

Ticket 23: Support for RSP taggin

Group 3 - Jonathan Velasco

May 25, 2008

Abstract

Matthew said RSP uses their naming convention for files, internally. This is a general ticket on how files should be tagged. A couple of suggestions is to use key-values pairs as metadata (minutes 20 March 2008)

1 Tasks

In order to implement this ticket, the task to be done are:

- Study and understand of the current code 3 hours
- Elaboration of design 2 hours
- Implementation and Coding 1.5 to 4.5 hours
 - Modifying the database 1 hours +/- 0.5 hour
 - Allow user to search through tags 1 hour +/- 0.5 hour
 - Populate the table 1 hours +/- 0.5 hour
- Tests
 - Run existing tests and check coverage
 - Test the modifications which are not cover on the existing test. 0 hour to 2 hour depending of existing tests

This ticket will be ideally be completed between 6.5 and 9.5 hours

2 Design

In order to address this ticket, some extra columns had been added to the 'file' table in the database. See figure 1

The column added are 'job', 'sequence' and 'shot'. Each file in RSP will be under a specific directory structure, therefore, the information related to the 'job', 'sequence' and 'shot' is extracted from the path of such file.

files
+id
+name
+bytes
+uid
+gid
+directory_id
+blocks
+sequence
+job
+shot

Figure 1: New 'file' table

```

1 # Add a new columns 'sequence', 'job' and 'shot' to table files
2 # for tag support for RSP name convention
3 class AddRspTagToFiles < ActiveRecord::Migration
4   def self.up
5     add_column :files, :sequence, :string
6     add_column :files, :job      , :string
7     add_column :files, :shot    , :string
8   end
9
10  def self.down
11    remove_column :files, :sequence
12    remove_column :files, :job
13    remove_column :files, :shot
14  end
15 end

```

Figure 2: File to add the extra columns into the 'file' table

3 Modifications Done

- In order to include the extra columns 'job', 'sequence' and 'shot' in the 'file' table, the file *earth/db/migrate/060_add_rsp_tag_to_files.rb* had been created. See figure 2. This task took aprox 30 minutes
- In order to allow the user to search through 'job', 'sequence' and/or 'shot', the file *earth/app/views/browser/_breadcrumb_and_filter.html.haml* was modified as showed in figure 3. This changes are reflected in figure 4. This took aprox 30 minutes
- In order to populate the new columns in the 'file' table, some changes in *earth/lib/earth_plugins/file_monitor.rb* where done as show in figure 5. The time expended in order to understand which file to change and how to do it was aprox 7 hours. After finding where the changes should be done, the time coding was aprox 10 minutes
- In order to execute the search, some lines were added to the file *earth/app/models/earth/file.rb* in order to return only the files matching the searching criteria of the user. See figure 6. This took aprox 20 minutes

```

30 .Right#Filter{:style => "margin: 1em 0 1em 0 0;"}
31 - form_tag({:overwrite_params => {}}, {:method => 'get'}) do
32   %table
33     %caption Filter
34     %tr
35       %td.first-row{:style => "text-align: right;"}= content_tag :label, "by filename:", "for" => "filter_filename"
36       %td.first-row= text_field_tag "filter_filename", params[:filter_filename], :size => 15
37     %tr
38       %td.second-row{:style => "text-align: right;"}= content_tag :label, "by user:", "for" => "filter_user"
39       %td.second-row= text_field_tag "filter_user", params[:filter_user], :size => 15
40       = auto_complete_stylesheet
41       = content_tag "div", "", :id => "filter_user_auto_complete", :class => "auto_complete"
42       = auto_complete_field "filter_user", :url => {:action => "auto_complete_for_filter_user"}
43     %tr
44       %td.third-row{:style => "text-align: right;"}= content_tag :label, "by job:", "for" => "filter_job"
45       %td.third-row= text_field_tag "filter_job", params[:filter_job], :size => 15
46     %tr
47       %td.fourth-row{:style => "text-align: right;"}= content_tag :label, "by sequence:", "for" => "filter_sequence"
48       %td.fourth-row= text_field_tag "filter_sequence", params[:filter_sequence], :size => 15
49     %tr
50       %td.last-row{:style => "text-align: right;"}= content_tag :label, "by shot:", "for" => "filter_shot"
51       %td.last-row= text_field_tag "filter_shot", params[:filter_shot], :size => 15
52     %tr

```

Figure 3: Modifications to allow the user to search by 'job', 'sequence' and or 'shot'

Figure 4: Incorporation of extra fields for searching by RSP directory structure convention

```

333 # Jon: Include 'job' and 'sequence' into the table 'files'
334 # the code belong use to be:
335 #   Earth::File.benchmark("Creating file with name #{name}", Logger::DEBUG, !log_all_sql) do
336 #     directory.files.create(:name => name, :stat => stats[name])
337 #   end
338 pathArray = "#{directory.path}".split('/')
339 seq = pathArray.last
340 job = pathArray[pathArray.length-2]
341 Earth::File.benchmark("Creating file with name #{name}", Logger::DEBUG, !log_all_sql) do
342   directory.files.create(:name => name, :job => job, :sequence => seq, :shot => name, :stat => stats[name])
343 end
344 # Jon : End include 'job' and 'sequence' into the table 'files'

```

Figure 5: Changes done to file_monitor.rb

```

79 # JON - support for job
80 filter_job = params[:filter_job]
81 filter_job = "" if filter_job.blank?
82 # JON - end
83
84 # JON - support for sequence
85 filter_sequence = params[:filter_sequence]
86 filter_sequence = "" if filter_sequence.blank?
87 # JON - end
88
89 # JON - support for shot
90 filter_shot = params[:filter_shot]
91 filter_shot = "" if filter_shot.blank?
92 # JON - end
93
94
101 # JON - search files using tags
102 # search files through job , sequence and shot
103 elsif filter_job != '*' && filter_sequence != '*' && filter_shot != '*'
104   ["files.job LIKE ? AND files.sequence LIKE ? AND files.shot LIKE ?", filter_job.tr('*', '%'), filter_sequence.tr('*', '%'), filter_shot.tr('*', '%')]
105
106 # search files through job and sequence
107 elsif filter_job != '*' && filter_sequence != '*'
108   ["files.job LIKE ? AND files.sequence LIKE ?", filter_job.tr('*', '%'), filter_sequence.tr('*', '%')]
109 # search files through shot and sequence
110 elsif filter_shot != '*' && filter_sequence != '*'
111   ["files.shot LIKE ? AND files.sequence LIKE ?", filter_shot.tr('*', '%'), filter_sequence.tr('*', '%')]
112 # search files through job and shot
113 elsif filter_job != '*' && filter_shot != '*'
114   ["files.job LIKE ? AND files.shot LIKE ?", filter_job.tr('*', '%'), filter_shot.tr('*', '%')]
115 # search files through job
116 elsif filter_job != '*'
117   ["files.job LIKE ?", filter_job.tr('*', '%')]
118 # search files through sequence
119 elsif filter_sequence != '*'
120   ["files.sequence LIKE ?", filter_sequence.tr('*', '%')]
121 # search files through shot
122 elsif filter_shot != '*'
123   ["files.shot LIKE ?", filter_shot.tr('*', '%')]
124 # JON - end
125 end
126 end

```

Figure 6: Changes made to the file model