting that controls LANSA Integrator tracing.

On IBM i server, joblog messages are now captured for failed activities
When an activity ends with other than normal result code, LANSA Composer captures the joblog messages that were issued during the execution of the failed activity. The joblog messages can be displayed and printed in the Processing Sequence Log displays (in both the LANSA Composer client application and in the web-browser based Operations Console) in a similar way to the way that the LANSA Integrator trace logs are presently displayed and printed.
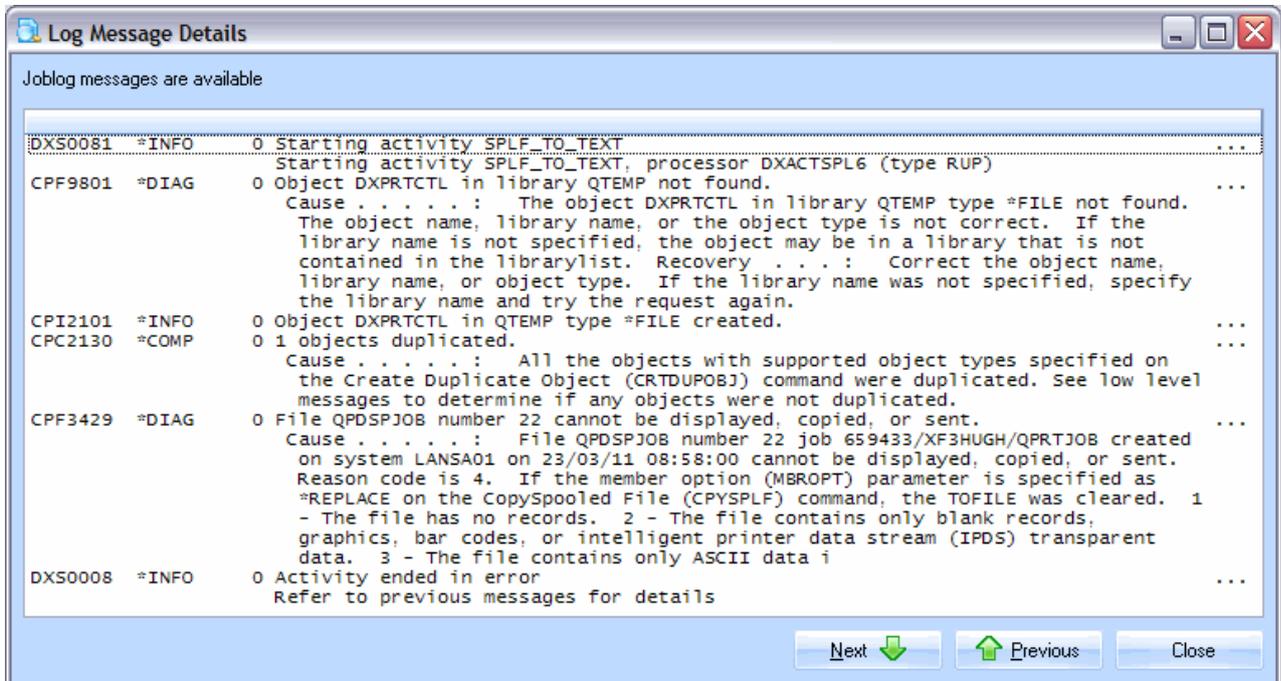
The availability of relevant IBM i job log messages directly in the Processing Sequence Log provides for faster and easier problem diagnosis and resolution in many cases.

SUSPEND and TERMINATE Processing Directives Enhanced

The TERMINATE processing directive may now specify a result code of *Warning*, in addition to *OK* and *Errors*.

The message text for the SUSPEND and TERMINATE directives may now contain embedded references to processing sequence variables and/or built-in variables which will be replaced at run-time with the corresponding variable values.
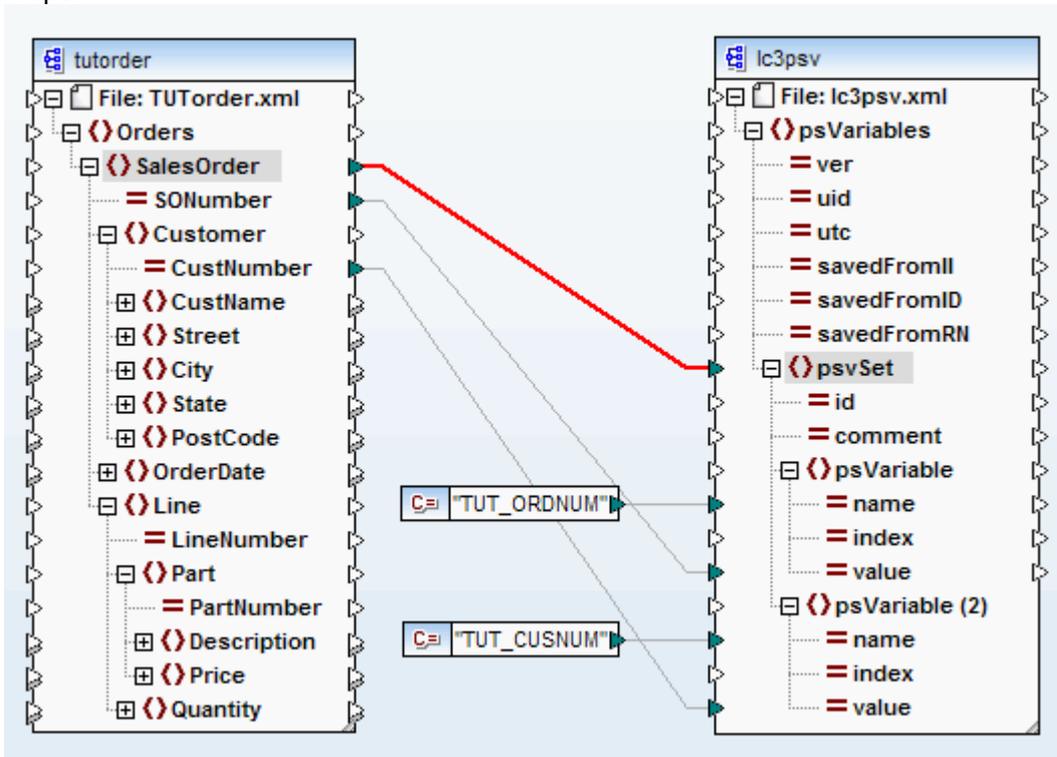

New Processing Sequence Variables (PSV) File Format Defined

LANSA Composer defines a document format to hold or receive Processing Sequence variables and variable lists. Documents that conform to LANSA Composer's format may be used with the new LOAD_PSVSET and SAVE_PSVSET activities and in Transformation Maps.

Mostly, processing sequence variables are used to hold variable data that is used to orchestrate the process - for example, paths to transaction documents that are being processed. Sometimes, however, it is necessary to define processing sequence variables that are used to hold application data items, such as the current order or customer number.

The new PSV file format and the related activities provide new options for loading (and saving) the values of processing sequence variables in these cases.

Batch Automation of IBM i 5250 Applications Using aXes Terminal Server
aXes is a separately licensed LANSA product that Web enables 5250 applications by dynamically transforming the 5250 data stream into a graphical interface for display in a Web browser.

This version of LANSA Composer provides a new suite of activities that are designed to permit a Processing Sequence to connect to and interact with an aXes Terminal Server to navigate, populate and interrogate IBM i 5250 screens. Using the aXes Terminal Server activities, a Processing Sequence can interact directly with existing IBM i 5250 applications running on the same or a different server system.
The Processing Sequence can execute activities to perform such operations as:
- Connect (and logon) and disconnect the 5250 session.
- Interrogate the values of current screen identifiers and field values.
- Set the values of 5250 screen entry fields.
- Send function keys such as ENTER and F3.
- Perform multiple operations with a single activity by using aXes terminal operation scripts.

The LANSA Composer guide contains a new chapter on this topic containing important information and considerations relating to the use of the new aXes Terminal Server activities.

3. Activities - New and Revised
This version of LANSA Composer is supplied with a range of new and revised Activities that deliver new flexibility and power "out-of-the-box":

3a. New Activities
The following new Activities are added in this version of LANSA Composer:
- The new **XSL_TRANSFORM** Activity provides the capability to apply an XSL transformation to an XML input file to transform it to, say, HTML or RTF. You might then deliver the transformed output via one of LANSA Composer's existing transport activities.

- The new **XML_SPLIT** activity can split an XML document into multiple documents at a specified element name.

- New **MSGQ_SEND** and **MSGQ_RECEIVE** activities support sending a message to or receiving a message from a named message queue on an IBM i server. They provide another option for inter-process communication, either between two processing sequences or between a LANSA Composer process and other processes. In addition, they permit processing sequences to interact with (especially to interrogate) existing message queues such as the QSYSOPR message queue.

- New **LOAD_PSVSET** and **SAVE_PSVSET** activities support loading and saving processing sequence variables from or to a Processing Sequence Variable (PSV) file. This support provides a new means of populating and exchanging data in processing sequence variables.

- A new suite of activities permit a Processing Sequence to connect to and interact with an aXes Terminal Server to navigate, populate and interrogate IBM i 5250 screens. This support for aXes Terminal Server is mentioned separately in this document. There are twelve new activities in this suite including, for example, **TS_CONNECT**, **TS_DISCONNECT**, **TS_SETBYNAME** and **TS_SEND**.

- The new **LAST_SPLF** activity (for IBM i servers only) retrieves the identity of the last spooled file created in the current job. In many instances, this new activity is the easiest and quickest way to retrieve the details necessary to perform further operations on a spooled file created by a prior step in the same Processing Sequence.

- The new **SPLF_TO_TEXT** activity (for IBM i servers only) converts a spooled file into a text document. The FORMAT parameter allows you to choose the format that is applied to the spooled file data when it is written to the text document. Amongst other things, you may be able to use this activity to mine data contained in spooled files by combining the activity with a Transformation Map that parses and extracts information from the generated text document using the FlexText component of the mapping tool.
- The new **SORT_LISTS** activity can sort one or more variable lists in "parallel" into the order of the values in the list specified by the first specified list.

- **FOR_EACH_TXTLIN** is an iterator activity similar to but simpler than **FOR_EACH_CSVROW**. It is able to process plain text files without being affected by any nominal value separator.

- **FOR_EACH_VAR** is a new iterator activity. It will iterate for each processing sequence variable extant in the variable pool. This activity is mainly intended for diagnostic purposes. It would not normally be used in live, completed BPI solutions. As a diagnostic aid, it will automatically log the variable name, index and value into the processing sequence log for each iteration.

- The new **SUBSTITUTE_VAR** activity formats an input string with variable substitution. It performs a similar function to the existing **SUBSTITUTE** activity but provides an alternate way of specifying the format string by directly referencing processing sequence variables and built-in variables.

- The new **CALCULATE** activity performs a simple arithmetic calculation, given an operator and two operands. It is intended for simple calculations, usually using integer values, of values that may be used in the process orchestration.

- A new **FTP_DIRLIST** activity lists files in a remote directory. It requires an active FTP configuration of any type and will connect to the remote directory as specified. It will then retrieve a list of file names from the remote directory as specified in the Activity parameter or the FTP configuration.

3b. Revised Activities

Several previously supplied Activities have been revised to add functionality and/or to support other new and revised functionality in this version of LANSA Composer:

- The capabilities of the **HTTP_INBOUND** activity have been enhanced so that more dynamic solutions are possible. It is now nominally an iterator activity, although it is unusual insofar as it iterates only once. It is defined as an iterator activity in order to allow any processing sequence directives nested beneath it to process the received content and prepare the response payload, if any. You can nest transformation maps, other activities or whatever directives you need to perform this. The activity's input parameters that specify the response payload are not evaluated until the iteration completes - so your nested processing sequence directives can set their values. In addition, two new parameters have been added, providing additional information about the received HTTP request. The activity no longer issues superfluous warning messages concerning the received content types.

- The existing **HTTP_OUTBOUND** activity has been renamed **HTTP_POST** (existing applications will continue to function as before). The HTTP_POST activity is used to send data to an HTTP server and optionally receive response data. The new HTTP_GET activity is used to retrieve a document from an HTTP server.

  Both activities provide new support for overriding the URI specified in the HTTP outbound configuration. In addition, for advanced applications, both activities have support for overriding the HTTP request method to use something other POST and GET respectively and for specifying additional HTTP header names and values to be included in the request.

  Furthermore, both activities now support host connection and response timeout values that can be specified in the HTTP outbound configuration.

- The **DIRECTORY_LIST** activity provides a new ORDERBY parameter that allows the output list of file paths to be ordered by name or by date last-modified.

- An earlier release of LANSA Composer added support in the **SMTP_EMAIL** activity for lists of email addresses to be specified for the TO, CC and BCC email addresses. However the documentation of the SMTP_EMAIL activity had not been updated to describe this new capability and so not all users may be aware of it. The notes and documentation have now been updated. Refer to the notes for the SMTP_EMAIL activity for more information on this feature.

- The **FTP_OUTBOUND** activity has two new optional parameters. These parameters can be used to specify override archive directories for both successful transactions and those ending in error. If neither is specified, the archive directory paths as specified in the FTP configuration will be used.

- The **FTP_INBOUND** activity has been changed to issue the CHGDIR command to the remote FTP server only if the remote directory is specified in the activity parameters or in the FTP configuration. A companion change was made to the FTP inbound configuration to permit the remote directory to be unspecified. This change permits support for cases where the FTP login is automatically associated with the one and only directory that is accessible to the FTP user.

  In addition, the FTP_INBOUND activity provides a new optional parameter: GETLIKE. This parameter allows override of 'GET files like' specification in the configuration and allows FTP_INBOUND to receive a different set of files than ones

specified by the configuration. If not specified, the value as specified in the configuration is used.

- The existing activities supporting sending and receiving emails have been renamed. The revised activities are **MAIL_SEND**, **MAIL_RECEIVE** and **MAIL_RECEIVEALL** (previously SMTP_EMAIL, POP3MAIL_SINGLE and POP3MAILALL respectively). Existing solutions will not be affected by this renaming - the solutions will simply show and use the new activity names following the upgrade. In addition, the MAIL_RECEIVEALL activity has a new parameter, POP3MAX, for specifying limits to the maximum number of emails retrieved. Both MAIL_SEND and MAIL_RECEIVE are now capable of handling in excess of fifty attachments.

- The **FOR_EACH_CSVROW** activity now has a new input parameter: SEPARATOR. This new parameter is used when a non-comma delimiter is used for the CSV file. If not specified, a default value of comma will be used. Refer to the activity notes for more information.

- New parameters FROMCCSID, TOCCSID and DTAFMT have been added to the **COPY_FILE** and **MOVE_FILE** activities. They correspond directly to parameters of the same name on the IBM i CPY and MOV commands and can be used to convert text between one encoding and another during a copy or move operation. The parameters apply only when LANSA Composer is running on an IBM i server.

- The **MSG_RECEIVE** activity has new input parameters WAITTIME and SELECTOR. The WAITTIME parameters specified the time (in milliseconds) that the activity waits for a qualifying message to arrive. The SELECTOR parameter is added to allow for message selectors to be specified. For more details on how to specify a message selector, please refer to the appropriate IBM i documentation.

- A new **ZIPARCHIVEPATH** output parameter has been added to the **ZIP_FILES** and **ZIP_DIRECTORIES** activities that provides the full path to the zip file. This simplifies further processing of the zip file, for example, to attach it to an email.

4. Transformation Maps New and Enhanced Features

This version of LANSA Composer provides enhancements to LANSA Composer's support for Transformation Maps and new features in the Transformation Map tool.

4a. Transformation Maps Support

More feedback and diagnostics for Transformation Map prepare

This version of LANSA Composer implements a new Prepare Transformation Map dialogue that:

- Seeks confirmation of the request
- Provides feedback via a "ticked" list of steps and a progress bar
- Clearly shows the result of the operation
- Provides access to a new and detailed Prepare log
- Provides a Retry feature when the Prepare has failed
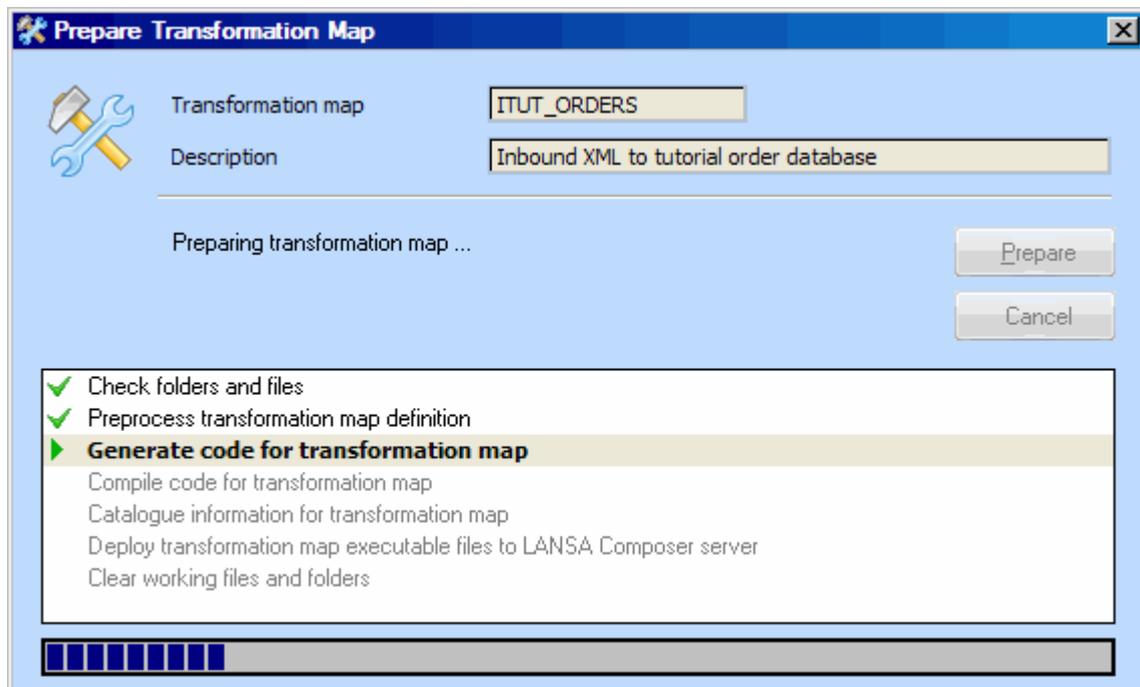
This should make comprehending and troubleshooting the Prepare step much more transparent and accessible.

Transformation Map information catalogued

When a Transformation Map is prepared, LANSA Composer now collects and catalogues a selection of information about the map. This includes:

- The date and time and user name when the map was prepared;
- LANSA Composer, Java and Altova MapForce version information relating to the Prepared map;
- File size and checksum information pertaining to the map definition file(s) (*) and the generated Java executable file;
- The map definition file(s) (*) and the generated Java executable file at Prepare time are saved in the LANSA Composer database (and can be restored if required);
- Selected mapping information collected from the Altova MapForce map definition, including the components used and the database table selections for database components.

(*) This now includes any Altova FlexText configuration files referenced by FlexText components used in the Transformation Map.

A new *Catalogued* command tab in the LANSA Composer client application provides the ability to view the catalogued information for a selected transformation map, including:

- The current Prepared status;
- Information about when and who prepared the map, and the LANSA Composer, Java and Altova MapForce versions that were used;

A visual tree representation of the major components used in the mapping, including database table selections.

This information is also included when the Transformation Map definition is printed and when it is exported and imported.

Note that Transformation maps will have to be prepared with this version of LANSA Composer in order to take advantage of these new features.

Verify Transformation Map files
A new *Check* button on the *Details* and *Catalogued* tabs verifies the current status of the transformation map definition and executable files.

The button displays a new dialogue that determines the file size and a checksum for the current versions of the Transformation Map files (in the file system) and compares them against the file sizes and checksums catalogued when the map was last prepared. Any inconsistencies are highlighted and a Restore function supports restoring the catalogued version of the file from the LANSA Composer database.
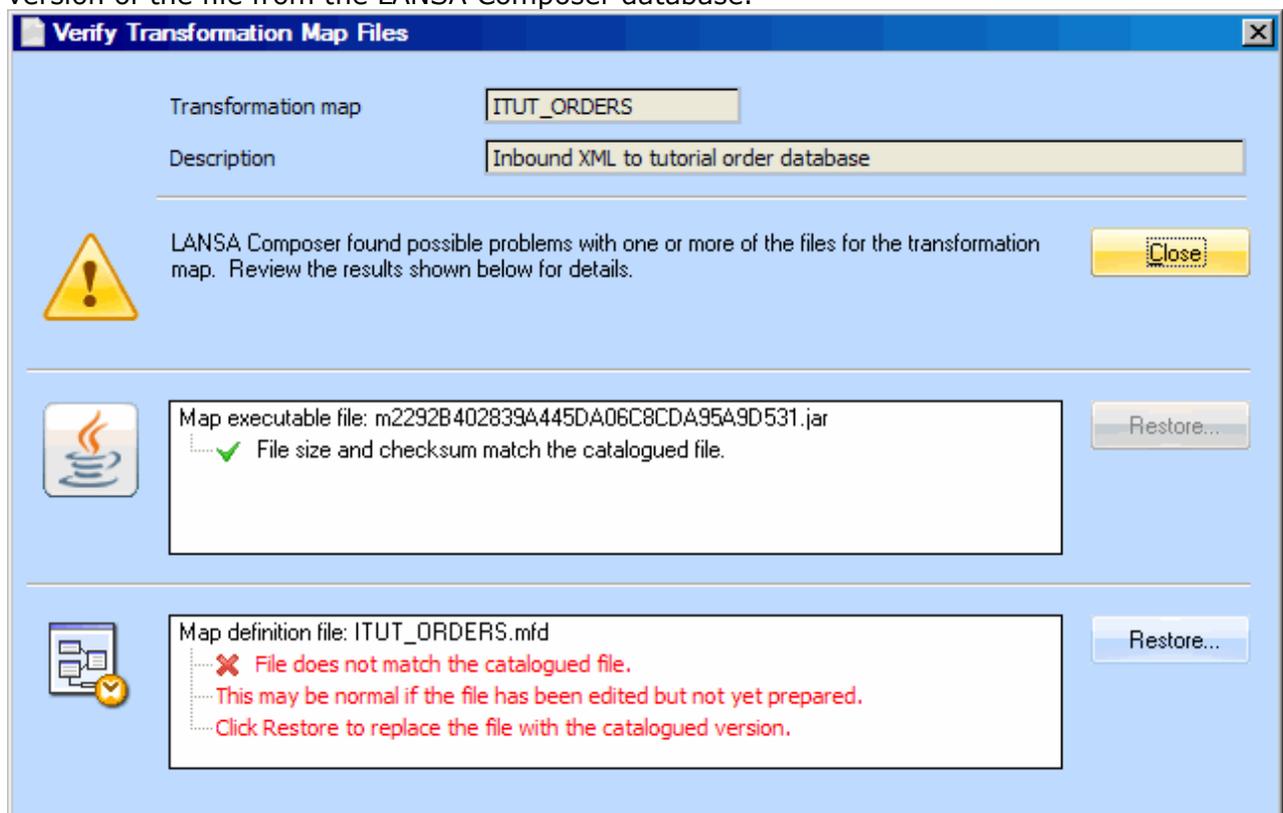
Filter Transformation Maps by database selection
The LANSA Composer client application provides a new filter that allows you to locate Transformation Maps that include reference to a specified database selection (typically a database table name). The new filter is effective at searching only for Transformation Maps that have been successfully prepared with this version of LANSA Composer.

Version history support for Transformation Maps

This version of LANSA Composer implements version history support for Transformation Maps, very similar to that provided for Processing Sequences.

When a Transformation Map is successfully prepared, the prior (prepared) version is archived (subject to a new system setting). This includes:

- The map attributes and data interchange attributes;
- The map parameters;
- The catalogued map information;
- The catalogued map definition and executable files and their checksums and sizes.

A new *Version History* command tab displays the available archived versions and allows them to be restored. Restoring a prior transformation map version restores all the above information, including the catalogued map files.

Network share for LANSA Integrator (JSM) instance folder is no longer required

In previous versions of LANSA Composer, when a transformation map was prepared, LANSA Composer saved the resulting .JAR file in a sub-folder of the JSM instance folder by means of a network share that gave access to that location. A system setting specified the path to use.

This handling has been revised and the network share for access to the JSM instance folder is no longer necessary. (The .jar file is transferred between the client and server through the program-to-program client-server connection instead.)
[At this point, the handling of the transformation map definition (*.mfd) files has not been changed and still requires a network path.]

This change reduces the dependence on network shares and improves the reliability of transformation map *Prepare* and *Import* and *Export* operations. It also simplifies the initial setup of LANSA Composer through the *Server Initialisation Wizard*.

4b. Transformation Maps New Features

LANSA Composer version 4.0 ships with MapForce version 2012. This is an update of the mapping tool provided with the prior version of LANSA Composer. The following summarises the changes that are most pertinent to its use with LANSA Composer:

Intermediate Variables

- Intermediate variables are a special type of component used to solve various advanced mapping problems. They store an intermediate mapping result for further processing.
- Variable results are always sequences, i.e. a delimited list of values, and can also be used to create sequences.
- Variables are structural components, with a root node, and do not have instances (XML files etc.) associated to them.
- Variables make it possible to compare items of one sequence, to other items within the same sequence.
- Variables can be used to build intermediate sequences. Records can be filtered before passing them on to a target, or filtered after the variable by using the position function for example.

New database engine supports direct JDBC connections

It is now possible to connect to supported databases from the MapForce application using JDBC (in addition to the existing OLE DB and ODBC support). Users who, for example, do not have the IBM i Access product installed (and consequently, no ODBC driver for an IBM i database) now have the option of connecting to their IBM i database (while *defining* their map) using JDBC, providing a suitable JDBC driver is installed on the client computer.

XBRL Support extended, including support for US-GAAP 2011

MapForce 2012 maps data to/from XBRL taxonomies or instance files using spreadsheets, databases, or CSV files as input/output documents. You can thus create interim reports, or filter specific data from XBRL instance documents.

MapForce supports the mapping between XBRL documents based on taxonomies such as:
- US-GAAP, including US-GAAP 2011
- IFRS
- COREP / FINREP

Support for SAP iDocs

SAP IDocs (intermediate documents) documents are used to exchange business data between SAP and non-SAP applications. The documents are a form of intermediate data storage which can be exchanged between different systems.

New EDI format IATA PADIS

PADIS (Passenger and Airport Data Interchange Standards) are a set of messages using the EDIFACT (ISO 9735) syntax.

Other new and enhanced MapForce features

MapForce version 2012 adds a number of other new and enhanced features that make your mapping projects easier, faster and more capable:
- Auto-alignment of components in the mapping window.
- New functions: parse-date and parse-time.
- Support for multiple row ranges in Excel components.
- Generation of X12 EDI 999 Implementation Acknowledgement component.
- Find function capability in Library window.
- Reverse mapping.
- Extendable IF-ELSE function.
- Node Name and parsing functions in Core Library.
- Ability to process multiple EDI messages per component.
- Improved database table actions dialog with integrated key generation settings.
- Formatting functions for dateTime and numbers for all supported languages.
- Enhancement to auto-number function.
- New timezone functions: remove-timezone and convert-to-utc.

5. Transaction Document Processing

Several enhancements have been made to the Transaction Document Processing features first introduced in LANSA Composer version 3.0:

- Enhancements and Additions to Transaction Document Processing Activities.
- User-defined transaction document processing statuses.
- Extended Document Type Definitions.

Enhancements and Additions to Transaction Document Processing Activities

- The **DISCOVER_DOC** activity has been modified to provide for additional criteria in Document type definitions. It now has a new parameter that will be matched against Document type group in Document types. In addition, processing can now make use of a document's second extension (eg: myfile.order.xml) as well as file name when classifying files.

- The new **TXDOC_REGOUTX12** activity is a specialisation of the existing TXDOC_REGOUTBND that provides additional parameters to facilitate registering pending outbound EDI X12 transaction documents.

- The **FOR_EACH_TXDOCO** iterator activity provides the ability to select and process pending outbound transaction documents. It has been considerably extended in this version. New inbound parameters provide for extended (but optional) search criteria such as by document type, standard or content type. New outbound parameters will give additional details concerning the document that has been retrieved.

- The new **FOR_EACH_TXDOCT** activity iterates for each transaction set (message) registered for a transaction document and provides identifying information for each.

- The new **TXDOC_KEYS** activity retrieves the staging file keys for a pending outbound transaction document.

User-defined transaction document processing statuses

In version 3.0 of LANSA Composer, the TXDOC_STATUS activity already permitted user-defined document processing status codes to be used. This version adds the capability to define the status codes using the existing LANSA Composer *Code Maintenance* facility. This allows the LANSA Composer Document Manager to display user-entered text descriptions corresponding to the user-defined status codes.
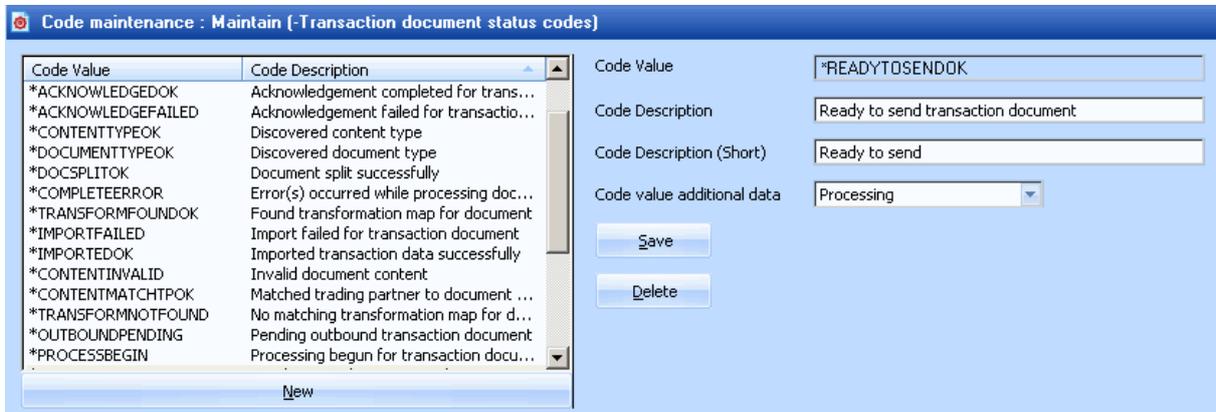
Extended Document Type Definitions

Several new fields has been added to Document type definitions to allow for specification of additional criteria to further classify Document types. These are:

- A secondary extension value: this should match with the second rightmost extension of evaluated documents
- A filename mask to be matched against the filename with the possibility of special characters ? and *.
- A group that can be matched with a group specified by means of a new parameter to the DISCOVER_DOC activity.
- And a sequence number to determine the order of processing.

6. LANSA Composer Request Server

Since version 3.0 and earlier, the LANSA Composer Request Server has provided support for the tightest possible integration between LANSA Composer and other LANSA applications, permitting each to invoke functions in the other, even though they may be executing in different LANSA run-time systems and/or different LANSA partitions, and even if they are running in different versions or EPC levels of the LANSA run-time.

For example, the LANSA Composer Request Server provides the support that allows the CALL_FUNCTION activity in a LANSA Composer Processing Sequence to "call" a LANSA function in a different LANSA configuration and/or partition and exchange variable information with it.

Because such calls between LANSA functions in different LANSA systems are not actually supported by LANSA, the LANSA Composer Request Server "emulates" this support by running the requests in a separate process or job and communicating with that process or job as necessary to pass and receive function or processing sequence details and parameter values.

In version 4.0, this support is extended to allow the "other" LANSA system to be on a different server system altogether. For example, the CALL_FUNCTION activity can now "call" a LANSA function that executes in a LANSA configuration on a separate server. The call still appears to the Processing Sequence to be a synchronous call (if specified) and variable data can be passed in both directions. The server systems involved can be Windows or IBM i server systems on either side (presently the combination of IBM i server to IBM i server is not supported however).

The definition of a LANSA system configuration in LANSA Composer has been extended to support this new functionality.

| Details | Attachments | Notes | Audit |

ID	EXAMPLE_LANSA
Description	Example LANSA system configuration
Status	Active

1. LANSA Composer connects to the request server using:

Connect to request server	On remote system
Keep connection active	Yes
Server type	LANSA Composer server for IBM i
Server LU partner name	SERVER02
Server partition	PRD
Use Windows credentials	No
User	LICPGMLIB
Password	*****
Client-server translation table	
Server-client translation table	
Server exceptional arguments	

2. The request server connects to the other LANSA system using:

|	
LANSA program library name	DC@PGMLIB
LANSA partition name	SYS
LANSA language code	ENG

NB: LANSA Composer must be installed on the target server system in order to use this support, so that LANSA Composer can communicate with components of the LANSA Composer request server on the remote system. In addition, a valid LANSA Composer Remote Request Server licence is required on the remote system.

7. LANSA Composer Client New and Enhanced Features

Several general enhancements have been made to the LANSA Composer client as described in the following sections:

7a. Configure User Access

A new feature in this version of LANSA Composer is the Configure User Access functionality. Using this functionality, access to major parts of the LANSA Composer client application functionality can be restricted for specific users, allowing them only to see specific items in the navigator and/or specific commands for those items.
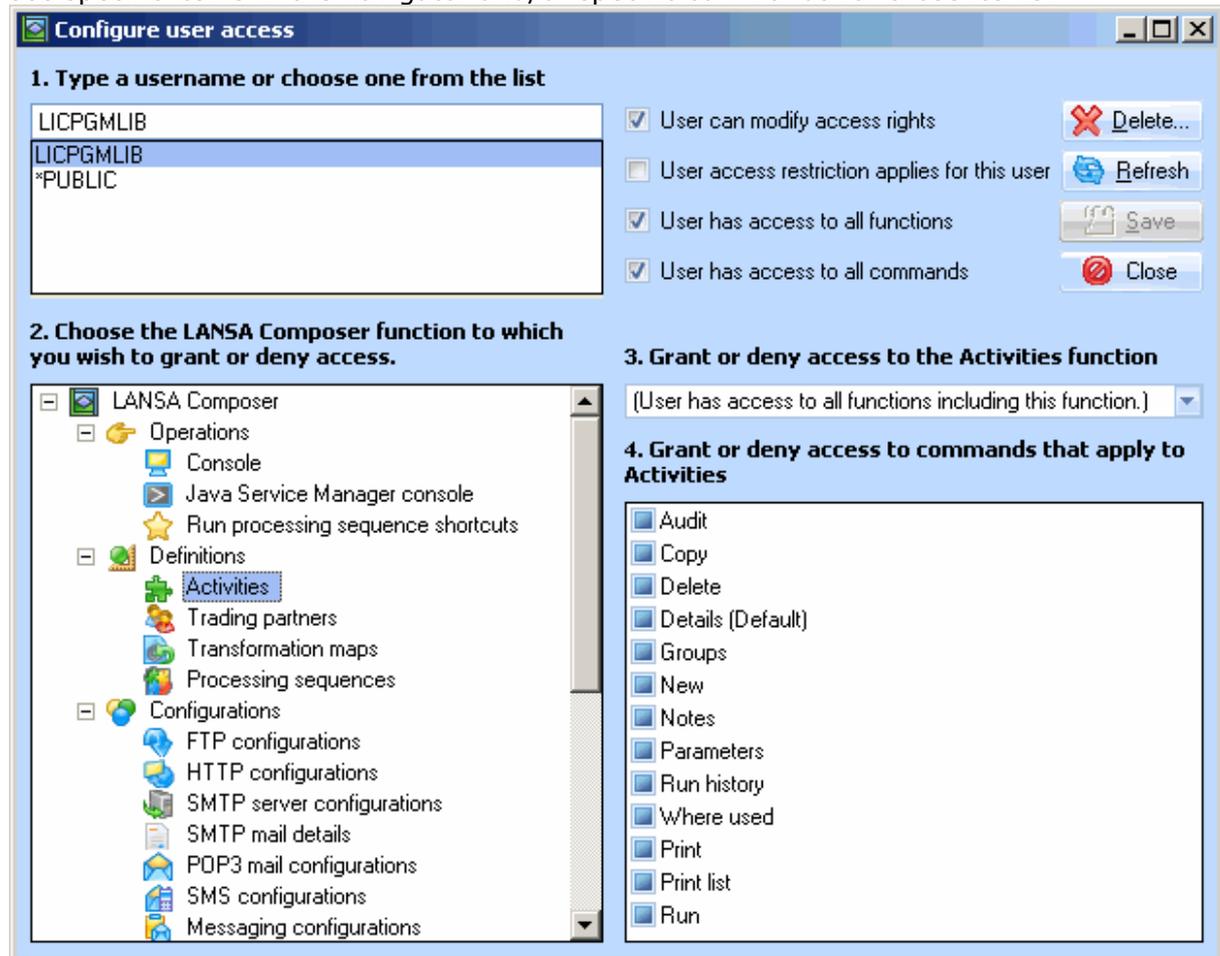
7b. Test Connection in Selected Configuration Types

A Test button has been added for the following configurations: FTP configurations, HTTP Configurations Outbound, SMTP Server mail configurations, POP3 mail configurations and LANSA system configurations. The Test button can be used to test that a connection can be successfully established using the current configuration parameter values.

7c. Generate Skeletal Activity Processor

This version of the LANSA Composer client can generate skeletal RDMLX source code to implement the activity processor for a custom activity.

The generated code includes code to read and write from the variable pool, to interact with the logging services and to load and unload a specified LANSA Integrator service if required. The developer can copy the generated code into their VL IDE and simply complete the application-specific functionality.

Best results will be obtained by running the code generation when the activity definition is complete, particularly with respect to the definitions of the activity parameters.
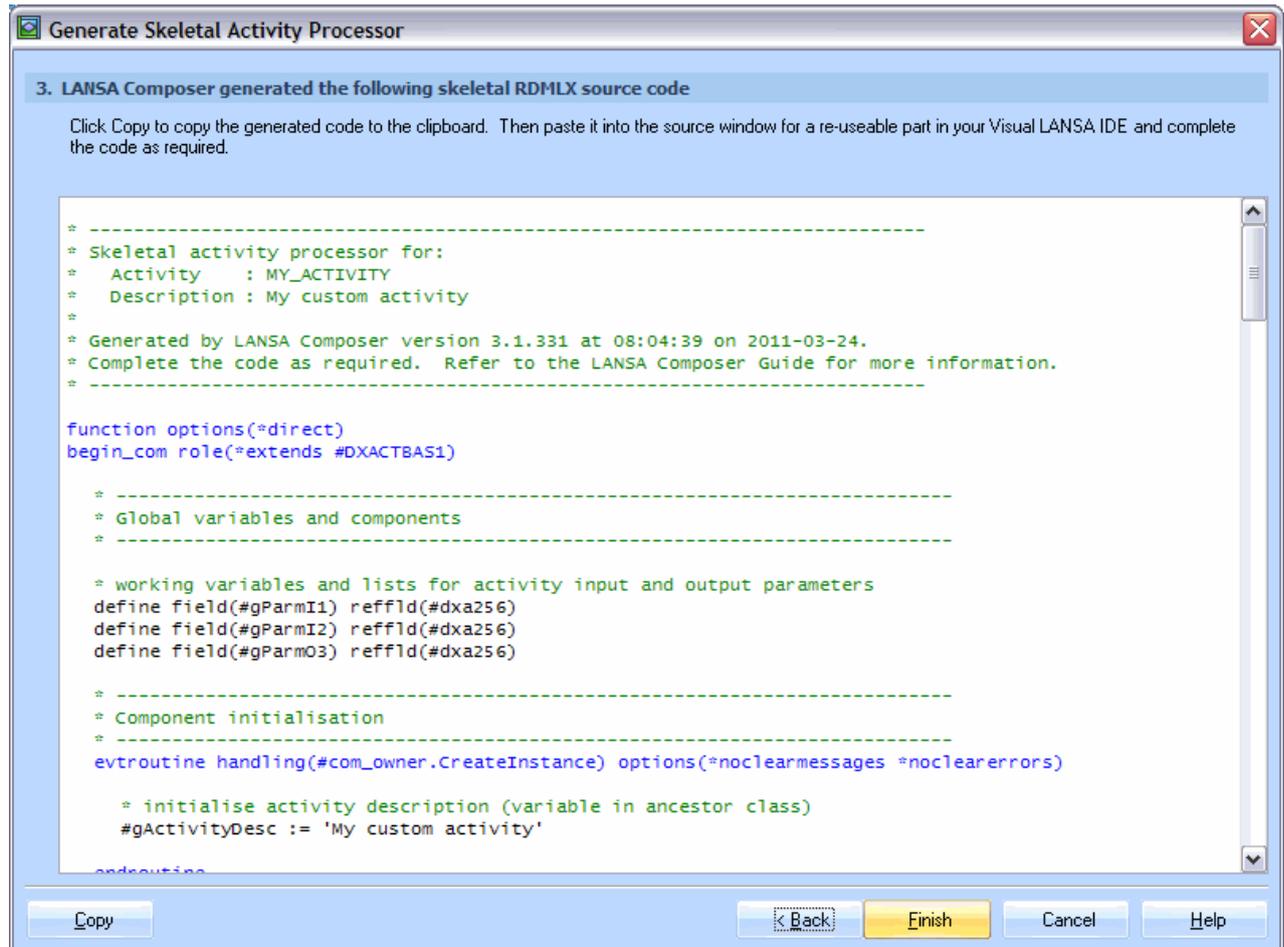
7d. Attach Files to LANSA Composer Definitions

You can now attach files to most integration components in LANSA Composer, including:

- Activities
- Transformation Maps
- Processing sequences
- All types of transport and database Configurations

You can attach any kind of file that you can access through the file system of your computer. LANSA Composer will hold the file attachment(s) in its database, permanently associated with the specific definition to which you attached it. When you export and import your LANSA Composer definitions, the attachments will be included (subject to export and import options).

You might use this feature to attach such things as:
- A copy of an XML Schema used in a Transformation Map;
- A description of operating procedures relating to a Processing Sequence;
- Contact information and/or pictures or copies of trading agreements with a Trading Partner.
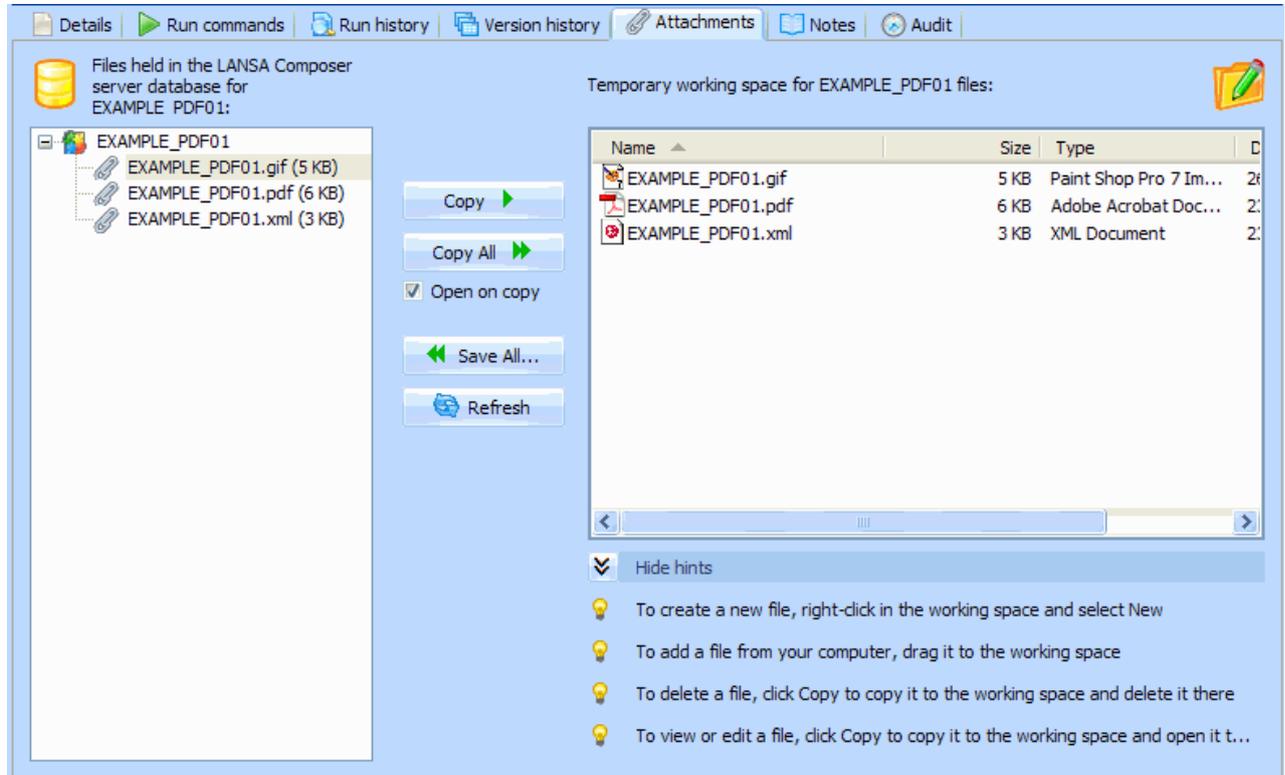

8. Other New and Enhanced Features

Other significant revisions to LANSA Composer in this release are:

Translations and language support

With this version, LANSA Composer is available in Japanese, as well as in English and French. In addition, many corrections, enhancements and extensions have been made in connection with double-byte character support.

Connection and Logon

The *Connect to LANSA Composer* Server window now provides the option to *Use Windows credentials* to use a Windows user profile and password when connecting to the LANSA Composer server. The server must have been configured for Single Sign On and the user enrolled first. In addition, long user names and passwords may now be used when the operating environment supports them.
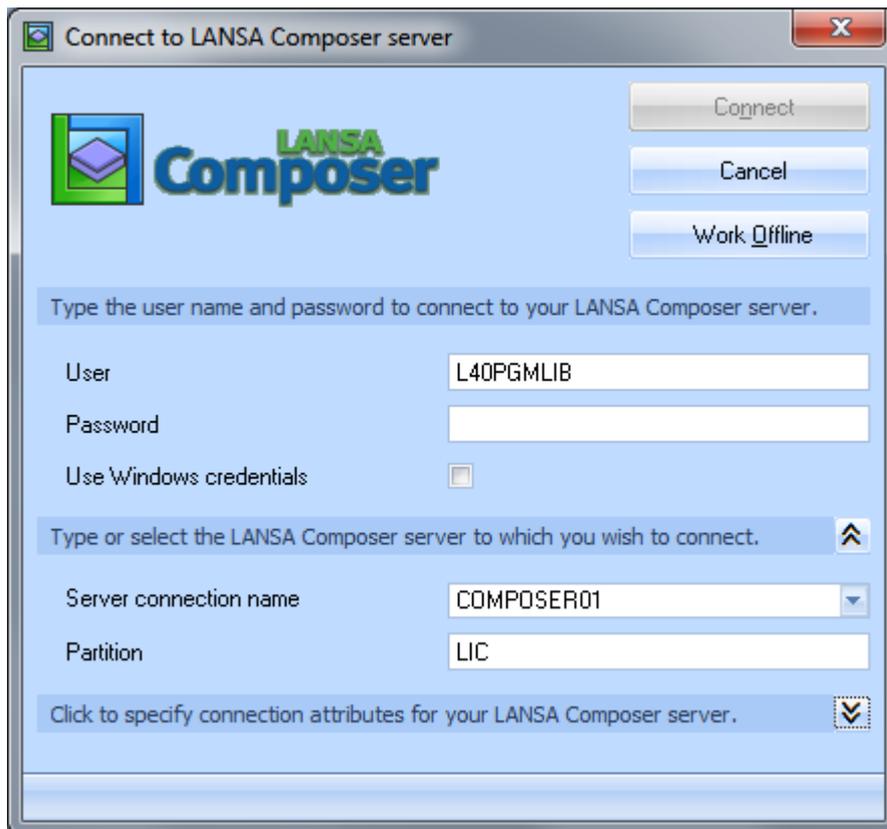
The *Connect to LANSA Composer Server* window now has a Delete button (in the *Connection Attributes* section) that allows you to remove server connection names that you may have previously used but that are no longer required. The Delete button removes the selected server connection name from LANSA Composer's remembered list - it does not delete the corresponding entry in the LANSA Communications Administrator.

Your choice of visual theme in the LANSA Composer client application (select *Appearance* from the *View* menu) is now remembered separately for each LANSA Composer server to which the client connects. This means that you can now use one LANSA Composer client installation to connect to multiple servers and each can have a different visual appearance. (Because of this change, you may notice the visual appearance change *after* connecting to the LANSA Composer server.)

LANSA Composer System Locks

The LANSA Composer client applications, the Processing Sequence engine and the Database Housekeeping tasks all now place a lock on the LANSA Composer system. A new *System Locks* option on the *Tools* menu in the LANSA Composer client interrogates the locks to display the jobs that are using LANSA Composer. In the event that jobs have ended in an uncontrolled fashion leaving unreleased locks on the system, this option may be used to release the locks.

Database Housekeeping

The Database Housekeeping submission will now close the LANSA Composer client after submitting the task. It is recommended that the Database Housekeeping task should have exclusive use of the LANSA Composer system. In addition a simple log file is now generated by this task that confirms the main tasks that were performed.

Support for Entering List of Email Addresses for Trading Partners

Support has been added in the Trading Partner *Details* user-interface to specify a list of email addresses in each of the three provided email address fields.
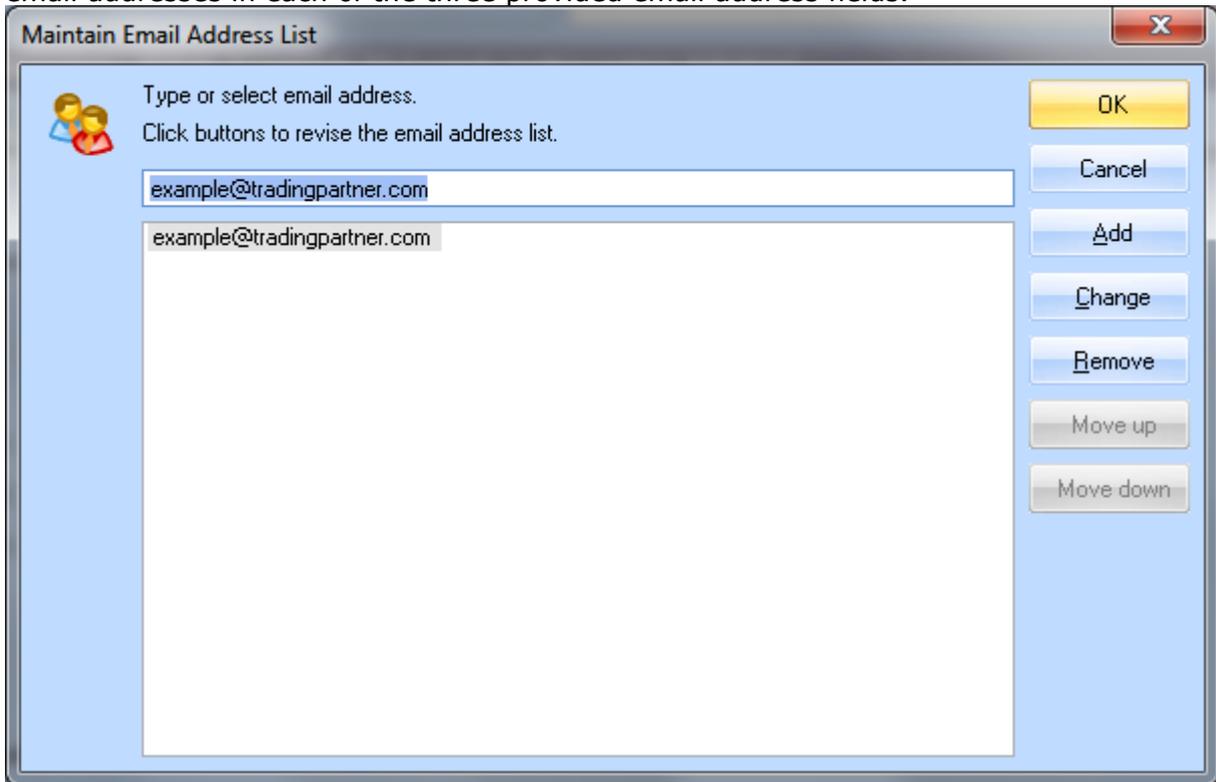

Note that LANSA Composer does not automatically use these fields, but their values may be referenced in Processing Sequences using the built-in variables provided. The email address lists (when entered using the prompter) are formatted suitably for use with the MAIL_SEND activity.

Enhancements to System Properties

LANSA Composer version 3.0 introduced the System Properties feature to LANSA Composer that allowed you to extend the standard LANSA Composer environment with properties that are specific to your own installation, environment or organisation. This release enhances this support to permit you to define system properties:

- that are "writeable", such that your Processing Sequences can assign values to them
- that can be evaluated (and set, when "writeable") by a call to a LANSA function that you define.
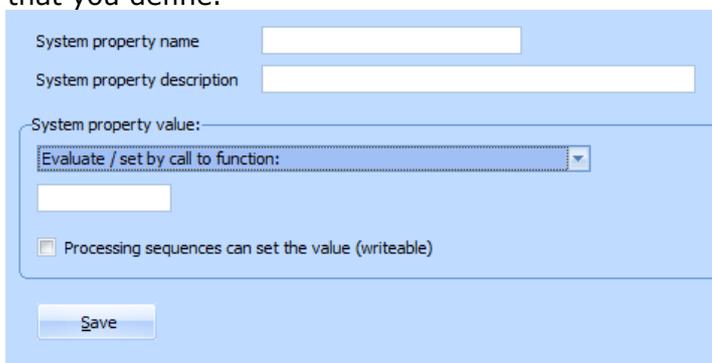
These enhancements add flexibility to the System Properties feature and enhance the potential to integrate your LANSA Composer solutions with your existing applications.

Enhancements to the Processing Sequence Log Viewer

Some enhancements to the *Processing Sequence Log* window to allow more log information to be visible (previously some text was truncated) and to provide a *Restart* button for restarting failed runs directly from the *Processing Sequence Log* window. Also the ability to view the captured IBM i joblog messages, as mentioned elsewhere.

Improvements to the Operations Console User-Interface

Several changes to the user-interface of the Operations Console (LANSA Composer's web-browser based operations interface) enhance its-ease-of-use.
The changes include:

- Submission of Processing Sequence Run or Restart no longer opens in a new browser window. Instead, the Run (or Restart) window is re-displayed, along with the submission message. In the case of Run (not Restart), the user can further alter parameter values and submit a further run (as many times as required). Otherwise, clicking Cancel returns to the previous display.
- Fuller use of the available page height on the Run History display (previously excessive white space was left at the bottom in some circumstances).
- Fuller use of the available width in the "Details" (right-hand) pane of the Processing Sequence Log display (previously only about 2/3 width used resulting sometimes in unnecessary wrapping and reduced clarity).
- LANSA Integrator trace details (and the new Joblog messages details on IBM i servers) are shown in fixed font for better alignment and readability.
- A new About LANSA Composer function provides version and web configuration and support information in the Operations Console. This feature may help in diagnosing web server configuration issues related to the use of the LANSA Composer Operations Console.

Language may be specified in LANSA System Configuration

Previously LANSA Composer requests passed through the LANSA Composer request server used the language code (eg: ENG) in which LANSA Composer is running. Now the language code can optionally be specified as part of the LANSA system configuration. If specified this will override the LANSA Composer language code. This allows LANSA Composer to execute CALL_FUNCTION activity to call a function in another LANSA system using a different language code to that used by LANSA Composer.

Export changes

LANSA Composer now allows you to include other Export Lists in the definition of an Export List and/or in an ad-hoc Export request. LANSA Composer will automatically export the super-set of all the selected items in all the referenced export lists, quietly disregarding any circular references to included Export Lists.

The export window now no longer shows supplied definitions by default. LANSA Composer users would not normally need or want to select these items for export. In the exceptional case where you do want to export supplied Processing Sequences, Transformation Maps or Activities, a new System Setting, *Show supplied definitions for export* will need to be enabled.

Export and Export Lists now show the last-changed user, date and time, where applicable for each item available for export. This makes it easier to identify and select recently changed items.

Shipped Example Definitions

The LANSA Composer server initialization wizard now offers the option not to import example definitions (example Processing Sequences, configurations etc). In addition the example processing Sequences are now no longer flagged as "supplied" definitions, meaning they can be deleted from the LANSA Composer system if desired.

Notes for supplied Activities use the LANSA Composer Guide

When displaying or printing notes for *supplied* activities, LANSA Composer now displays or prints the corresponding topic from the LANSA Composer Guide (lansa091.chm). This results in notes that are better formatted and more readable.

Activity notes are automatically generated

LANSA Composer generates default notes for an activity if no notes have been defined. All shipped activities have notes supplied, but custom activities may not. The generated notes give the most basic description and overview of input and output parameters in the

absence of user-entered notes. The generated notes also provide a starting point and template for user-entered notes.

LANSA Composer help is more sensitive to context

Previously LANSA Composer help simply opened the LANSA Composer Guide at the title page. In this version, it has been enhanced to usually open the guide at a topic that is pertinent to the function that the user is performing. For example, it will open the Activities topic when working with Activities in the LANSA Composer client.

New LANSA Composer Tutorial Orders Application

This version of LANSA Composer provides a very simple LANSA application that provides access to the LANSA Composer Tutorial Orders database. It is provided to provide a quick means to verify tutorial results as well as to provide a sample 5250 application for use with aXes terminal server activities in examples and tutorials.

LANSA Composer updated to use MSXML 6

LANSA Composer has been updated in this version to use MSXML 6 for XML parsing and writing support for internal XML implementations including the *Export* and *Import* features. Previously LANSA Composer used MSXML 4 and when installed on recent editions of Windows, this sometimes caused immediate issues during the *Server Initialisation Wizard* and elsewhere when the MSXML 4 support was not already installed on the computer.

New example processing sequence: Example_PDF01

The processing sequence EXAMPLE_PDF01 has been added to the existing suite of processing sequence examples. This example serves as a demonstration of how LANSA Integrator's PDFDocumentService can be used in a custom-written function or activity invoked by LANSA Composer to generate richly-formatted PDF documents.