Calc Dependency & Performance Work

(mostly thanks to Kohei Yoshida)

Michael Meeks <michael.meeks@collabora.com>

mmeeks, #libreoffice-dev, irc.freenode.net

“Stand at the crossroads and look; ask for the ancient paths, ask where the good way is, and walk in it, and you will find rest for your souls...” - Jeremiah 6:16
Overview

- Calc – the ancient
- Calc – the modern: 4.3
- Calc – the latest: 4.4
  - Dependency rework
  - Chart performance
  - Export performance
Baseline: state of play 1x year ago.
Ancient (ScBaseCell)

ScDocument → ScTable → ScColumn

ScBaseCell

- Broadcaster (8 bytes)
- Text width (2 bytes)
- Cell type (1 byte)
- Script type (1 byte)

ScValueCell
ScStringCell
ScFormulaCell
ScEditCell
ScNoteCell*

Scattered pointer chasing walking cells down a column...
Modern (4.2/4.3) (mdds::multi_type_vector)
Ancient: pre 4.2

ScFormulaCell → ScTokenArray → Tokens

ScFormulaCell → ScTokenArray → RPN → Tokens

ScFormulaCell → ScTokenArray → ... → Tokens

ScFormulaCell → ScTokenArray → ... → RPN

ScFormulaCell → ScTokenArray → ... → ...
Modern: 4.2 / 4.3+

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScTokenArray

ScFormulaCellGroup

... Tokens

... RPN
Memory usage

Test document used:
Area data structure is somewhat complex, and packed with data, consider eg. 

=SUM(A1:A10) 
Filled down 100x … 

This produces a large number of intersecting / overlapping dependency ranges.
New things: Stuff you get in 4.4.
Dependencies 4.4

A single entry for the area slot machine; and a single listener.

No large SvtListener / Broadcaster linked lists – explicitly notifies cells it knows.

Time & Space savings.

Copying large formula groups: ~2x as fast over perfwork5 lifetime to 4.4
Other big wins:

- Script-Type optimizations
  - Asian, Complex, Latin, etc.
  - Determines fonts → sizing → row heights → we cache it.
    - Preserving this across copy/paste
    - Accelerated calculation down row-spans based on type.

- Chart Optimizations
  - Don't re-construct charts for every change to a range they depend on.
  - Wait until they become visible ....
FastParser: in 4.2 / 4.3 ...

Matus Kukan:

- 1 sheet with 100k numbers & 100k formulae
- Significant wins: threaded parsing
- “Fast” a mis-nomer:
  - uno::Sequence ...

4.6X vs. 4.1.3

seconds

Calc 4.1.3
Calc
Reference
Export in 4.4

- **FastSerializer**: *Matus Kukan*
  - Similar complete nonsense in the Serializer: individual sys-calls for “<” “table” “:” “table-cell” ... (ono) for “<table:table-cell ...”
  - Added: buffering, reduced allocation, improved stringification → finally ~fast.

- **Zip parallelism**: *Matus Kukan*
  - We build a lot of streams, then compress 1x by one – instead parallelize all stream compression (off by default in 4.4)
  - Avoid wasteful image re-compression: big ODP/PPTX saving. ~20% win for some presentations: a better way to get compression parallelism.

- **XclExpRow** – row / column style calculation: now threaded.
- deque → vector conversion removal ~20%
Questions ...

- Calc Performance
  - Continues to improve ...
  - Major code re-factors and representation improvements
- Plenty more to do ...
  - Poke me to get involved.
- Many thanks to all who support LibreOffice
  - We can't do it without you ...
  - Thanks to Kohei too for the great work here.

Oh, that my words were recorded, that they were written on a scroll, that they were inscribed with an iron tool on lead, or engraved in rock for ever! I know that my Redeemer lives, and that in the end he will stand upon the earth. And though this body has been destroyed yet in my flesh I will see God, I myself will see him, with my own eyes - I and not another. How my heart yearns within me. - Job 19: 23-27