Games ‘FileDirectorRetentionParsing’

Making File Directors Retention processes central to retention processing

# Requirements:

Generate a list of references from File Director that contain certain retention criteria and send to an API endpoint.

# Solution:

Filedirector’s retention ability has limited filtering and doesn’t allow to trigger external hooks. Without the process/workflow module being licensed it is very limited and being that there are limited FileDirector partners who utilise the workflows scripting processes, it was considered at the time that if there were any future support requirements away from C2S it could be handled by developers without a filedirector background. So, creating a simple project in C# that parses the information from the retention profiles in the database and sends the field references to the external API endpoint created by IdComputersoftware was an easier prospect from support and allows for further options in the filtering as these can be interpreted. This tool can then run from a task scheduler at the increments required and not specific to FileDirectors own scheduler which runs against all profiles on all cabinets at the same time.

Pros

* It allows us to work away from the obstructions that File Director imposes
* It can run independent to the main FileDirector retention process schedule
* We can add in extra logic around what it runs against
* The process only gathers the detail it needs from the database and nothing more, so it is quick to process.

Cons

* Not internal to FileDirector, so brings a slight obfuscation on its use and location.
* If the Filedirector changes its retention processing, it could cause the retention process system to no longer be able to interpret it.

# Setting up a retention process in FileDirector

You would complete as normal, but you would choose for nothing to happen for the destruction options. FileDirector should ignore these, which allows our own process to run against it.

# Making the FileDirectorRetentionParsing process aware of it

Within the applications config file, you would enter a new String to the array in the RetentionIdList. This is the retention Id

Graphical user interface, text, application

Description automatically generated

# Setting up a retention process New Options for filtering

Setting the Filed values for the fields has three new options to add additional logic into the profiles by including in the field filter. The current logic for FileDirector retention only is an equal value ie. FieldA = “Test”. Where there are instances that you may require other logic around these field values.

If field Is blank {c2sretention\_isblank}

Example: {c2sretention\_isblank}

If field is not blank {c2sretention\_notblank}

Example: {c2sretention\_notblank}

And ‘Not’ condition {c2sretention\_not}

Example: {c2sretention\_not}Test