LibreOffice Calc

Now available on your GPU

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

• A bit about:
  • GPUs ...
  • Spreadsheets
• Internal re-factoring
  • OpenCL optimisation
  • new calc features
  • XML / load performance
• Calc / GPU questions ?
• LibreOffice 4.2 : the FOSDEM release ...
• Questions ?
Why use the GPU?
APUs – GPU faster than CPU

- Tons of un-used Compute Units across your APU
- Sadly double precision is slower.
  - And Precision is non-negotiable for spreadsheets IEE764 required.
- Better power usage per flop.

Flops : note the log scale ...

1. for some ops: things GPU's were designed for, like LiteCoin mining ...
Developers behind the calc re-work:

Kohei Yoshida:  
MDDS maintainer  
Heroic calc core re-factorer  
Code Ninja etc.

Markus Mohrhard  
Calc maintainer,  
Chart2 wrestler  
Unit tester par Excellence  
etc.

Matus Kukan  
Data Streamer,  
G-builder,  
Size optimizer ..

Jagan Lokanatha  
Kismat Singh

A large OpenCL team,  
Particularly I-Jui (Ray) Sung
Spreadsheet Geometry

An early Spreadsheet C 3000 BC
Aspect ratio: 8:1
Contents:

Victory against every land ...
who giveth all life forever ...

50% of spreadsheets used to make business decisions.

Excel 2003
64k x 256
Aspect: 256:1

Excel 2010
10^6 x 16k
Aspect: 16:1

The 'Broom Handle' aspect ratio.

Columnar data structures
Spreadsheet Core Data Storage
Abstraction of Cell Value Access
ScBaseCell Usage \textbf{(Before)}

ScDocument

- Undo / Redo
- Change Tracking
- Content Rendering
- Excel Filter (xls, xlsx)
- RTF Filter
- Quattro Pro Filter
- HTML Filter
- External Reference

Document Iterators

- CSV Filter
- Conditional Formatting
- Chart Data Provider
- Cell Validation
- DIF Filter
- SYLK Filter
- DBF Filter
- CppUnit Test

UNO API Layer

VBA API Layer

ODF Filter
Abstraction of Cell Value Access
ScBaseCell Usage (After)

ScDocument

Document Iterators

Biggest calc core re-factor in a decade+

Dis-infecting the horrible, long-term, inherited structural problems of Calc.

Lots of new unit tests being created for the first time for the calc core.

Moved to using new 'MDDS' data structures.

2x weeks with no compile ...
Before (ScBaseCell)

ScDocument

ScTable

ScColumn

ScValueCell

ScStringCell

ScEditCell

ScFormulaCell

ScNoteCell*

ScBaseCell

Scattered pointer chasing walking cells down a column ...

Broadcaster (8 bytes)
Text width (2 bytes)
Cell type (1 byte)
Script type (1 byte)
After (mdds::multi_type_vector)

ScDocument

- ScTable
  - ScColumn
    - svl::SharedString block
    - double block
    - EditTextObject block
    - ScFormulaCell block
- Broadcasters
- Cell notes
- Text widths
- Script types
- Cell values
Iterating over cells (old way)

... loop down a column ... and the inner loop:

```cpp
double nSum = 0.0;
ScBaseCell* pCell = pCol >maItems[nColRow].pCell;
++nColRow;
switch (pCell->GetCellType())
{
  case CELLTYPE_VALUE:
    nSum += ((ScValueCell*)pCell)->GetValue();
    break;
  case CELLTYPE_FORMULA:
    // ... something worse ...
  case CELLTYPE_STRING:
  case CELLTYPE_EDIT:
    // ...
  case CELLTYPE_NOTE:
    // ...
}
```
Iterating over cells (new way)

double nSum = 0.0;

for (size_t i = 0; i < nChunkLength; i++)
    nSum += pDoubleChunk[i];

ONO. from a vectoriser ...
Shared Formula
Before

ScFormulaCell → ScTokenArray → ... Tokens
ScFormulaCell → ScTokenArray → ... RPN
...
After

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCell

ScFormulaCellGroup

ScTokenArray

... Tokens

... RPN
Memory usage

Test document used:
Shared string re-work

• String comparisons were slow
  • Also not tractable for a GPU
  • Case-insensitive equality is a hard problem – ICU & heavy lifting.

• String comparisons a lot in functions, and Pivot Tables.

• Shared string storage is useful.

• So fix it …
Concept

svl::SharedString

svl::SharedString

svl::SharedString

svl::SharedStringPool

Original string pool

Upcased string pool
String comparison (old way)

```cpp
util::TransliterationWrapper* pTransliteration = NULL;
OUString aStr1, aStr2;

if (bCaseSensitive)
    // Case sensitive transliterator.
    pTransliteration = ScGlobal::GetCaseTransliteration();
else
    // Case insensitive transliterator.
    pTransliteration = ScGlobal::GetpTransliteration();

// Parse both strings to check equality.
bool bEqual = pTransliteration->isEqual(aStr1, aStr2);
```
String comparison (new way)

svl::SharedString aStr1, aStr2;

const rtl_uString* p1;
const rtl_uString* p2;

if (bCaseSensitive)
{
    // Get pointers to original strings in the pool.
    p1 = aStr1.getData();
    p2 = aStr2.getData();
}
else
{
    // Get pointers to upcased strings in the pool.
    p1 = aStr1.getDataIgnoreCase();
    p2 = aStr2.getDataIgnoreCase();
}

// Compare pointer values.
bool bEqual = p1 == p2;
OpenCL / calculation ...
Why OpenCL & HSA …

- GPU and CPU optimisation …
  - Why write custom SSE2/SSE3 etc. assembly detect arch, and select backend cross platforms.
  - Instead get OpenCL (from APU vendor) to generate the best code …

- Heterogenous System Architecture rocks:
  - An AMD64 like innovation:
  - shared Virtual Memory Address space & pointers: GPU ↔ CPU.
  - Avoid wasteful copies, fast dispatch
  - Great OpenCL 2.0 support.
  - Use the right Compute Unit for the job.
#pragma OPENCL EXTENSION cl_khr_fp64: enable
int isNan(double a) { return isnan(a); }
double legalize(double a, double b) { return isNan(a)?b:a;}
double tmp0_0_fsum(__global double *tmp0_0_0)
{
    double tmp = 0;
    {
        int i;
        i = 0;
        tmp = legalize(((tmp0_0_0[i])+(tmp)), tmp);
        i = 1;
        tmp = legalize(((tmp0_0_0[i])+(tmp)), tmp);
        i = 2;
        tmp = legalize(((tmp0_0_0[i])+(tmp)), tmp);
    } // to scope the int i declaration
    return tmp;
}
double tmp0_nop(__global double *tmp0_0_0)
{
    double tmp = 0;
    int gid0 = get_global_id(0);
    tmp = tmp0_0_fsum(tmp0_0_0);
    return tmp;
}
__kernel void DynamicKernel_nop_fsum(__global double *result, __global double *tmp0_0_0)
{
    int gid0 = get_global_id(0);
    result[gid0] = tmp0_nop(tmp0_0_0);
}
__kernel void
tmp0_0_0_reduction(__global double* A,
  __global double *result,
  int arrayLength, int windowSize)
{
  double tmp, current_result = 0;
  int writePos = get_group_id(1);
  int lidx = get_local_id(0);
  __local double shm_buf[256];
  int offset = 0;
  int end = windowSize;
  end = min(end, arrayLength);
  barrier(CLK_LOCAL_MEM_FENCE);
  int loop = arrayLength/512 + 1;
  for (int l = 0; l < loop; l++) {
    tmp = 0;
    int loopOffset = l*512;
    if((loopOffset + lidx + offset + 256) < end) {
      tmp = legalize(((A[loopOffset + lidx + offset])+
        (tmp)), tmp);
      tmp = legalize(((A[loopOffset + lidx + offset +
        256])+(tmp)), tmp);
    } else if ((loopOffset + lidx + offset) < end)
      tmp = legalize(((A[loopOffset + lidx + offset])+
        (tmp)), tmp);
    shm_buf[lidx] = tmp;
    barrier(CLK_LOCAL_MEM_FENCE);
    for (int i = 128; i > 0; i /= 2) {
      if (lidx < i)
        shm_buf[lidx] = ((shm_buf[lidx])+
          (shm_buf[lidx + i]));
      barrier(CLK_LOCAL_MEM_FENCE);
    }
    if (lidx == 0)
      current_result =((current_result)+(shm_buf[0]));
    barrier(CLK_LOCAL_MEM_FENCE);
  }
  if (lidx == 0)
    result[writePos] = current_result;
}

double tmp0_0_fsum(__global double *tmp0_0_0) {
  double tmp = 0;
  int gid0 = get_global_id(0);
  tmp = ((tmp0_0_0[gid0])+(tmp));
  return tmp;
}
double tmp0_nop(__global double *tmp0_0_0) {
  double tmp = 0;
  int gid0 = get_global_id(0);
  tmp = tmp0_0_fsum(tmp0_0_0);
  return tmp;
}
__kernel void
DynamicKernel_nop_fsum(__global double *result,
  __global double *tmp0_0_0)
{
  int gid0 = get_global_id(0);
  result[gid0] = tmp0_nop(tmp0_0_0);
}
Performance numbers for sample sheets.

- min_max_avg_r
- destination-workbook
- dates-worked
- stock-history
- ground-water

Shorter is better

30x → 500x faster for these samples vs. the legacy software calculation on Kaveri.

Yet another log plot ... milliseconds on the X axis ...
How that works in practice:
Enabling Custom Calculation

- Turn on OpenCL computation: **Tools → Options**
Enabling OpenCL goodness

- Auto-select the best OpenCL device via a micro-benchmark
- Or disable that and explicitly select a device.
Big data needs Document Load optimization
Parallelized Loading ...

- Desktop CPU cores are often idle.
- XML parsing:
  - The ideal application of parallelism
  - SAX parsers:
    - "Sucking icAche eXperience" parsers
      - read, parse a tiny piece of XML & emit an event …
      - punch that deep into the core of the APP logic, and return ..
      - Parse another tiny piece of XML.
  - Better APIs and impl's needed: Tokenizing, Namespace handling etc.
- Luckily easy to retro-fit threading ...
- Dozens of performance wins in XFastParser.
XML format lameness ...

- Spreadsheets have a great way of expressing repeated formulae:
  - R1C1 notation:
    - =SUM($A$1:$A$5)-A1
    → =SUM(R1C1:R5C1)-R(-2)C(-1)
  - Looks ugly – but it’s constant down a column.
  - Lunatic standardizers for ODF ( & OOXML ) ignored me on this …
- Formulae hard and expensive to parse, so don’t …
  - Predictive generation down a column & comparison.
    - Removes tons of token allocations etc.
Parallelised load:
(boxes are threads).

- Split XML Parse & Sheet populate

- Parallelised Sheet Loading ...

- Parallel to GPU compilation

=COVAR(A1:A300,B1:B300)

→ OpenCL code

→ Ready to execute kernels

Tools->Options->Advanced->”Experimental Mode” required for parallel loading
Does it work? with GPU enabled

Wall-clock time to load set of large XLSX spreadsheets: 8 thread Intel machine

<table>
<thead>
<tr>
<th>Spreadsheet Name</th>
<th>Calc 4.1.3</th>
<th>Calc</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>num-formula-2-sheets-1m.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>numbers-formula-8-sheets-100k.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>numbers-formula-100k.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>numbers-100k.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sumifs-testsheet.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stock-history.xlsm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>matrix-inverse.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mandy.xlsm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mandy-no-macro.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>groundwater-daily.xlsm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dates-worked.xlsx</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Shorter is better

Apologies for another log scale: Average 5X vs. 4.1.3
Quick demo & questions on Calc / GPU bits?
Other LibreOffice 4.2 Features
UI → Layout conversion

• 70% complete +280 dialogs in 4.2
• Thanks to:
  - Caolán McNamara (Red Hat), Manal Alhassoun (KACST), Olivier Hallot (EDX), Faisal M. Al-Otaibi (KACST), Laurent Balland-Poirier, Efe Gürkan Yalaman, Krisztian Pinter, Jan Holesovsky (Collabora), Andras Timar (Collabora), Cao Cuong Ngo, Gergo Mocsi, Katarina Behrens, Abdulmajeed Ahmed (KACST), and Alia Almusaireae (KACST)
• Checkout Caolan's blog to help out ...

![UI Layout Dialog Conversion](chart.png)

- Layout UI
- old dlg
- old tab-page
- Remaining
Unit testing ...

- 216 new CPPUNIT_TESTs
- 2160+ new CPPUNIT_ASSERTS
- Lots of format import / test / export / re-import / re-test – round-trip interop.
Much improved Android Tablet / Phone: Impress remote

With thanks to Artur Dryomov

Connection failed

Make sure you enabled remote control. Go to "Tools → Options → LibreOffice Impress → General" in LibreOffice Impress.

You should enable experimental features at "Tools → Options → LibreOffice → Advanced" as well.
Initial iOS Impress remote control

With thanks to Siqi Liu – sign up for the Beta

I'm not sure if I'm missing something but what should be a simple task just doesn't want to work. I'm trying to add a drop shadow to a UIView in iOS 6. I'm using storyboards and auto layout. I'm drawing the UIView in the storyboard scene with a white
GDrive integration (via libcmis)

With thanks to Cuong Cao Ngo and Cedric Bosdonnat
Chart: new Trend lines

Thanks to Tomaž Vajngerl
New Feature: Math

New Math Panel with colors

Thanks to: Marcos Paulo de Souza
Writer: New Default Template

Thanks to:
Ahmad Harthi (KACST)
Faisal M. Al-Otaibi (KACST)
Thanks to Tamás Zolnai
New Feature: Sifr Icon Set

Thanks to Issa Alkurtass (KACST), Norah A. Abanumay (KACST)
New Feature: Start Screen

Large, beautiful, anti-aliased previews in tiles ...
Better Windows Integration ...

- Group Policy Integration / Active Directory lockdown (Hungarian E-Government Competence Center)
- Windows Grouped in Task-bar (Jesus Corrius)
- Recent Documents in Task-bar (Jesus Corrius & Fridrich Strba)
New Feature: Firebird Database

Bringing love to the 'Base' backend
Thanks to
Andrzej Hunt
Lionel Elie Mamane
An Experimental feature replacing HSQLDB in 4.3 ...
Misc. New Features

- OOXML import/export fidelity
  - round-trip of un-interpreted data (*CloudOn, SyneZip, Igalia*)
  - Agile encryption (*Tomaz Vajngerl*)
- BCP47 Language Tags (*Eike Rathke*)
- Import filter for various e-book formats, mostly Palm-based (*David Tardon*)
  - FictionBook 2, PalmDoc, PeanutPress (eReader), Plucker, TealDoc, zTXT
- Import of more legacy Mac document types (*Laurent Alonso*)
AbiWord Import

Thanks to Fridrich Strba

AbiWord 2.0 - The Next Step
(Toward World Domination)
Martin Siever and Dominic Lackowicz

New features in 2.0
AbiWord 2.0 has many new features not present in the 1.0 release. There are features that provide improvements to the AbiWord User Interface and those that increase the richness of the Word Processing document. The following table lists new Word Processing features and the file formats the feature is exported to and imported from. Of course all features are supported by the AbiWord native XML file format.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Imported from</th>
<th>Exported to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
<td>MS Word, RTF, WordPerfect, HTML, OpenOffice.org, Draw (other XML formats)</td>
<td>MS Word, RTF, WordPerfect, HTML, OpenOffice.org, Draw (other XML formats)</td>
</tr>
<tr>
<td>Footnotes</td>
<td>RTF, MS Word</td>
<td>RTF, Latex</td>
</tr>
<tr>
<td>Endnotes</td>
<td>RTF, MS Word</td>
<td>RTF, Latex</td>
</tr>
<tr>
<td>Revision Marks</td>
<td>RTF, MS Word</td>
<td>RTF, Latex</td>
</tr>
<tr>
<td>MailMerge</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The Table feature in AbiWord is very powerful. Cells within tables can be merged either horizontally or vertically via an easy-to-use non-modal dialog. Rows and Columns can be adjusted interactively by dragging ruler controls or table lines. The new build includes a powerful automatic table insert widget which allows table dimensions to be created interactively. We also have the ability to nest tables to arbitrary depth. This feature is not available in any other product.

<table>
<thead>
<tr>
<th>Feature</th>
<th>New to Unix for 2.0</th>
<th>New to Unix for 2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-allowed text</td>
<td>New to Unix for 2.0</td>
<td></td>
</tr>
<tr>
<td>Gtk 2.0 GUI elements</td>
<td>New to Unix for 2.0</td>
<td></td>
</tr>
<tr>
<td>GNOME integration</td>
<td>Drag and drop images, Embed AbiWord in Nautilus, Drag and drop URL's in Galeon/Mozilla</td>
<td></td>
</tr>
</tbody>
</table>

[Not all of AbiWord's table features (like nested tables) are available on these products.]
Initial Keynote Filter

Apple Keynote import

Since version 4.2, LibreOffice can import Apple Keynote presentations.

This is a preview release: we can import many parts of the content, but details (like formatting) are often lost.

Anyway, the main author of the code believes in the "release early, release often."  

Samples:

Thanks to David Tardon  
(RedHat)
GNOME 3.0 Menu

- Activities
- LibreOffice Writer

Menu:
- File
- Edit
- View

Menu:
- New
- Format
- Tools

Menu:
- Preferences...
- Help
- About LibreOffice
- Quit LibreOffice
Commits per month

Code commits per month by affiliation

- Tata Consultancy Services
- SYNERZIP
- SIL
- RedHat
- Oracle
- Openismus
- New Contributors
- MultiCoreWare
- Linagora
- Lanedo
- Known contributors
- KACST
- Intel
- Funky
- IBM
- Igalia
- Collabora
- CodeWeavers
- Linoded
- SUSE
- CloudOn
- Bobiciel
- Canonical
- Savannah
- Assigned
- KACST
- CodeThink
- Bobiciel
- Assigned
- ALTA
Active developers per month

Active developers each month by affiliation

Tata Consultancy Services
SYNERZIP
RedHat
Openismus
New Contributors
MultiCoreWare
Linagora
KACST
Igalia
SUSE
Collabora
Canonical
Assigned
ALTA

[Chart showing active developers per month by affiliation with various companies and contributors.]
Fast Tiled Rendering in textures

Tablet display

LibreOffice / Android 4.x

This is a picture of our wedding, with some nice text flow going on around it, of course.

A small table with an embedded image.

Large Pre-rendered area for fast pan / zoom etc.

And chart of the week:

Persistant Sins

- Jealousy
- Pride
- Lust
- Idleness
Misc. Project Bits Recently
LibreOffice Project & Software

• Open Source / Free Software
• One million new unique IPs per week (that we can track)
  • Double the weekly growth one year ago.
• Tens of millions of users, and growing fast.
• Hundreds of contributing coders.
• Around a thousand developers (including QA, Translators, UX etc.)
• http://www.libreoffice.org/
Advisory Board Members

This slide’s layout is a victim of our success here ...
LibreOffice Conclusions

• LibreOffice is innovating:
  • Going interesting places no-one has gone before:
    - OpenCL in a generic spreadsheets a first
    - Why write 5x hand-coded assembler versions and select per platform.
      • there is already a tool for that.
  • Run your workload on the right Compute Unit to save time & battery.

• LibreOffice is growing & executing
  • We’re improving a lot – but there is still a long way to go.
  • We need your help ! Please do see me & get involved ...

• LibreOffice has ambitious future plans
  • We need your help to accelerate them ...

• Thanks for all of your help and support !

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