Making Your Business Run Better

MetaData

Module Description and Operation

Author: Patrick Jenkins

Version: 2-0

Date: 2015-08-01

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

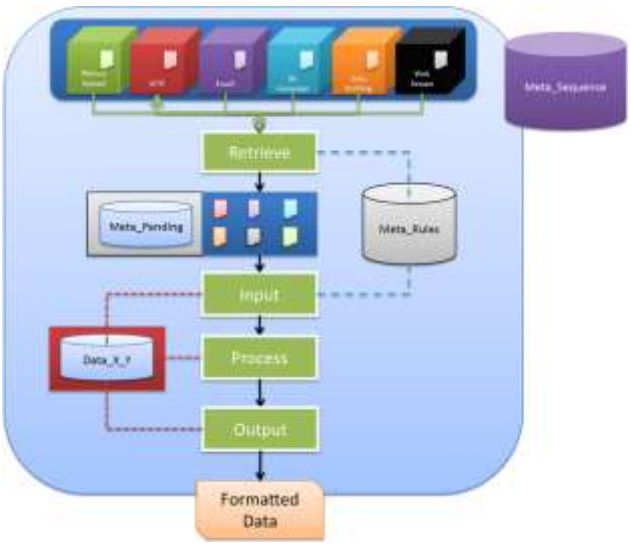
48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

1 MetaData

MetaData from All n One is a dedicated data module in bxp. It allows data from numerous sources to be retrieved, transformed and pushed out to reports and other systems. Each stage of the process allows for custom rules to be applied. The stages are Retrieve, Input, Process and Output.

Combining data from numerous Excel spreadsheets to generate one comprehensive report is now just a click away. Our clients have already begun to use MetaData to save hundreds of operational hours in report generation and replacement of laborious, time consuming tasks. Here are some examples:



Combining data from numerous sources such as phone systems, HR systems, quality assurance scores, and countless other sources, to combine the data and calculate a score which could be loaded into a payroll system are now easily managed.

Using the highly modularised approach, each process is broken down into simple individual steps called rules. Rules are combined up into sequences. Sequences are combined up into Programs. Enabling very complicated processes and procedures to become a single click of a button, or even an automated task. ISO compliance and process management with full audit trails is now a matter of a simple once off setup and guaranteed compliance.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

Have your report sitting waiting for you to collect first thing on Monday, while all of your data retrieval, collation, processing and output formatting is done on a Sunday by bxp. Make Monday meetings more effective with all your reports ready and waiting to go. With automated data outputs now you can easily have a web front end to anything from an AS400 to integrating live stream technologies. bxp makes data processes simpler.

2 Understanding storage

The elephant in the room is that the person using the MetaData module must have an understanding of databases. This module and the following operations require you to have a working understanding of database Structured Query Language (SQL) and also understand what goes into manipulating raw data. This is often done by reporting staff in organisations worldwide and does required a technical skillset that is often considered by numerous organisations as specialised.

Within bxp we use the MySQL database engine. For some of the nuances of MySQL can be found here: http://php.about.com/od/learnmysql/ss/mysql.htm

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

2.1 Acronyms and Abbreviations

2.1.1 All n One

API	Application Programming interface
BDM	Business Development Manager
bxp	Business eXpress Platform
CEO	Chief Executive Officer
COO	Chief Operating Officer
CTO	Chief Technical Officer
Ops	Operations
Snr	Senior
ASA	Cisco Adaptive Security Agent
DP	Data Protection
DPC	Data Protection Commissioner
FTP	File Transfer Protocol
HR	Human Resources
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
SAM	System Access Management
SFTP	Secure File Transfer Protocol
MD	MetaData

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

2.2 Version History

Version	**2-0**
Date	2015-09-07
Author	Philip Lacey
Modifications	Transitioned Id to reflect the massive updates, doc style change and all the work PJ has done.

Version	1-9
Date	2015-09-01
Author	Patrick Jenkins
Modifications	Complete documentation update

Previous version history archived.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

3 Structure of Processing

A rule is an instruction to be performed.

A sequence is a set of rules to be performed on all records or a subset of records.

A program is the sequential execution of a number of sequences

A sequence can belong to more than one program.

4 Types of Processing

Retrieve

Gets the raw data into the bxp infrastructure by what whatever medium available and chosen.

Input

Field mapping and processing rules for extracting data from Excel, structured data or XML documents

Process

Can be considered a number of stages which may / may not be required.

1. Initial data setup
2. Ensuring data is all on one line
3. Data Tidy up
4. Data conversion
5. Data transfer

Output

A sequence of post processing tasks including communications, report generation and follow up tasks.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

5 Development sequence

When building a MetaData program a general build approach can be adopted. This document has been built to follow the general steps in constructing a program. A worked example using a Nortel phone system report is provided to attempt to make easier to understand the stages involved.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

6 Preparation: Create the initial data source storage structure

Before you can do anything with the system you need to have a database to store the data in. If you're using an existing database structure such as an Inbound, Outbound, Blended campaign or Testing Centre Exam or QA program, then you have a structure to work with already.

The easiest thing to do is to upload an Excel file manually and create the database structure from the upload. This is fine if your data source is an Excel spreadsheet. If you are working with other formats, convert them into a spreadsheet first. This is a suggestion but saves a lot of time and precision issues.

Main Menu > Metadata > Data Management > Create Database from Excel Data

You can view the loaded resulting data next to see how when the data has been cleaned up, it is represented in the system.

Main Menu > Metadata > Data Management > Data - Export

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

7 Retrieve Management

The retrieval process is how bxp sets about getting files or data into the system so that the input process can load it into our database.

Most commonly people use the Retrieve – Manual Upload function to quickly put a file onto the server for the Input rules to process.

Main Menu > Metadata > Retrieve Management > Retrieve – Manual Upload

However there are a host of various sources from which bxp can retrieve files. Files and data can be PUSHED or PULLED into bxp.

Pushed = A user or system pushes the file or data into the secure file storage back end of bxp. These types of transfer do not require a retrieve rule as an external party will get the file into bxp.

Source: Manual Upload
Source: SFTP file transfer into bxp

Pulled = bxp is requested to go to a source and retrieve files or data there and bring them back into the secure file storage back end of bxp. These sources require bxp to have a Retrieve rule so that it knows what to perform.

Source: SFTP file transfer pulling from a server remote from bxp
Source: Email Attachment

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

8 Input Rules

With the files now available to bxp, it becomes necessary to get the data from the file and into the storage structure. There are a number of file types support which are documented here.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

8.1 Function: Read_Delimited

Delimited files are very common. A CSV file sees values separated by commas. However a CSV file is often a covering title for a character delimited / separated file. Often the tab character or hash character (#) or semicolon (;) are used to delimit fields.

Options	Description
Destination Database	The database into which the file will attempt to be loaded.
Filename Pattern	If there are multiple files in the back end, it is important to help differentiate the files using patterns in their file names. *.csv would be all files that end with .csv for example
Delete When Done	Should the file be deleted when processed to prevent reprocessing of the same file
Force Field Count Match	If the field counts do not match, should the record be inserted. If there are too many fields in the data, the first matching records are added. If there are a too few fields, the matches are made to the first available fields. If this option is true, non-matching records will not be added at all.
Field Mapping	A custom order of the fields. Otherwise default blind matching will be used. Blinding matching means that first item of data is placed in the first field. Second item of data goes into the second field and so on.
Delimiting Character	What character delimits the fields

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

8.2 Function: Read_Excel

Reading from an Excel spreadsheet is supported for all types of Excel spreadsheet. There are a number of options also available for Excel sheets.

Options	Description
Destination Database	The database into which the file will attempt to be loaded.
Filename Pattern	If there are multiple files in the back end, it is important to help differentiate the files using patterns in their file names. *.xls would be all files that end with .xls for example
Delete When Done	Should the file be deleted when processed to prevent reprocessing of the same file
Force Field Count Match	If the field counts do not match, should the file be inserted. If there are too many fields in the data, the first matching records are added. If there are a too few fields, the matches are made to the first available fields. If this option is true, non-matching files will not be added at all.
Field Mapping	Should the system expect a header row in the data and should it attempt to perform Field Mapping using the BE field mapping engine

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

8.3 Function: Read_Structured

Reading structured data is equally as easy but we must provide the spacing of the data.

Options	Description
Destination Database	The database into which the file will attempt to be loaded.
Filename Pattern	If there are multiple files in the back end, it is important to help differentiate the files using patterns in their file names. .dat would be all files that have .dat somewhere in the file name.
Delete When Done	Should the file be deleted when processed to prevent reprocessing of the same file
Force Field Count Match	If the field counts do not match, should the record be inserted. If there are too many fields in the data, the first matching records are added. If there are a too few fields, the matches are made to the first available fields. If this option is true, non-matching records will not be added at all. The field count is matched on the total width of the data line. If too many / too few characters the line is considered not to match. The positional data will be taken regardless of length and spaces appended where fields are too short.
Field Mapping	A custom order of the fields. Otherwise default blind matching will be used. Blinding matching means that first item of data is placed in the first field. Second item of data goes into the second field and so on.
Field Sizes	The exact widths of fields to retrieve. Combined with field mapping the fields are loaded.

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9 Processing Rules

The MetaData module of bxp contains a number of processing rules that allows the MetaData program creator to manipulate the data of a database to meet any final output requirement. The Processing rules are broken down into the following 8 Categories.

1) String Management functions
2) Mathematical functions
3) Date & Time management
4) Data Management functions
5) Data Conversion functions
6) Cross Database functions
7) CCL specific functions
8) QA Specific functions

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.1 String Management functions

9.2 Append

The append rule allows for any string to be appended to the end of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to append the new string to
5) New string to append

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]]String to append[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Append[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]Add This String To The End[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.3 Prepend

The prepend rule allows for any string to be appended to the start of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to prepend the new string to
5) New string to append

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]]String to prepend[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Prepend[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]Add This String To The Start[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.4 Encapsulate

The encapsulate rule allows for any string to be to be added to and surround the contents of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to encapsulate the new string with
5) New string to append

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]]String to encapsulate[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Encapsulate[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]Surround With This[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.5 Remove Numbers

The remove numbers rule simply removes all numbers from the contents of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to remove number from
5) New string to append

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]] [[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

RemoveNumbers[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.6 Strip Tags

The strip tags rule simply removes all HTML tags (
, <b>, <table>, </table> etc.) from the contents of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to remove HTML tags from
5) New string to append

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

StripTags[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.7 Replace

The replace rule simply allows for the replacement of a string with another from the contents of an existing string stored in a Database field.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field that contains the string to attempt the replacement on
5) Value to search for
6) Value to replace with

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Filed containing live value[[-SEP-]]Field to attempt replacement on[[-SEP-]]Value to search for[[-SEP-]]Value to Replace with[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Replace[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]strCDA_1_field_0_3[[-SEP-]]Find This[[-SEP-]]Replace With This[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.8 Concatenation

The concatenation rule simply allows for the combining of the contents of one database field with another within the same record of the same database

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) First field to be concatenated
6) Form id (the same as the current sequence)
7) Spacer (string to separate the two fields)
8) Form id (the same as the current sequence)
9) Second filed to be concatenated

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination Field[[-SEP-]]Form Id[[-SEP-]]First Field[[-SEP-]][[-SEP-]]Form Id[[-SEP-]][[-SEP-]]Spacer[[-SEP-]]Form Id[[-SEP-]]Second Field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Concat[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]]1[[-SEP-]][[-SEP-]] - [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.9 Read From

The read from rule simply allows for the for the searching of a string for a search sting and if found remove every character before and including the search string

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Field to perform the searching on
6) Value to search with

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form ID[[-SEP-]]Field To Search [[-SEP-]]Value To Search For[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

ReadFrom[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_428_field_0_2[[-SEP-]]Find This[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.10 Read Until

The read until rule simply allows for the for the searching of a string for a search sting and if found remove every character after and including the search string

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Field to perform the searching on
6) Value to search with

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form ID[[-SEP-]]Field To Search [[-SEP-]]Value To Search For[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

ReadUntil[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_428_field_0_2[[-SEP-]]Find This[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

9.11 Set Value

The set value rule simply allows for a database field to be updated with a new string, defined in the MetaData Program.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Value to insert

Example:

Rule Name[[-SEP-]]Form ID[[-SEP-]]Destination Field[[-SEP-]]Form Id[[-SEP-]][[-SEP-]]Value To Insert[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

SetValue[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]][[-SEP-]]Add This Value[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10 Mathematical functions

10.1 Addition

The addition rule allows for any two numbers stored in database fields to be added together.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field Containing the first number
5) Field Containing the second number

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]First Number field[[-SEP-]] Second Number field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Add[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]] strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.2 Subtraction

The subtraction rule allows for any two numbers stored in database fields to be subtracted.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field Containing the first number
5) Field Containing the second number

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]First Number field[[-SEP-]] Second Number field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Subtract[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.3 Multiplication

The multiplication rule allows for any two numbers stored in database fields to be multiplied together.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field Containing the first number
5) Field Containing the second number

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]First Number field[[-SEP-]] Second Number field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Multiply[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.4 Division

The division rule allows for any two numbers stored in database fields to be divided by each other.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field Containing the first number
5) Field Containing the second number

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]First Number field[[-SEP-]] Second Number field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Divide[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]] strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.5 Modulus

The Modulus rule allows for any two numbers stored in database fields to be mod'ed together to get the remainder.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field Containing the first number
5) Field Containing the second number

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]First Number field[[-SEP-]] Second Number field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Modulus[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]strCDA_1_field_0_3[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.6 Numeric

The numeric rule allows for any of the database fields to be converted into a number i.e. Make sure the contents are a number. If the contents are not recognised the field is defaulted to 0.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Field Containing the value to be converted

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]] Field to be converted[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Numeric[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]] strCDA_1_field_0_1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.7 Summation

The summation rule allows for any of the database field to have their contents summed together for the entire group of records returned by the Sequence, this summation value is then stored in all of the records that were included in the summation.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Field to summate

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]Field to summate[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Sum[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.8 Count

The count rule allows for the counting of the number of records returned by the sequence. The count value is then store in every record returned by the sequence.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Count[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.9 Is Unique

This function can be used to see if this records value is unique or exists elsewhere.

Is Unique will return the string "True" or "False" into the source data field

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form Id to Search in
5) Fields to search on
6) Filed containing the search value

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Search Form Id[[-SEP-]]Field to search on[[-SEP-]]field value to search with[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

IsUnique[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]2[[-SEP-]]strCDA_2_field_0_1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

10.10 Group Count

This function is used to group values, count them up and store the count in a field of those "grouped" records.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Filed to group by

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]field to group on[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

GroupCount[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

10.11 Group Sum

This function is used to group values, total them up (despite them being stored as strings) and store the count in a field of those "grouped" records. Sum will also compensate for text data being in the fields as well as null and empty values.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Filed to group by
5) Field containing the values to summate

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]field to group on[[-SEP-]]Field containing the values to summate[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

GroupCount[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11 Date & Time management

11.1 Date

The date rule allows for a field in the database to be formatted into universal date format i.e. YYY-MM-DD

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Filed containing the date to format

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]
Date to format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Date[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.2 Date|Time

The date|time rule allows for a field in the database to be formatted into universal date time format i.e. YYY-MM-DD HH:MM:SS

Required fields to execute rule:

6) Rule Name
1) Form id (the same as the current sequence)
2) Field to store the end result
3) Form id (the same as the current sequence)
4) Filed containing the date to format

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]
Date to format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DateTime [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.3 Date|Time Now

The date|time now rule allows for a field in the database to be updated with the current date and time in universal date time format i.e. YYY-MM-DD HH:MM:SS

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DateTimeNow [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.4 Report Time To Seconds

The report time to seconds rule converts a field in the database that contains a report time to a seconds value.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field containing the value to convert

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Field containing the value to convert[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

ReportTimeToSeconds[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]] strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.5 Seconds To Report Time

The seconds to report time rule converts a field in the database that contains seconds to a report time.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Field containing the value to convert

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Field containing the value to convert[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

SecondsToReportTime[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]] strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.6 Time

The time rule allows for a field in the database to be formatted into universal time format i.e. HH:MM:SS

Required fields to execute rule:

1) Rule Name
5) Form id (the same as the current sequence)
6) Field to store the end result
7) Form id (the same as the current sequence)
8) Filed containing the date to format

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]
Date to format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DateTime [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.7 Time To Seconds

The time to seconds rule allows for a field in the database to be converted from a time in the form HH:MM:SS to seconds

Required fields to execute rule:

2) Rule Name
9) Form id (the same as the current sequence)
10) Field to store the end result
11) Form id (the same as the current sequence)
12) Filed containing the date to format

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]
Date to format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

TimeToSeconds [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.8 ZULU

The ZULU rule allows for a field in the database to be formatted into ZULU Date TIme format i.e. ZYYYY-MM-DDTHH:MM:SS

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Filed containing the date to format

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]Date to format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Date[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.9 Day Between

The Days Between rule calculates the difference between a passed in date and now, it will return the number of days.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Filed containing the date check

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]]Date to check[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DaysBetweenGivenAndNow [[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

11.10 Month To number

The month to number rule converts the text name of a month to the its numeric value.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field to store the end result
4) Form id (the same as the current sequence)
5) Filed containing the month to convert

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination field[[-SEP-]]Form Id[[-SEP-]] Month to conver[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

ConvertMonthTostrNumber[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

12 Data Management functions

12.1 Deletion

The Deletion rule allows for records to be deleted from any form

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) intCDA_X_Id

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]intCDA_X_Id[[-SEP-]][[-SEP [[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Delete[[-SEP-]]1[[-SEP-]]intCDA_1_Id[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

12.2 Remove Duplicate Records

The remove duplicate records function allows for the deletion of all duplicate records in a form.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field that contains the values to group on

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]intCDA_X_Id[[-SEP-]][[-SEP]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DeDupe[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

12.3 Flag Duplicate Records

The flag duplicate records function allows for the duplicate records in a form to be flagged as duplicates

Required fields to execute rule:

1) Rule Name
2) Form 1 Id
3) Form 1 fields
4) Form 2 Id
5) Form 2 fields
6) Outcome for CCL
7) Add CCL
8) Fields to update
9) Values to insert
10) CCL Coments
11) Comments additional
12) Current Record Id

Example:

Rule Name[[-SEP-]]Form 1 Id[[-SEP-]]Form 1 fields[[-SEP-]]Form 2 ID[[-SEP]]Form 2 fields[[-SEP-]]Outcome CCL[[-SEP-]]Add CCL[[-SEP-]]Fields to update[[-SEP-]]values to update[[-SEP-]]CCL Coments[[-SEP-]]Additional comments[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

DeDupe_Flag[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_8,strCDA_1_field_0_7[[-SEP-]]1[[-SEP-]]strCDA_2_field_0_79,strCDA_2_field_0_80[[-SEP-]]Flagg CCL Outcome[[-SEP-]]true[[-SEP-]]strCDA_1_field_0_23[[-SEP-]]True[[-SEP-]]MetaData Program - Flag[[-SEP-]]true[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

13 Data Conversion functions

13.1 Convert Data field

There may be CampaignList or CampaignSearch type fields with the databases where cross campaign field translation would be required to link a parent record with a child record.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Destination field
4) Form Id contains the search values
5) Fields that contain the search values
6) Form Id to search in
7) Fields to search in
8) Form Id to get return values
9) Return value fields

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Destination Fields[[-SEP-]]Form Id values[[-SEP]]Fields containing values[[-SEP-]]Form Id To Search[[-SEP-]]Fields containing values[[-SEP-]]Form Id return values[[-SEP-]]Fields with Return values[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

13.2 Convert Form Id

The convert form Id rule allows for the conversion of a form Id to be converted to any piece of information in the campaign table

Required fields to execute rule:

1) Rule Name
2) Field in the campaign table to return
3) Destination field
4) Field with the search value
5) Form id (the same as the current sequence)

Example:

Convert_FormId_FormField[[-SEP-]]Field to return[[-SEP-]]Destination field[[-SEP-]]Search value[[-SEP-]]form id[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Convert_FormId_FormField[[-SEP-]]strCampaign_Name[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

13.3 Convert Client Id

The convert client Id rule allows for the conversion of a clients Id to be converted to any piece of information in the client table

Required fields to execute rule:

1) Rule Name
2) Field in the campaign table to return
3) Destination field
4) Field with the search value
5) Form id (the same as the current sequence)

Example:

Convert_FormId_FormField[[-SEP-]]Field to return[[-SEP-]]Destination field[[-SEP-]]Search value[[-SEP-]]form id[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

Convert_ClientId_ClientField[[-SEP-]]strClient_CompanyStaffIdNumber[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]strCDA_1_field_0_2[[-SEP-]]1[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

14 Cross Database functions

14.1 Copy To Destination

This function allows you to specify a source value and then to sequentially add that value to a subsequent or previous number of fields.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Field containing the data to copy
4) Destination field
5) Number of times to copy the data
6) Gap between each insert

Example:

Rule Name[[-SEP-]]Form Id[[-SEP-]]Fields to copy[[-SEP-]]Destination fields[[-SEP]]number of times to copy the data[[-SEP-]]Gap between each copy[[-SEP-]] [[-SEP-]] [[-SEP-]] [[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

14.2 Cross Database Update

This function allows for the updating of records in one database with values stored in another

The Options field should be set to True if a CCL is required to be inserted as well in the destination campaign.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Destination field
4) Form id (the same as the current sequence)
5) Fields separated by comma's
6) Form id (the same as the current sequence)
7) Fields to separate by comma's

Example:

Rule Name[[-SEP-]] Form id [[-SEP-]]Destination field[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Options[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

CrossDatabase_Update[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]2[[-SEP-]] strCDA_2_field_0_1 [[-SEP-]]3[[-SEP-]]strCDA_3_field_0_1[[-SEP-]]True[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

14.3 Cross Database Insert

This function allows for the creation of new records in a database using the values stored in another database. The system will auto generate the values for the system fields including the ID, Future Use, Status, Last Date Time, Last Agent Id, unless explicitly stated.

The Options field should be set to True if a CCL is required to be inserted as well in the destination campaign.

Required fields to execute rule:

1) Rule Name
2) Form id (the same as the current sequence)
3) Destination field
4) Form id (the same as the current sequence)
5) Fields separated by comma's
6) Form id (the same as the current sequence)
7) Fields to separate by comma's

Example:

Rule Name[[-SEP-]]Form id[[-SEP-]]Destination field[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Options[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

CrossDatabase_Insert[[-SEP-]]1[[-SEP-]]strCDA_1_field_0_1[[-SEP-]]2[[-SEP-]]strCDA_2_field_0_1 [[-SEP-]]3[[-SEP-]]strCDA_3_field_0_1[[-SEP-]]True[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

14.4 Cross Database Update Simple/Custom

This function allows for the updating of records in one database with values stored in another, using a custom group by clause

Required fields to execute rule:

1) Rule Name
2) Table name (values to be updated in)
3) Fields to update
4) Files that contain the values to update
5) Form Id to search on
6) Update search field
7) Form id containing search value
8) Update search value

Example:

Rule Name[[-SEP-]] Table name [[-SEP-]]Fields to update[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Form id[[-SEP-]]Fields[[-SEP-]]Options[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

CrossDatabase_Update_Custom[[-SEP-]]CDA_1[[-SEP-]]strCDA_1_field_0_7[[-SEP-]]strCDA_2_field_0_1[[-SEP-]]intCDA_2_Id[[-SEP-]]strCDA_1_field_0_7[[-SEP-]]CDA_2[[-SEP-]]strCDA_1_field_0_7[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

14.5 Cross Database Update Simple/Custom

This function allows for the updating of records in one database with values stored in another, using a custom group by clause. If a matching record is not found in the destination database then the MetaData will not perform and update. It does however perform a brand new record insert

Required fields to execute rule:

1) Rule to process
2) Table to insert to
3) Fields to insert values into
4) Values to insert into the fields
5) Table with the values
6) Fields to search for records
7) values to search for
8) Boolean add CCL
9) CCL Outcome
10) CCL Comments
11) Boolean update current status
12) New Status
13) Part Matching Enabled
14) Alternate matching patterns fields
15) Alternate matching patterns value

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Example:

CrossDatabase_Update_Smart[[-SEP-]]11[[-SEP-]]strCDA_11_field_0_4,strCDA_11_field_0_7,strCDA_11_field_0_8,strCDA_11_field_0_9,strCDA_11_field_0_10[[-SEP-]]strCDA_12_field_0_4,strCDA_12_field_0_7,strCDA_12_field_0_8,strCDA_12_field_0_9,strCDA_12_field_0_10[[-SEP-]]12[[-SEP-]]strCDA_11_field_0_16{{--SEP--}}strCDA_11_field_0_17{{--SEP--}}strCDA_11_field_0_18{{--SEP--}}strCDA_11_field_0_4[[-SEP-]]--AccountFirstname--{{--SEP--}}--AccountSurname--{{--SEP--}}--AccountDOB--{{--SEP--}}--AccountIOB--[[-SEP-]]true[[-SEP-]]Real Time update from IOB [[-SEP-]]Real Time update process from DB Id:12[[-SEP-]]true[[-SEP-]]Real Time update from DB 11[[-SEP-]]true[[-SEP-]]strCDA_11_field_0_16{{--SEP--}}strCDA_11_field_0_17{{--SEP--}}strCDA_11_field_0_18{{##SEP##}}strCDA_11_field_0_16{{--SEP--}}strCDA_11_field_0_17{{--SEP--}}strCDA_11_field_0_4{{##SEP##}}strCDA_11_field_0_16{{--SEP--}}strCDA_11_field_0_18{{--SEP--}}strCDA_11_field_0_4{{##SEP##}}strCDA_11_field_0_17{{--SEP--}}strCDA_11_field_0_18{{--SEP--}}strCDA_11_field_0_4[[-SEP-]]--AccountFirstname--{{--SEP--}}--AccountSurname--{{--SEP--}}--AccountDOB--{{##SEP##}}--AccountFirstname--{{--SEP--}}--AccountSurname--{{--SEP--}}--AccountIOB--{{##SEP##}}--AccountFirstname--{{--SEP--}}--AccountDOB--{{--SEP--}}--AccountIOB--{{##SEP##}}--AccountSurname--{{--SEP--}}--AccountDOB--{{--SEP--}}--AccountIOB--[[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

15 CCL Specific functions

15.1 Insert Into CCL Simple

This function allows you to get information from the CCL table of a form and insert the data returned from the CCL table into a CDA table

Required fields to execute rule:

1) Rule Name
2) CCL Table (the same as the current sequence)
3) Fields to retrieve from the CCL table
4) CDA table to insert data to
5) Fields to insert the data to

Example:

Rule name[[-SEP-]]CCL table[[-SEP-]]Fields to retrieve[[-SEP-]]CDA table[[-SEP-]]Fields to insert[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

InsertFromCCL_Custom[[-SEP-]]CCL_1[[-SEP-]]intCCL_1_CDAId,dteCCL_1_FirstCCLDateTime[[SEP-]]CDA_2[[-SEP-]]strCDA_2_field_0_2,strCDA_2_field_0_8[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

15.2 Insert From CCL Custom

This function allows you to get information from the CCL table of a form and insert the data returned from the CCL table into a CDA table

Required fields to execute rule:

1) Rule Name
2) CCL Table (the same as the current sequence)
3) Fields to retrieve from the CCL table
4) CDA table to insert data to
5) Fields to insert the data to

Example:

Rule name[[-SEP-]]CCL table[[-SEP-]]Fields to retrieve[[-SEP-]]CDA table[[-SEP-]]Fields to insert[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

InsertFromCCL_Custom[[-SEP-]]CCL_1[[-SEP-]]intCCL_1_CDAId,dteCCL_1_FirstCCLDateTime[[SEP-]]CDA_2[[-SEP-]]strCDA_2_field_0_2,strCDA_2_field_0_8[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd

16 QA Specific functions

16.1 Cross Database QA Fail All Check

This function is for the checking of a QA record to see if an automatic fail question is flagged. And returning either a 0/1 or a True/False

Required fields to execute rule:

1) Rule Name
2) Form Id To Search
3) Form to store result
4) Field that contains the field to search in
5) Field that contains the value to search with
6) Filed that is to be updated with the result
7) Return output option

Example:

Rule name[[-SEP-]]Search form[[-SEP-]]Result form[[-SEP-]]Field to search in [[-SEP-]]Filed with search value[[-SEP-]]field to update[[-SEP-]]return format[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

Example (with replacement values):

CrossDB_QA_Fail_All_Process[[-SEP-]]1[[-SEP-]]2[[-SEP-]]intCDA_1_Id[[-SEP-]]strCDA_2_field_0_6[[-SEP-]]strCDA_2_field_0_32[[-SEP-]]boolean[[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]][[-SEP-]]

48 / 49 Western Parkway Business Park, Lower Ballymount Road, Dublin 12, Ireland
t: +353 1 429 4000 (Dublin) +44 207 692 0705 (London) e: info@bxpsoftware.com w: www.bxpsoftware.com

Directors: Nick Wheeler • Chris Thomson • Philip Lacey
VAT 642 070 G • Company Registration No. 400703 • bxp software is owned and distributed by All n One Ltd