



Collabora Productivity

# Adding support for grouping header entries in Calc Online

By Marco Cecchetti  
Collabora Productivity

[marco.cecchetti@collabora.co.uk](mailto:marco.cecchetti@collabora.co.uk)

# Calc Grouping

What they are, what they are used for

	A	B	E	F	G	H	I
2	CDL Project Plan without Lows2	500	87.3	1%			
3	Requirements and Development	500	77.3	1%			
4	Module 1	500	77.3	4%			
5	Eligible Transaction Sources	500	37.3	45%			
12	Stakeholder Discussions	500	31.3	44%			
18	Business Rules for Incoming Load	500	16.5	3%			
27	Definition of Intake Priorities (Met	500	3.3	20%			
30	File Load process map and gap	500	64.2	0%			
35	Product Data Source process map	500	19.0	0%			
39	Business Rules for Outgoing Data	500	20.6	0%			
42	Junk data analysis - approach opt	500	25.5	0%			
45	Document Intake Queue technical re	500	0.2	0%			
46	Process flow for Intake Queue	500	0.8	0%			
47	Document Intake Queue business re	500	0.3	0%			
48	Requirements Review and Post Rev	500	0.7	0%			
49	Finalize Source uptake Schedule	500	1.7	0%			
50	Development	500	77.3	0%			
70	Module 2	500	62.0	0%			
71	Decide on best record v/s best sc	500	2.1	0%			
77	Set thresholds - for all attributes	500	5.1	0%			
84	Document process flow for each p	500	4.0	0%			
88	Evaluate Compare Methods	500	2.3	0%			
94	General Requirements	500	24.6	0%			
102	Development	500	60.0	0%			
120	Manual Processing	500	69.6	0%			
121	Document business requirements	500	28.8	0%			
130	Document system requirements	500	28.7	0%			
138	Create business process flow cha	500	6.4	0%			
142	Development	500	59.3	0%			
161	Audit Tracking and Error Mgmt	500	67.6	0%			
162	Evaluate WAP processes and impac	500	0.3	0%			
163	Error management requirements	500	1.3	0%			
164	Understand current process and det	500	0.3	0%			
165	Requirements for Audit tracking	500	1.3	0%			

	1	2									
	1	2	3	4	A	B	E	F	G	H	I
	2	CDL Project Plan without Lows2			500	87.3	1%				
	3	Requirements and Development			500	77.3	1%				
	4	Module 1			500	77.3	4%				
	5	Eligible Transaction Sources			500	37.3	45%				
	12	Stakeholder Discussions			500	31.3	44%				
	18	Business Rules for Incoming Load			500	16.5	3%				
	27	Definition of Intake Priorities (Met			500	3.3	20%				
	30	File Load process map and gap			500	64.2	0%				
	35	Product Data Source process map			500	19.0	0%				
	39	Business Rules for Outgoing Data			500	20.6	0%				
	42	Junk data analysis - approach opt			500	25.5	0%				
	45	Document Intake Queue technical re			500	0.2	0%				
	46	Process flow for Intake Queue			500	0.8	0%				
	47	Document Intake Queue business re			500	0.3	0%				
	48	Requirements Review and Post Rev			500	0.7	0%				
	49	Finalize Source uptake Schedule			500	1.7	0%				
	50	Development			500	77.3	0%				
	70	Module 2			500	62.0	0%				
	71	Decide on best record v/s best sco			500	2.1	0%				
	77	Set thresholds - for all attributes			500	5.1	0%				
	84	Document process flow for each p			500	4.0	0%				
	88	Evaluate Compare Methods			500	2.3	0%				
	94	General Requirements			500	24.6	0%				
	102	Development			500	60.0	0%				
	120	Manual Processing			500	69.6	0%				
	121	Document business requirements			500	28.8	0%				
	130	Document system requirements			500	28.7	0%				
	138	Create business process flow cha			500	6.4	0%				
	142	Development			500	59.3	0%				
	161	Audit Tracking and Error Mgmt			500	67.6	0%				
	162	Evaluate WAP processes and impac			500	0.3	0%				
	163	Error management requirements			500	1.3	0%				
	164	Understand current process and det			500	0.3	0%				
	165	Requirements for Audit tracking			500	1.3	0%				

# What is needed to have them on online

- Rendering is already fine
- Client side:
  - implementing JS UI for group controls
  - message to be sent to server to update the group entry state .....



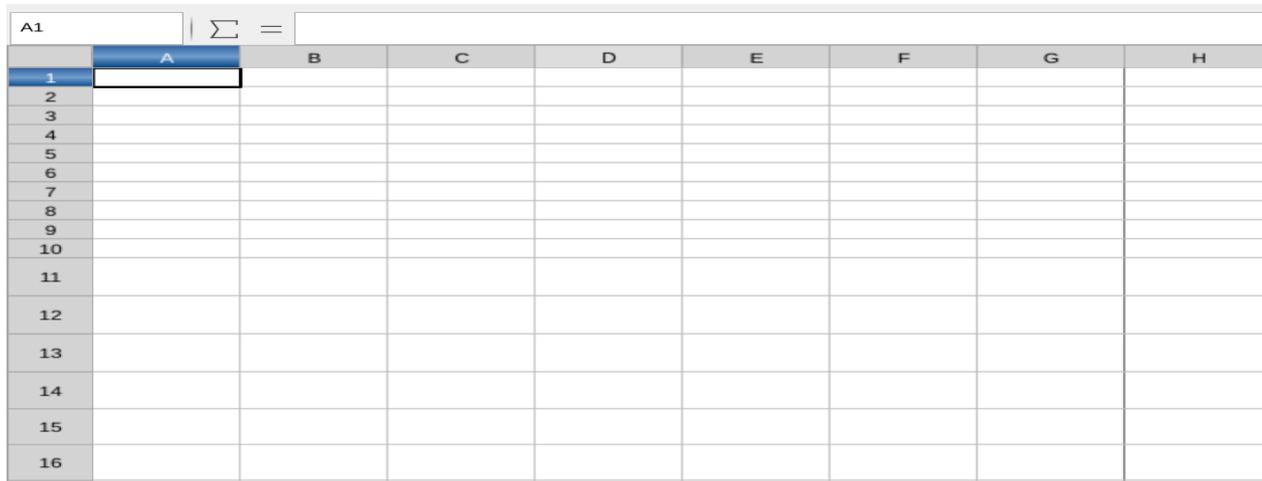
```
/**  
* Show/Hide a single row/column header outline for Calc documents.  
*  
* @param bColumn - if we are dealing with a column or row group  
* @param nLevel - the level to which the group belongs  
* @param nIndex - the group entry index  
* @param bHidden - the new group state (collapsed/expanded)  
*/
```

- Retrieving group data from core as we do for header data;

# Retrieving header data

- What is the header data ?
  - For each row entry:
    - position from the top of the document + entry index
- We send only row header data related to rows in the visible range
- We cache row entry position for the first and last row in the visible range

# Further header data size optimization



	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								

- We send only row entries data when a row height change occurs
- $\{\{255, 0\}, \{255, 1\}, \dots, \{255, 10\}, \{510, 11\}, \dots, \{510, 16\}\}$   
=>  $\{255, 0\}, \{510, 11\}, \{510, 16\}$
- It has required a `bit` of rework on the client side

# Changes on the core side:

- The outline state change when one of the following methods of ScOutlineDocFunc are called: ShowMarkedOutlines, HideMarkedOutlines, HideOutline
- OnLOKShowHideOutline takes care of:
  - performing cache position invalidation
  - invalidating headers
  - notify cursor overlay is changed
  - notify markdata is changed

**Using Canvas.2D for  
rendering headers  
and group outline on  
the client**

# Switching from <div> to <canvas> for rendering headers in client

- Earlier per each row header entry:

```
<div class="spreadsheet-header-row" style="height: 16px;">  
  <div class="spreadsheet-header-row-text" rel="spreadsheet-row-11"  
    style="line-height: 13px; height: 13px;">11</div>  
  <div class="spreadsheet-header-row-resize" style="height: 3px;"></div>  
</div>
```

- Needed because of group controls UI

# Handling mouse events

- Earlier event handlers was attached to the <div> element
- Keep track of current index
  - current index is used to insert/remove show/hide header entries
- Handle on mouse over, on mouse out for header entries
- Handle mouse over resize area
  - change cursor appereance
  - disable/enable click event

# The rendering of an header entry occurs on

- On selecting or unselecting a row entry
- On mouse over or out the header entry
- On scrolling => header is updated
- Rendering is performed through canvas.2D primitives by using the header data parsed by `ControlRowHeader.fillRows`

# Main change involved with header data subdivided in ranges

- Earlier `ControlRowHeader.fillRows` populated an array where the generic entry looked like `{pos, text-content, selected}`
- Now a `DataImpl` instance is populated

# ControlHeader.DataImpl

- A way for providing info on a header entry
- It create on the fly a single header data entry  
**{index, position, size}**

```
_computeEntry: function (startIndex, index) {  
    var entry = this._dataMap[startIndex];  
    var pos = entry.pos + (index - startIndex) * entry.size;  
    pos = this._twipsToPixels(pos);  
    var size = this._twipsToPixels(entry.size);  
    return {index: index, pos: pos, size: size};  
}
```



# Retrieving and rendering group data

- Group data is collected and sent to the client together with header data

```
{ "level": "1", "index": "0", "startPos": "975", "endPos":  
"2295", "hidden": "0" }
```

- On RowHeaderControl.fillRows:
  - if there are groups resize canvas
  - parse and collect group data
  - draw any group entry

# Further features

- Header corner has its own canvas element and its own event handling
- Menu for grouping (group and outline)



Collabora Productivity

# Thank you for listening!

**By Marco Cecchetti**

[marco.cecchetti@collabora.co.uk](mailto:marco.cecchetti@collabora.co.uk)