Auto-Numbering Manual

Written by Ahmed el-Sawalhy

V3.1

Contents

Features

- String, date, parameter, and attribute patterns
- Run numbering on a condition
- Random strings, with number-to-letter ratios, and "start with a letter" flag
- Optional numbering sequence, with padding
- Reset interval -- periodic or one-time reset -- for numbering sequence
- Locking when busy to avoid duplicate indexes
- Option to use plugin instead of workflow step, which allows the generation of numbering for entities that lock after the operation
- Option to validate unique generated string
- Option to generate without updating a record (return the generated string only)
- Support for plugin step inline configuration
- Use a backlog to avoid long DB locks
    - The solution reserves an index, and if a rollback happens, the index is saved for future use by another run
    - This might cause out-of-order indices
- Create different index sequence per field value

Trigger

Workflow

Use the custom step provided in the auto-numbering assembly inside a workflow.

If you choose a field name in the auto-numbering configuration, the step will update the record automatically; otherwise, use the string returned in the output parameter of the step to update the record manually.

Optionally, set the input parameters (semicolon separated). Those values will replace "`{>param-#}`" in the format string.

The step returns the resulting string, the index as an integer and as a padded string (streams are not returned).

Plugin

Configuration-based

The plugin can run on the creation and update of a record. Register a plugin step on pre-validation, and on pre or post-operation stages.

For pre-validation, register a step on the types 'PreCreateTargetAutoNum' (pre-operation) **and** 'PrevalCreateTargetUseBacklog' (pre-validation). A special feature called 'backlog' will be triggered, which I will be explained later.

For pre-operation, register a step on the type "PreCreateTargetAutoNum" (Create) or "PreUpdateTargetAutoNum" (Update). For post-operation, register a step on the type "PostCreateTargetAutoNum" or "PostUpdateTargetAutoNum."

For steps registered on the Update event, and for ones registered on post-operation stages in general, add a post-image that includes the fields specified in "`{[logicalname]}`" patterns in the format string in the configuration.

In all the above steps, add the unique ID generated in the config record to the unsecure config of the step.

Inline Configuration

An alternative to using config records is to add the config directly to the unsecure config of the plugin step

Must be in the following format: "`[format string];;[target field name]`", and optionally "`;;true`" at the end for unique validation.

Only Field Value, Random String, and Date formats are supported; e.g. "`Test-{name}-{!un-5}-{@yyyy};;ldv_name`".

This is very useful for non-indexed auto-numbering, especially when using a combination of field values from the record to form the generated string.

Placeholders

The following are variables that can be inserted between '`{`' and '`}`' to be replaced by their function upon record creation:

Index Value

The auto-numbering index is inserted in-place of "{>index}". This is the default indexing, saved directly in the configuration itself.

Streams

If you want a different index sequence for different field values, specify the field name in the format string directly: "{>index-[field-name]}" ({>index-casetypecode}). This will result in all values having different sequences; when the value is set in the record, the auto-numbering will use the appropriate "stream" or sequence.

Additionally, you can choose to use a specific stream: "{>index-[field-name]:[field-value]}" ({>index-casetypecode:1}).

Field Value

Add "{[field-name]}" to insert a value from the triggering record. The value is raw, i.e. the value is not formatted.

Formatting

To format the value, add an '@' after the field name and then the format; e.g. "{casetypecode@1025}", which is replaced by the Arabic label of the Case Type field value.

- "@[lang-id]": replaces an option-set value with the localised label
- "@[date-format]": replaces a date value with its custom formatted equivalent (e.g. @dd-MM-yyyy)
- "@": replaces the lookup GUID with its name, which is retrieved from the primary name field of the entity
- "@[precision]": replaces a decimal value with another at the specified precision (e.g. @2)

Depth traversal is supported as well by using the .-operator on lookups; e.g. "{primarycontactid.gendercode}".

Conditions

If you require even more control over this placeholder, you can specify nested conditions. A basic condition: "{{[field-name]}??[filled-value]::[empty-value]}"; e.g. "{{address}??FILLED::EMPTY}", which translates to "if address is filled, print 'FILLED', else print 'EMPTY'".

A more advanced condition: "{{casetypecode}==1??NUM-{customerid.accountnumber}::EMAIL-{customerid.emailaddress1}}" (result: NUM-123 or EMAIL-test@test.com).

Nested conditions are supported as well.

Supported operators: ==, >=, <=, and !=.

Random Strings

If you require a sequence of random letters to be generated, use "{![u|l|n]-[length]}"; e.g. "{!un-5}" might result in "YUD52".

- "u": takes from the uppercase pool of letters

- “l”: takes from the lowercase pool of letters
- “n”: takes from the 0-9 pool of numbers

You can specify the ratio of numbers to letters in the auto-numbering configuration record.

Alternatively, specify the pool to pick from; e.g. "{!A,B,C,a,b,c-4}".

Date

Dates can be custom formatted and inserted into the generated string.

- Must start with an '@'
- The time zone of the owner of auto-numbering record is used to determine the output string
- Followed by a day, month, or year in the standard format
- E.g. "{@dd}" or "{@mm/yy}"

WF Parameters

When using the custom step to generate an auto-number, you can pass some parameters into the step.

In the properties of the custom step, there is a field for the parameters. Insert the values you would like to pass to the step in order, and separate them with a semi-colon (;).

In the placeholder, specify the index of the parameter to use, starting with 1; e.g. “{>param-3}”.

Notes

Any spaces in the generated string are replaced with an '_'.

Date Formats

- Year: "y yy yyy yyyy" => "8 08 008 2008"
- Month: "M MM MMM MMMM" => "3 03 Mar March"
- Day: "d dd ddd dddd" => "9 09 Sun Sunday"
- Hour 12/24: "h hh H HH" => "4 04 16 16"

Reserved Characters

- '{' and '}'
- Leading '!', '@', '?', and '$'
- '-' and ',' in the string list after an '!'
- Equality operators, ‘??’, and ‘::’

Examples

- "Test-{{createdon}??{createdon@hh:mm}::NO_DATE}-{!un-5}-{@yyyy}-{>index}-{>param-3}"
    - With current index 5, padding 3, the user in cairo (9AM) and server in London (7AM), and 'PA;RA;M' as input parameter
    - Results in "Test-09:00- YAM76-2015-005-M"