

# LibreOffice Calc

## *Now available on your GPU*

Michael Meeks <[michael.meeks@collabora.com](mailto: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 ?



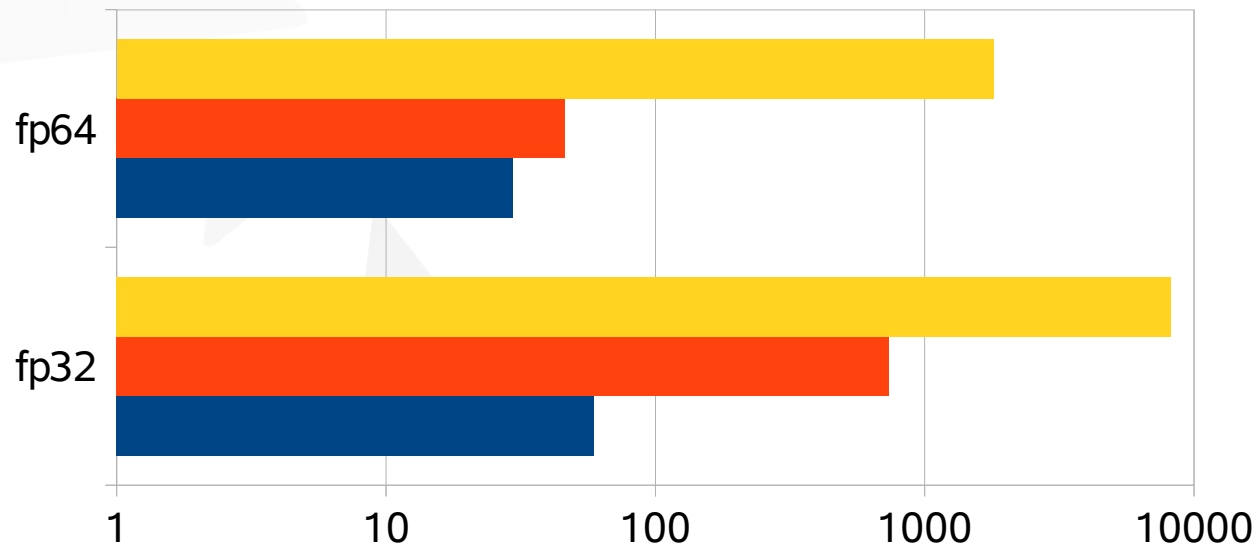
# Why use the GPU ?



# APUs – GPU faster than CPU<sup>1</sup>

- Tons of un-used Compute Units across your APU
- Sadly double precision is slower.
  - And Precision is non-negotiable for spreadsheets IEEE764 required.
- Better power usage per flop.

Numbers based on a Kaveri 7850K APU - & top-end discrete Graphics card.



Flops : note the log scale ...

1. for some ops: things GPU's were designed for, like Litecoin mining ...



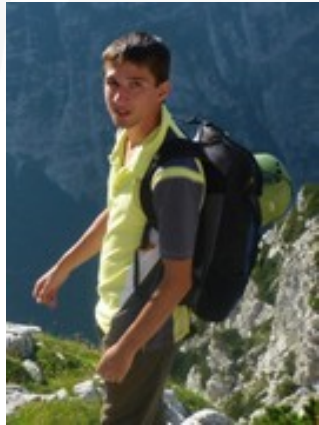
# Developers behind the calc re-work:



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 ..



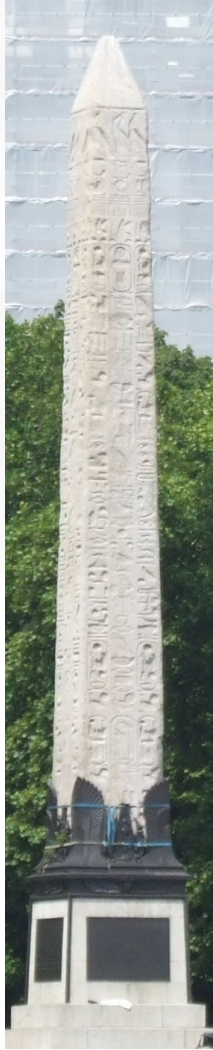
Jagan Lokanatha  
Kismat Singh



*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.**

Columnar data structures

Excel 2003

64k x 256

Aspect:  
256:1

Excel 2010

$10^6$  x 16k

Aspect:  
16:1

The 'Broom  
Handle'  
aspect  
ratio.

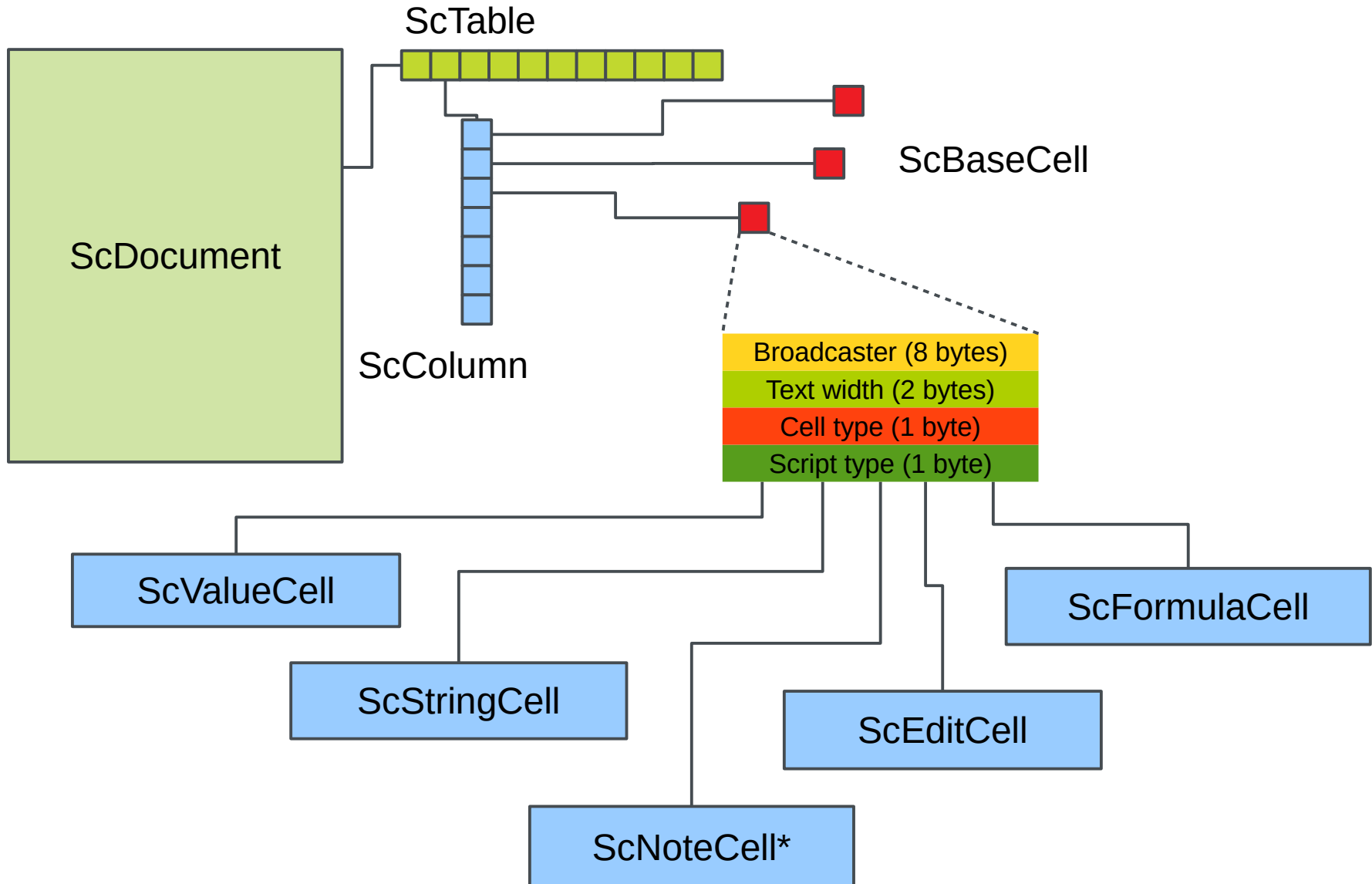




# Spreadsheet Core Data Storage



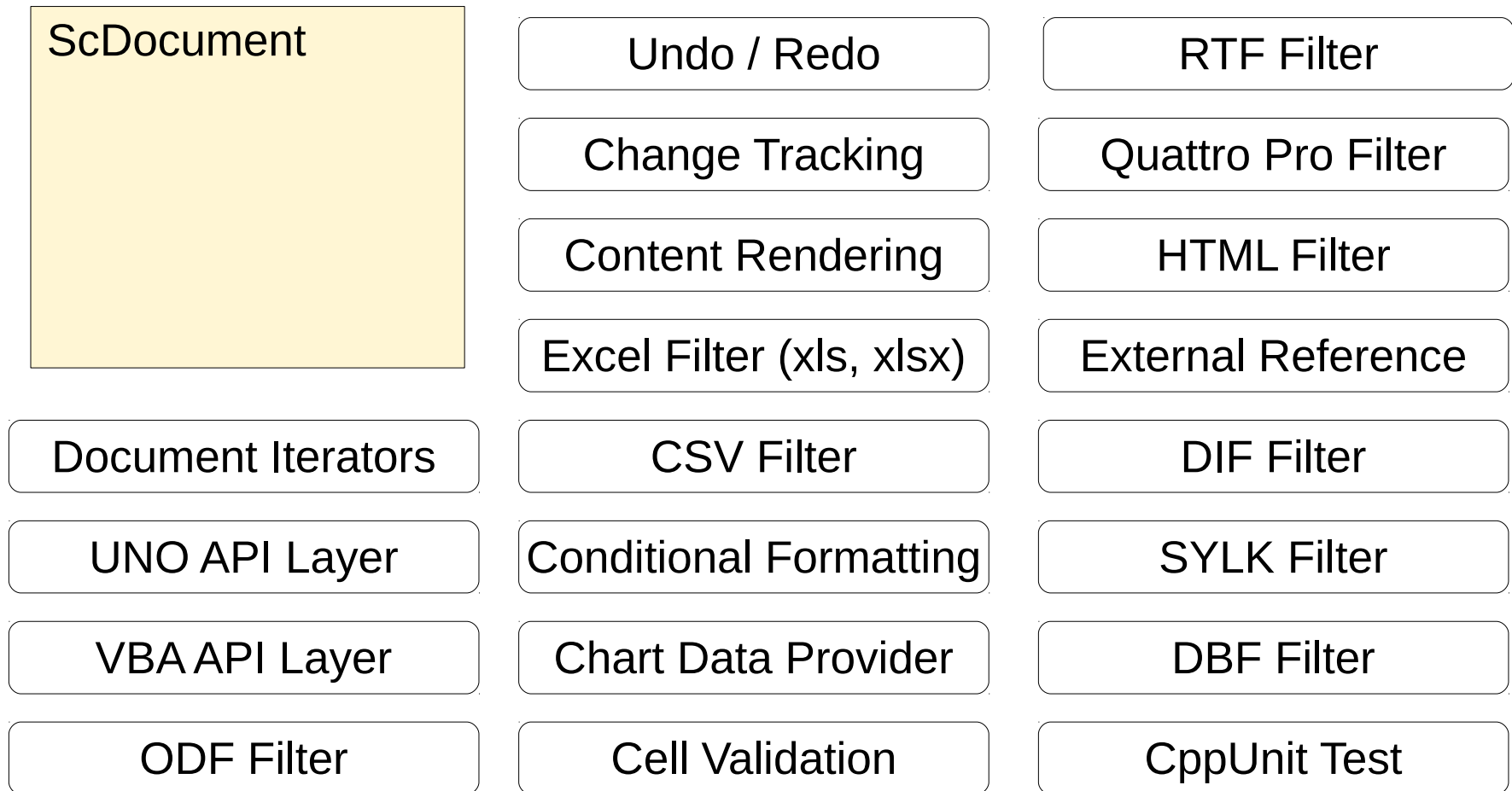
# Before (ScBaseCell)





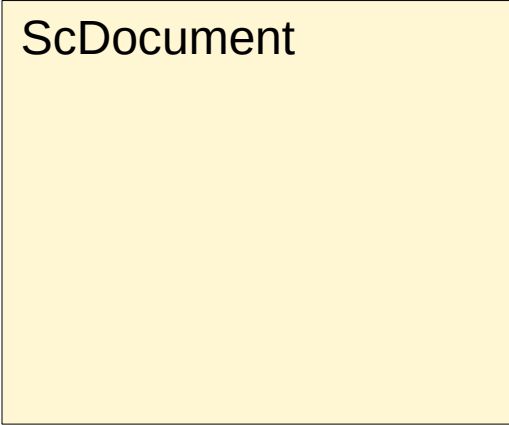
# Abstraction of Cell Value Access

ScBaseCell Usage (Before)



# Abstraction of Cell Value Access

ScBaseCell Usage (After)



ScDocument



Document Iterators

**Biggest calc core re-factor  
in a decade+**

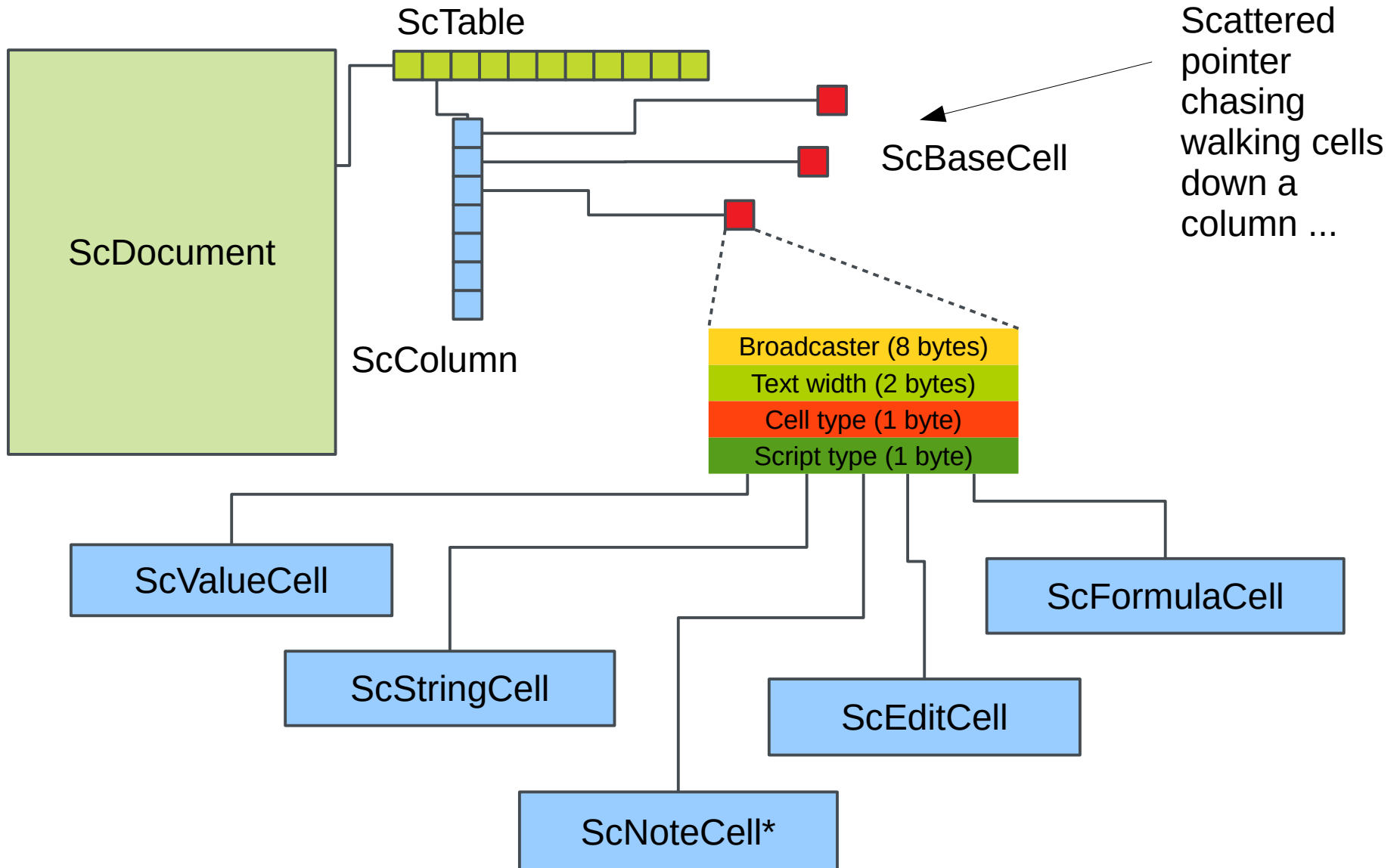
Dis-infecting the horrible,  
long-term, inherited  
structural problems of Calc.

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

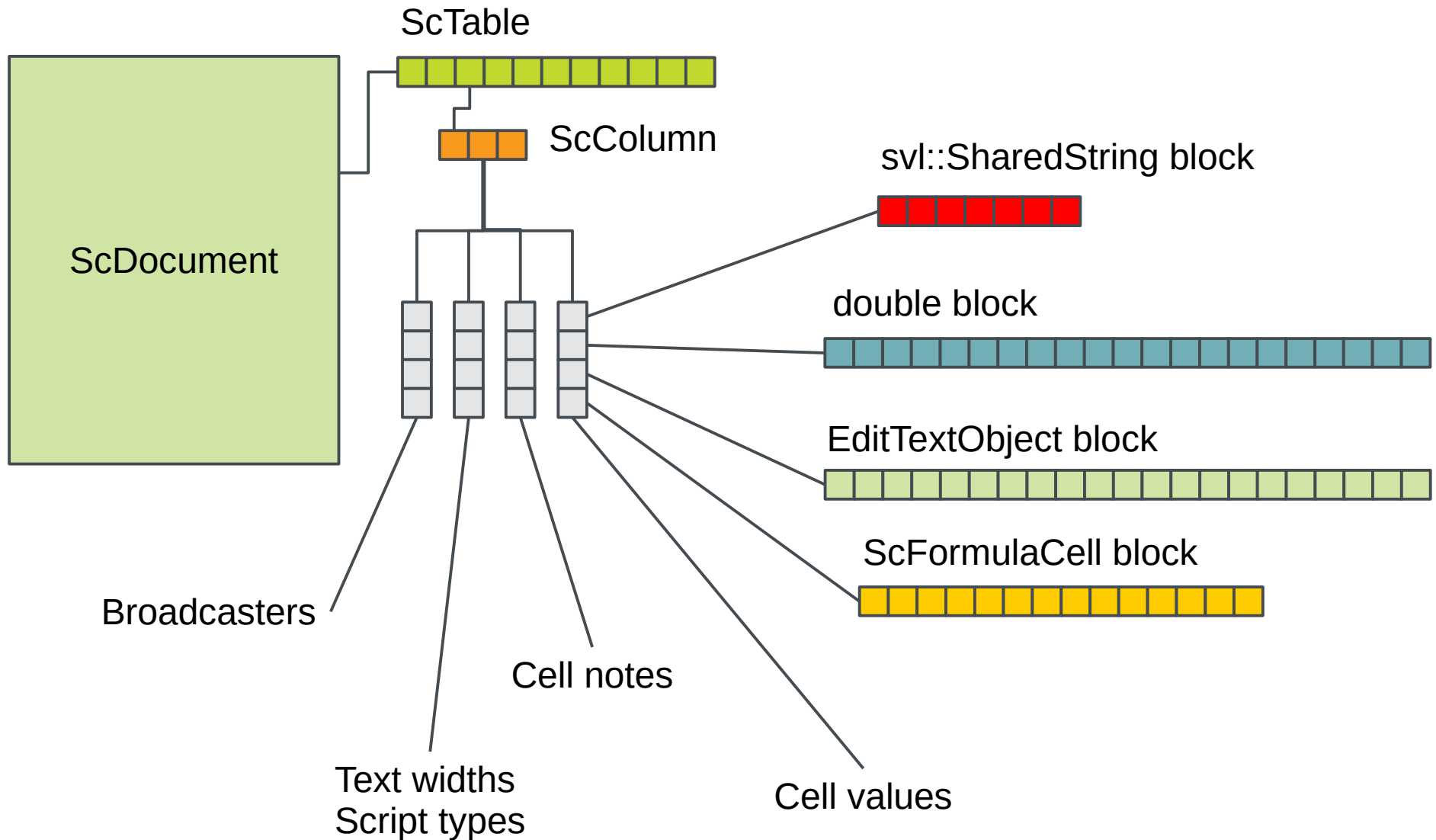
Moved to using new 'MDDS'  
data structures.

2x weeks with no compile ...

# Before (ScBaseCell)



# After (mdds::multi\_type\_vector)



# Iterating over cells (old way)

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

```
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 ...

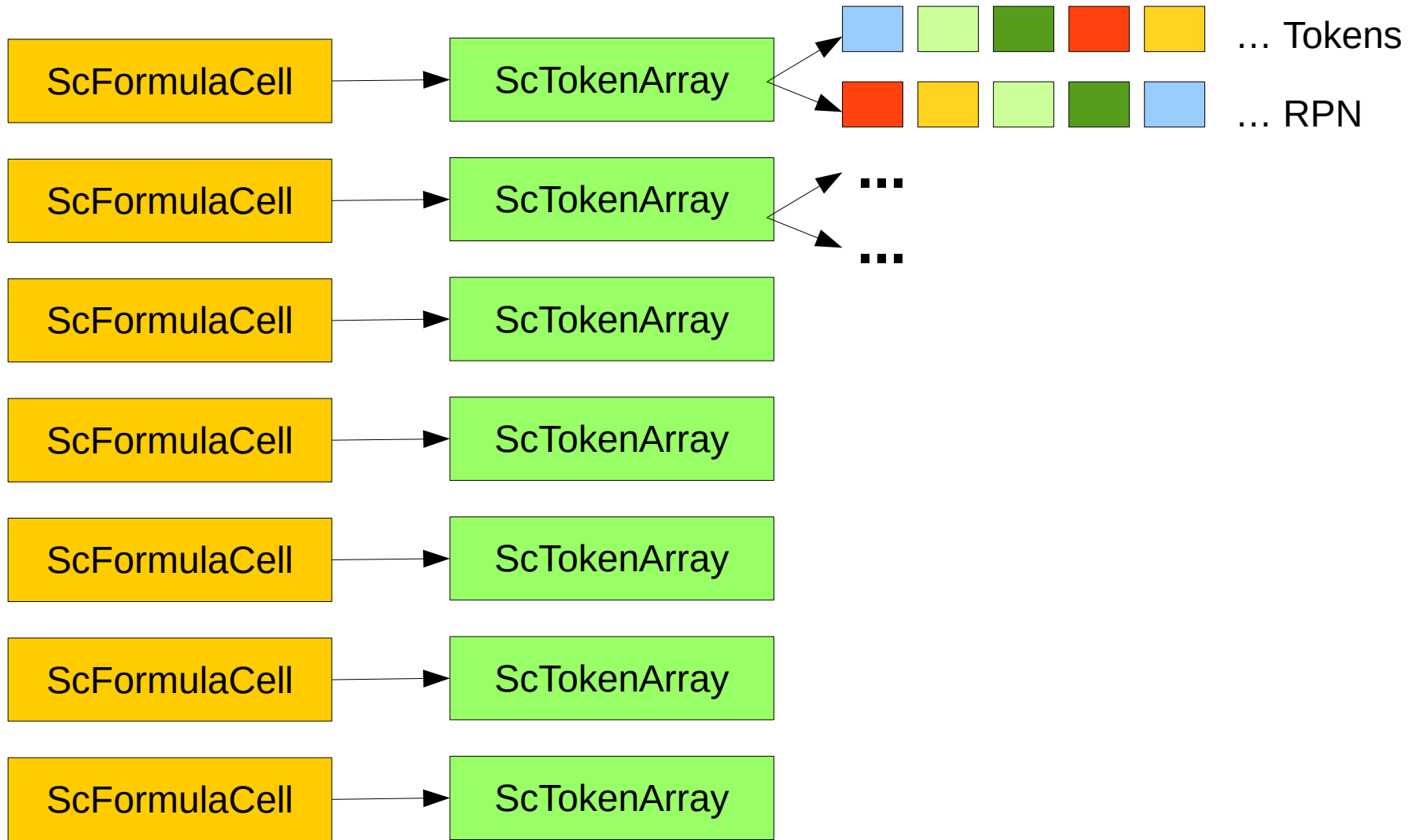


# Shared Formula

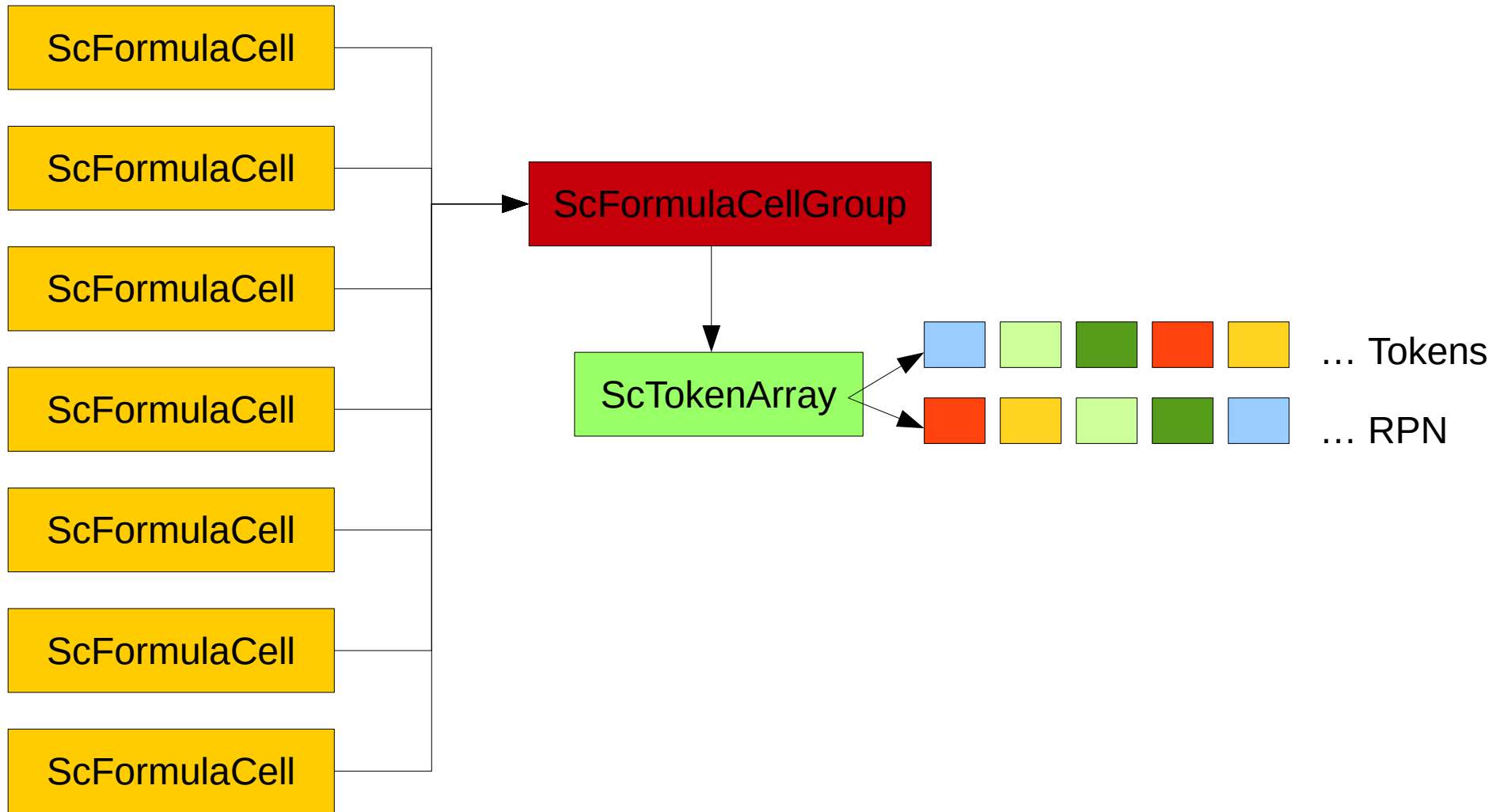




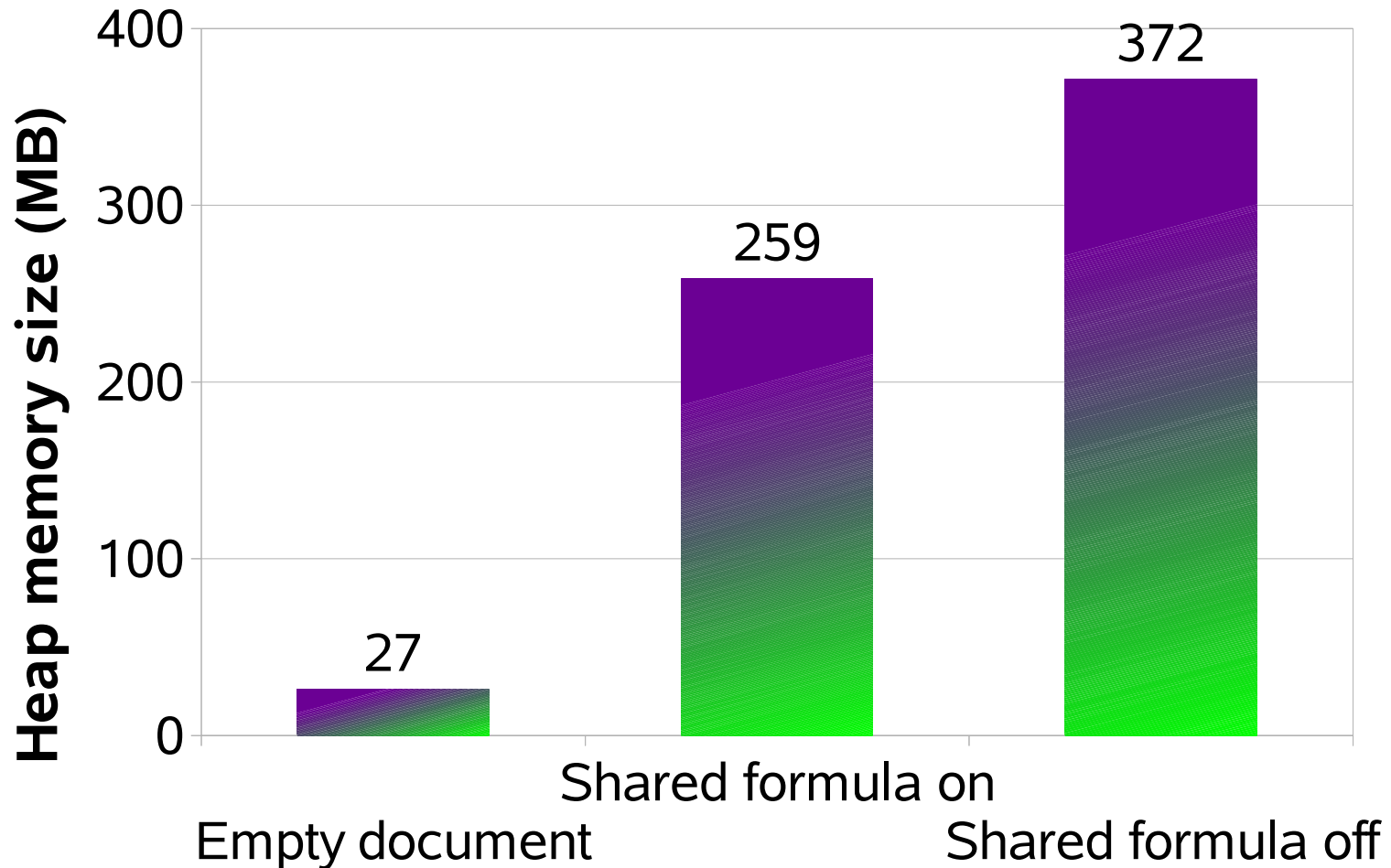
# Before



# After



# Memory usage



Test document used:

<http://kohei.us/wp-content/uploads/2013/08/shared-formula-memory-test.ods>

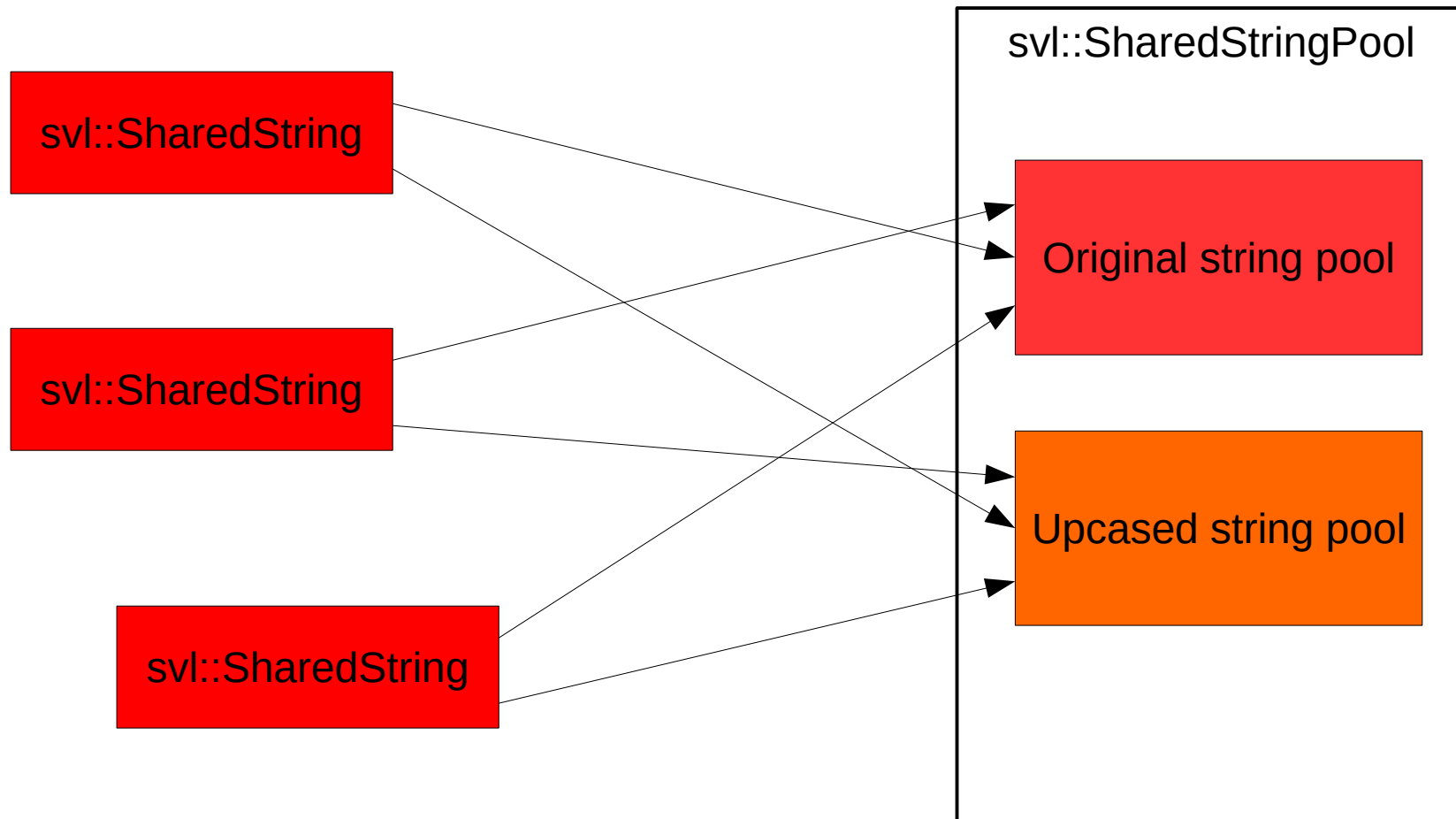


# Shared string re-work

- 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



# String comparison (old way)

```
utl::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;
```

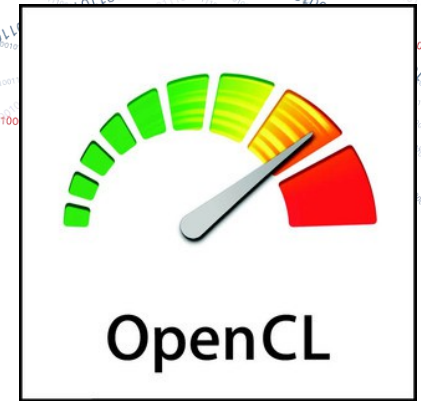




# OpenCL / calculation ...



# 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.



# Auto-compile Formula → OpenCL

```
#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)
{
```

	A	B	C
1	=SUM(\$B\$1:\$B\$3)	1	3
2		2	2
3		3	1

```

    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);
}
}

```

*Formulae compiled idly / on entry in a thread ... to hide latency.*

*Kernel generation thanks to:*



```

__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;
}

```

## The same formula for a longer sum ...

### Compiled from standard formula syntax

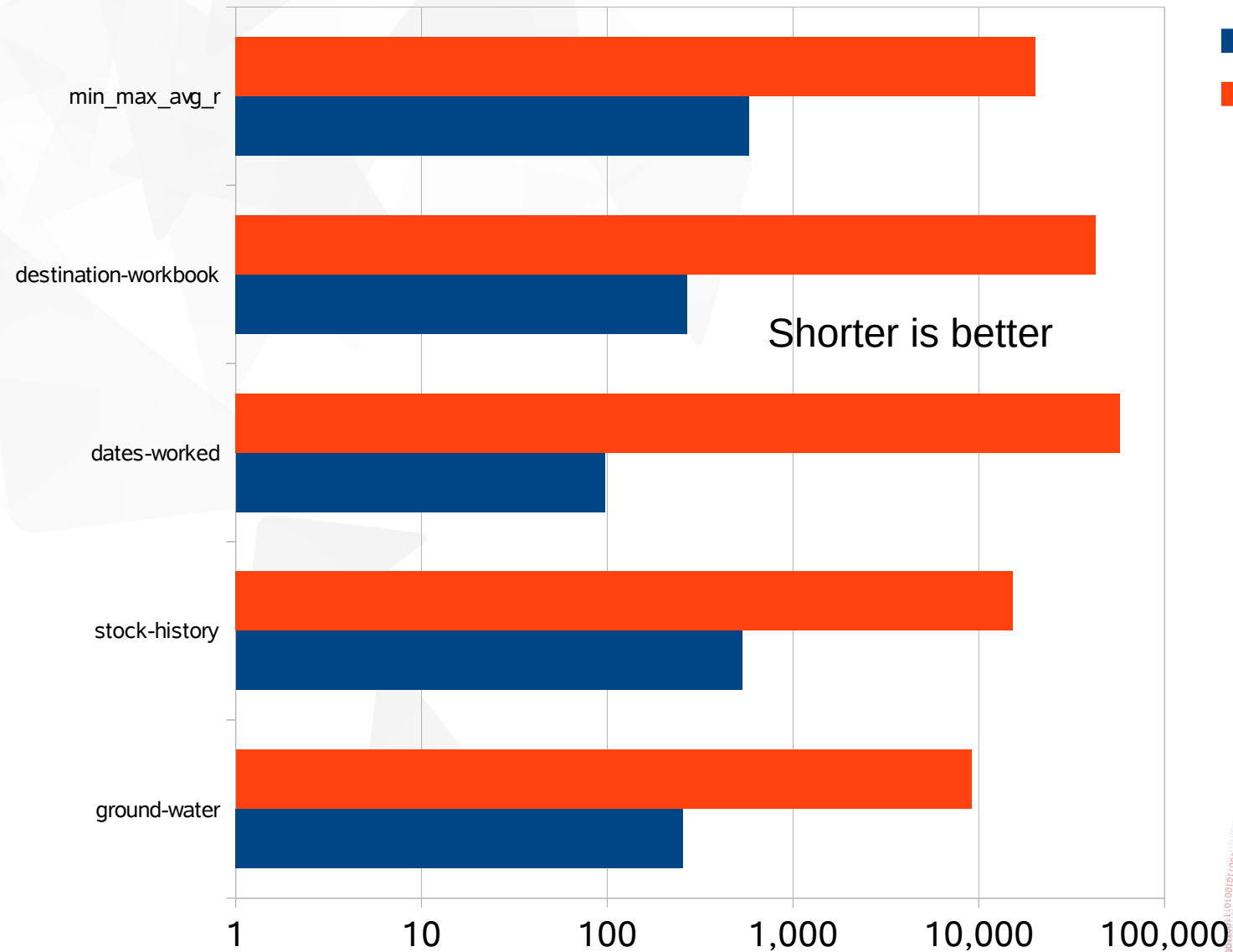
```

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.



*Yet another log plot ... milliseconds on the X axis ...*

■ GPU / OpenCL  
■ Software

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

on Kaveri.

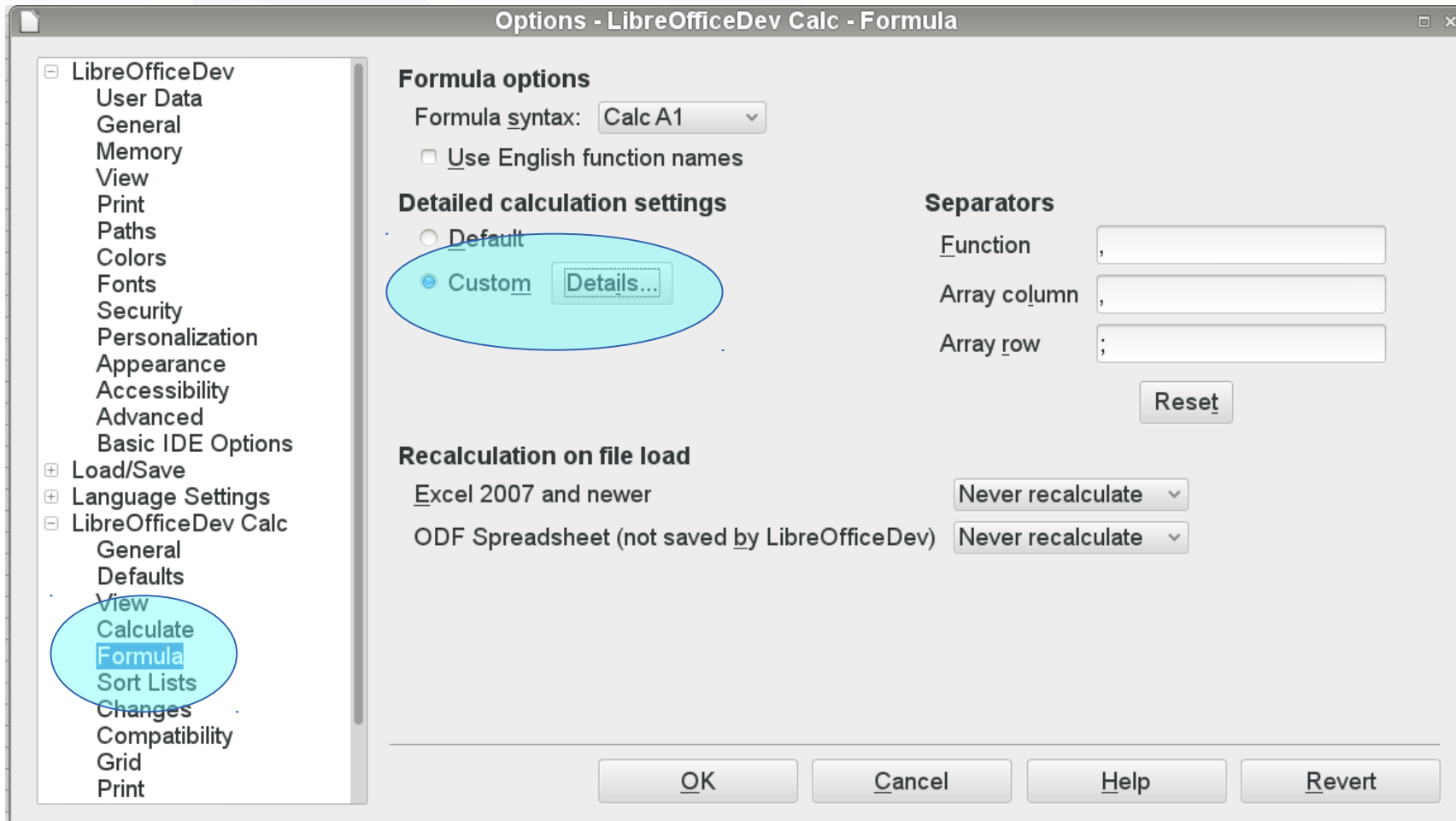


How that works in practise:



# Enabling Custom Calculation

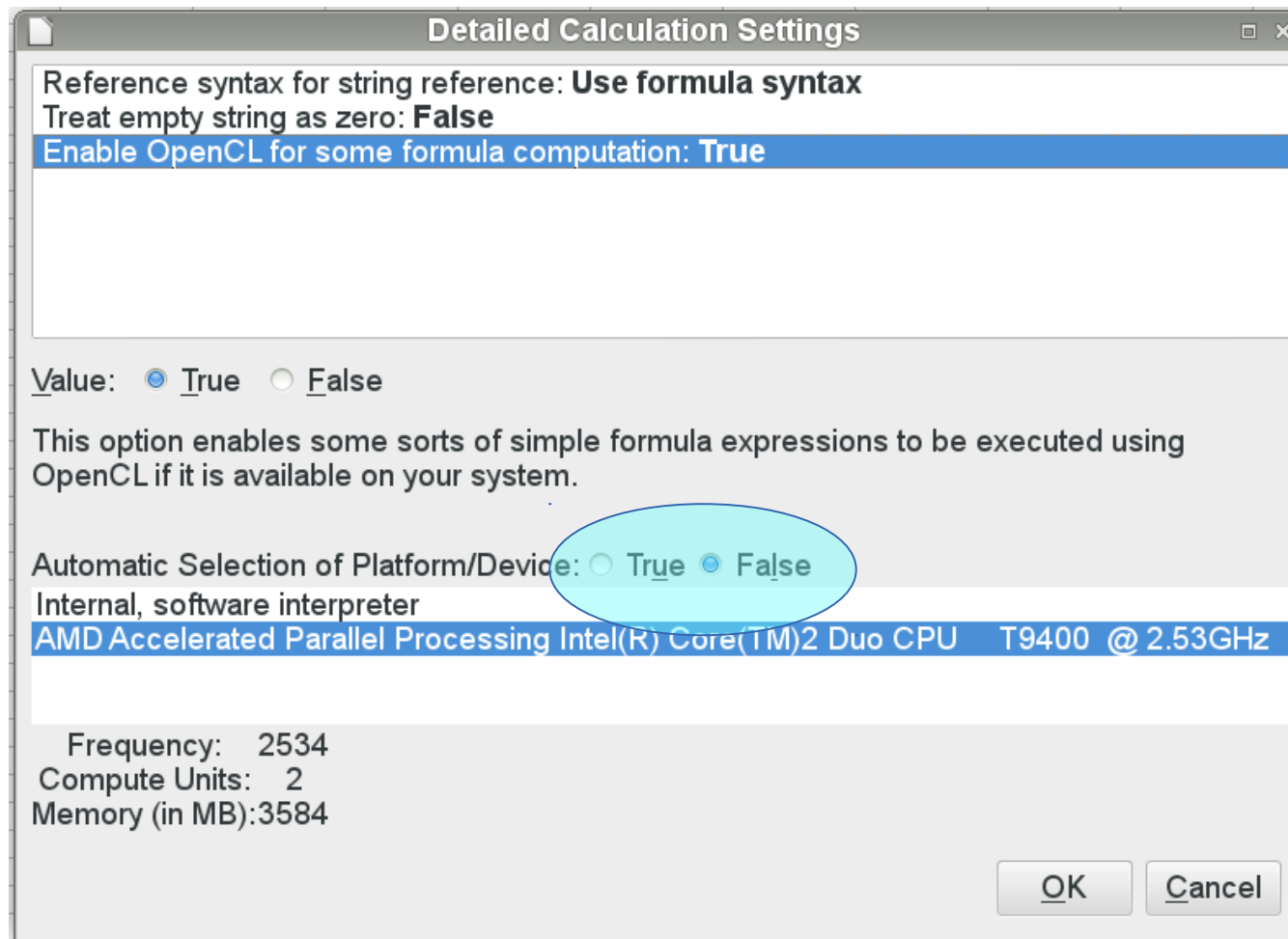
- Turn on OpenCL computation: **Tools** → **Options**





# 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:
    - “**S**ucking **i**c**A**che **e****X**perience” 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.



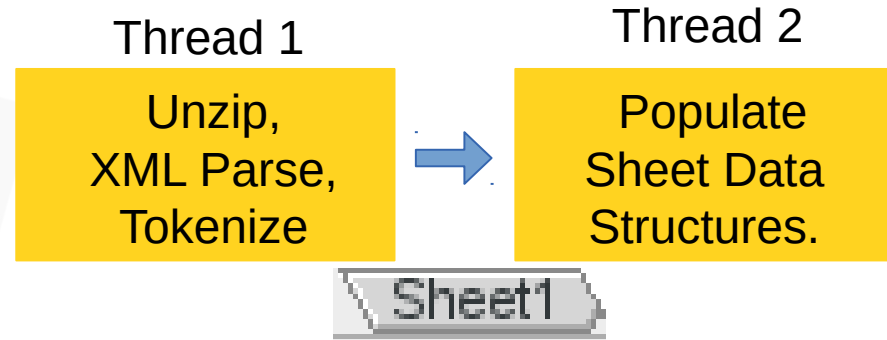
# XML format lameness ...

- 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.

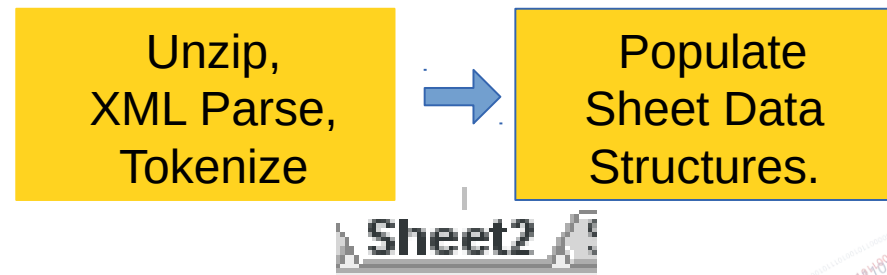


# Parallelised load: (boxes are threads).

- Split XML Parse & Sheet populate



- Parallelised Sheet Loading ...



Progress bar  
thread

... etc. Sheet 3 Sheet 4

- Parallel to GPU compilation

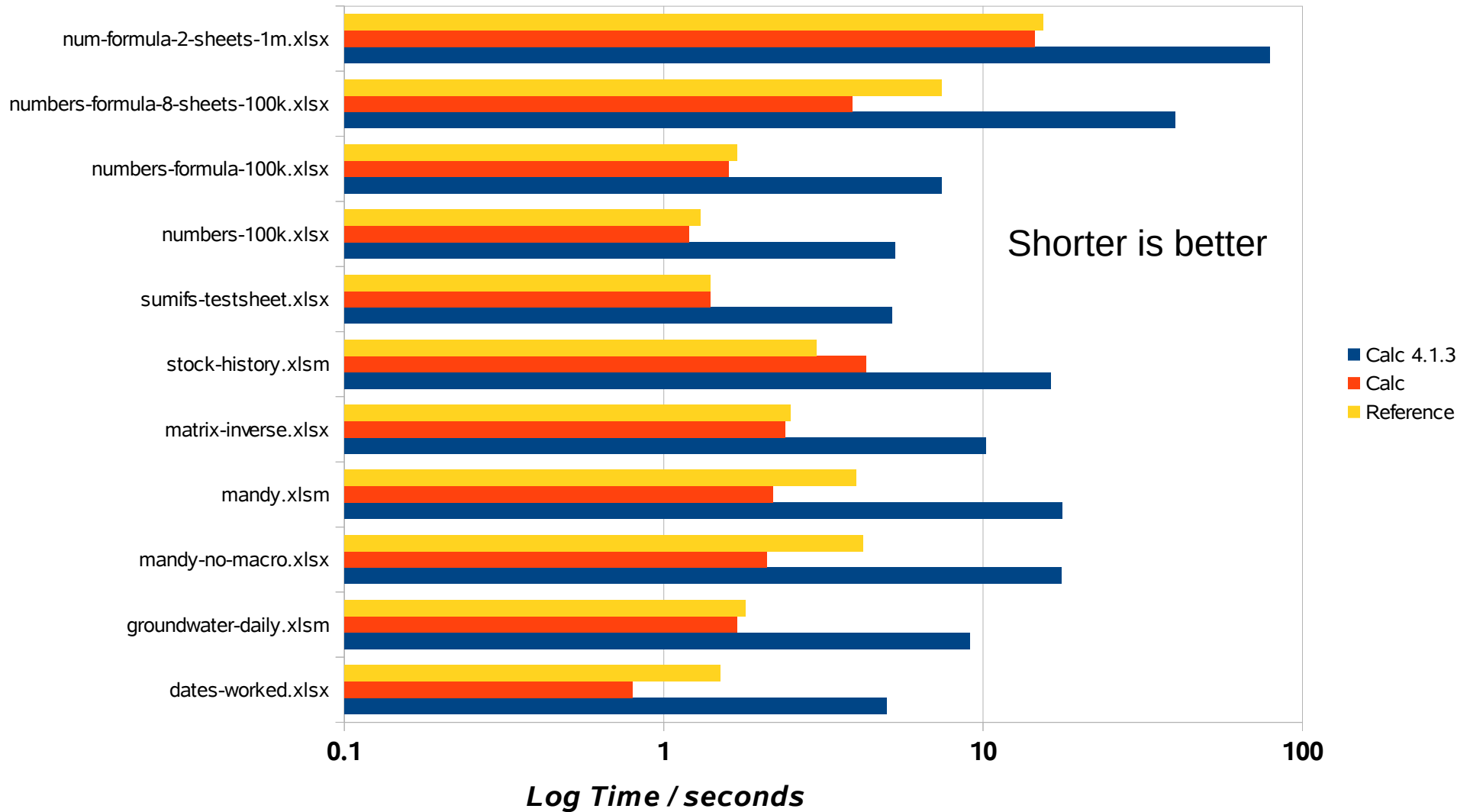
=COVAR(A1:A300,B1:B300)  
→ OpenCL code  
→ Ready to execute kernels

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



Apologies for another log scale: **Average 5X vs. 4.1.3**



# Quick demo & questions on Calc / GPU bits ?



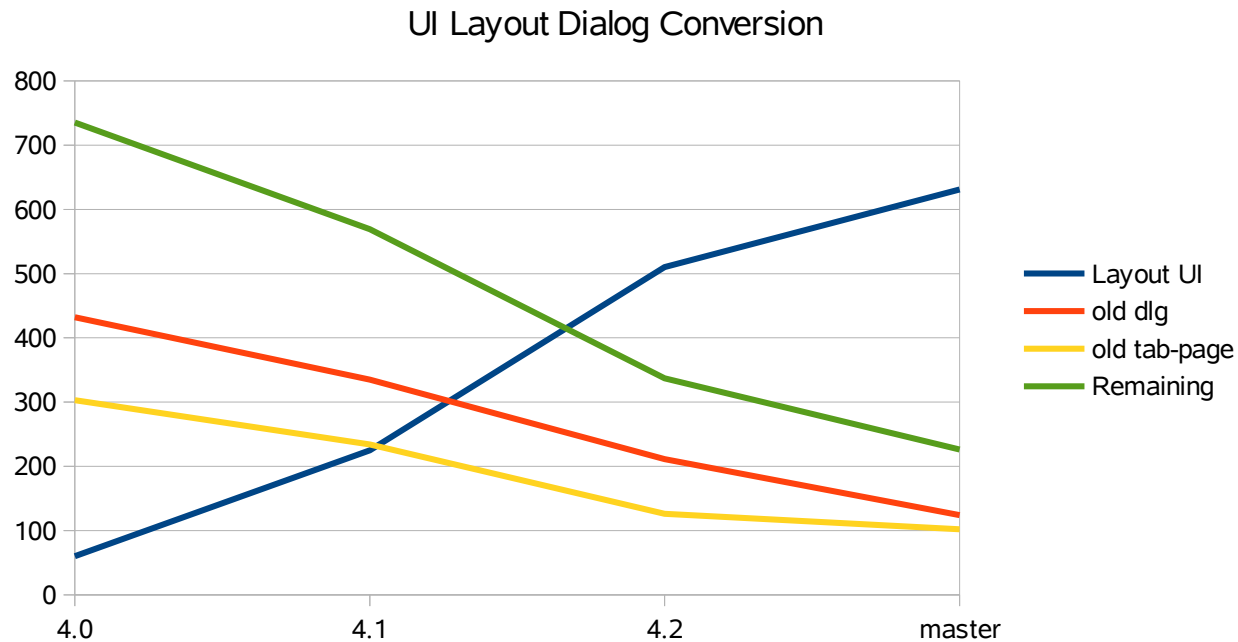
# Other LibreOffice 4.2 Features





# UI → Layout conversion

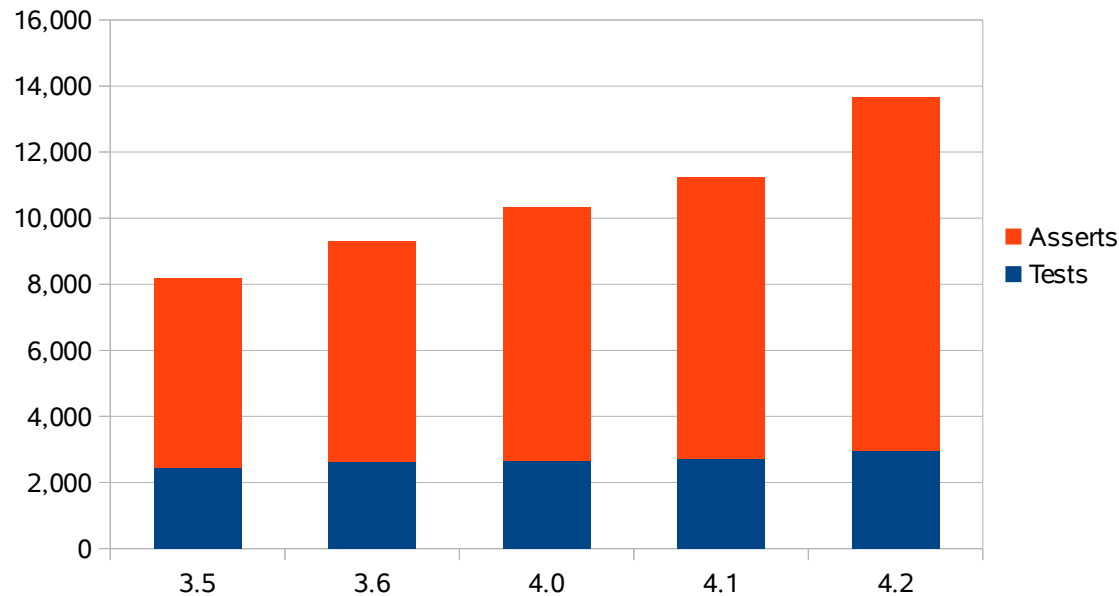
- 70% complete +280 dialogs in 4.2
  - Thanks to:
    - **Caolán McNamara** (Red Hat), Manal Alhassoun (KACST), Olivier Hallot (EDX), Faisal M. Al-Otaibi (KACST), Laurent Balland-Poirier, Efe Gürkan Yalaman, Krisztian Pinter, Jan Holesovsky (Collabora), Andras Timar (Collabora), Cao Cuong Ngo, Gergo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



# Unit testing ...

- 216 new CPPUNIT\_TESTS
- 2160+ new CPPUNIT\_ASSERTS
- Lots of format import / test / export / re-import / re-test – round-trip interop.

Growth in unit tests over time  
count of various CPPUNIT macros



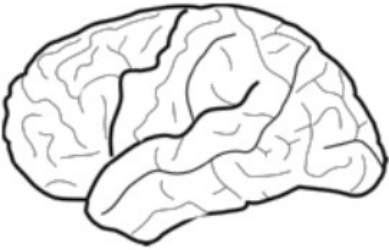
# Much improved Android Tablet / Phone: Impress remote

With thanks to *Artur Dryomov*

3G 1:30

Slide 3 of 12  
5 minutes remaining

Рефлекторная деятельность мозга



Navigation icons: back, home, recent apps

3G 1:31

Slide 3 of 12  
5 minutes remaining

<b>Мышление</b> 1	Ступени познания: • чувственная; • логическая. 2	Рефлекторная деятельность мозга 3
Ступени становления: • дологический; • логический. 4	<b>Понятия</b> 5	<b>Суждения</b> 6
<b>Умозаключения</b> Три вида умозаключений 7	Путь к умозаключению: • индукция; • дедукция; • аналогия. 8	<b>Типы мышления</b> По степени разобрatности: • дологический; • логический. 9
<b>Типы мышления</b> По характеру задач: • творческое; • практическое. 10	<b>Типы мышления</b> По содержанию задач: • прикладно-действительное; • научно-теоретическое; • словесно-логическое. 11	Предпочли мышления: • словесно-логическое; • образное; • пространственное; • математическое. 12

Navigation icons: back, home, recent apps

3G 1:2

Connection

## Connection failed

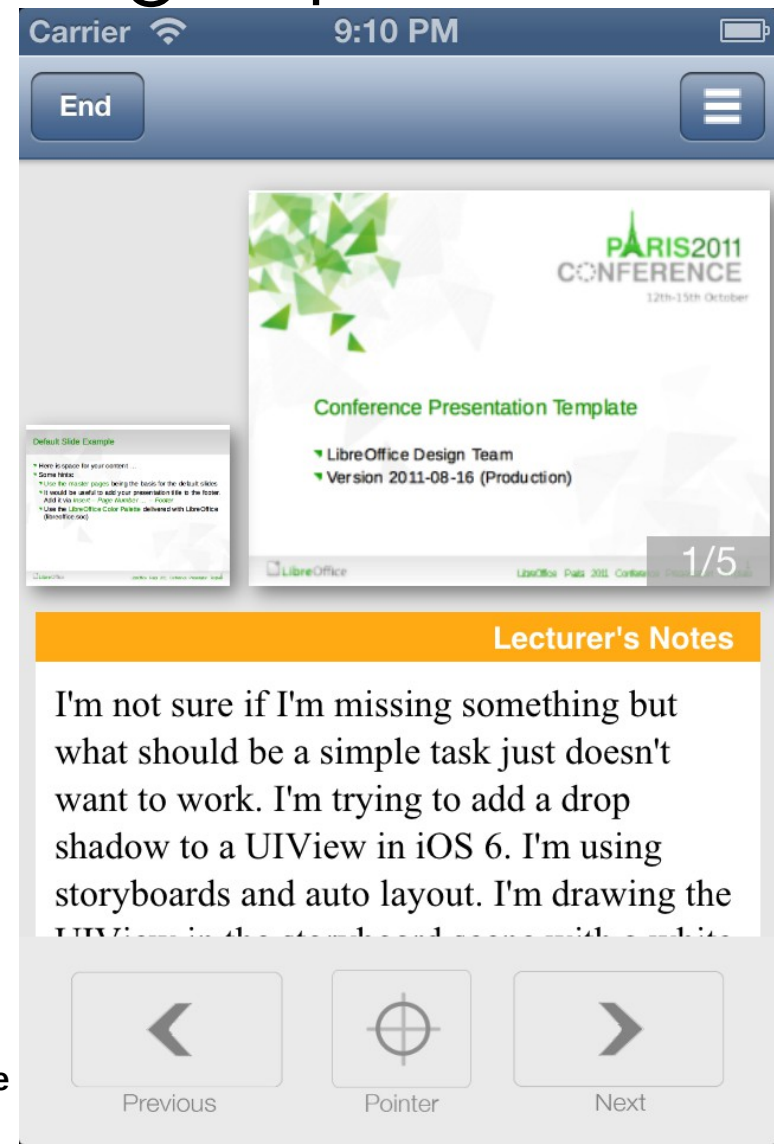
Make sure you enabled remote control. Go to "Tools → Options → LibreOffice Impress → General" in LibreOffice Impress.

You should enable experimental features at "Tools → Options → LibreOffice → Advanced" as well.

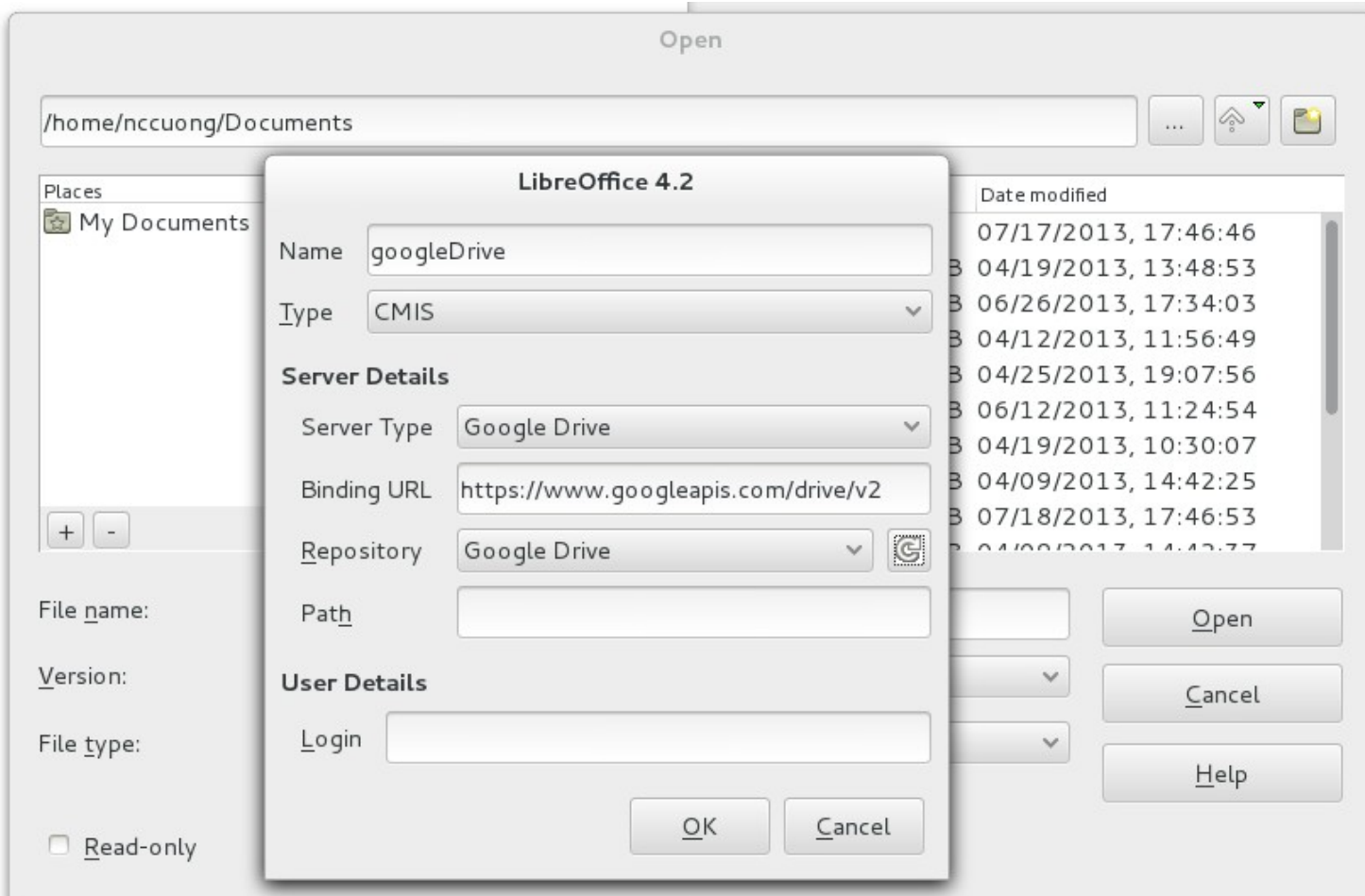
Navigation icons: back, home, recent apps

# Initial iOS Impress remote control

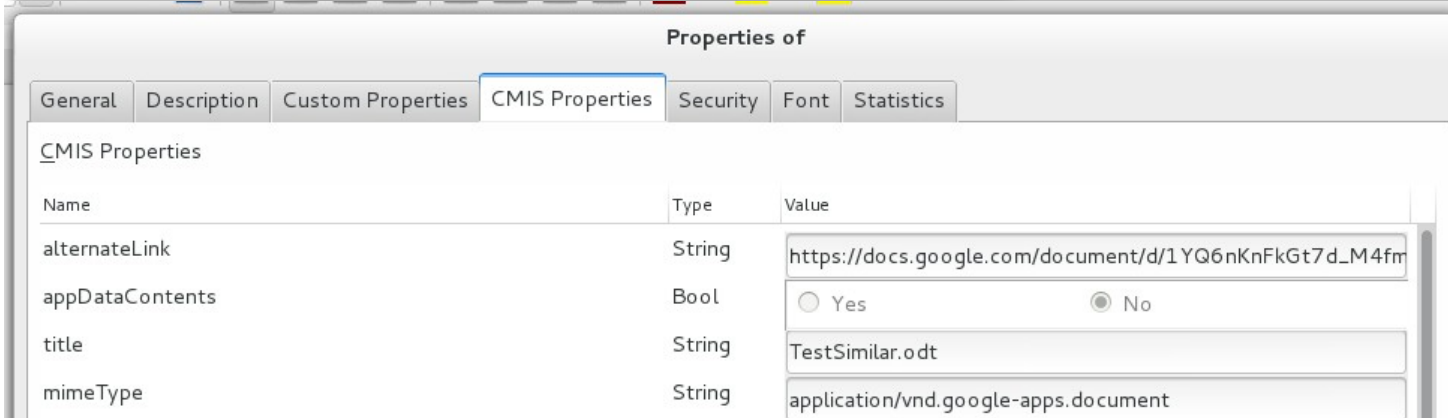
With thanks to *Siqi Liu* – sign up for the Beta



# GDrive integration (via libcmis)

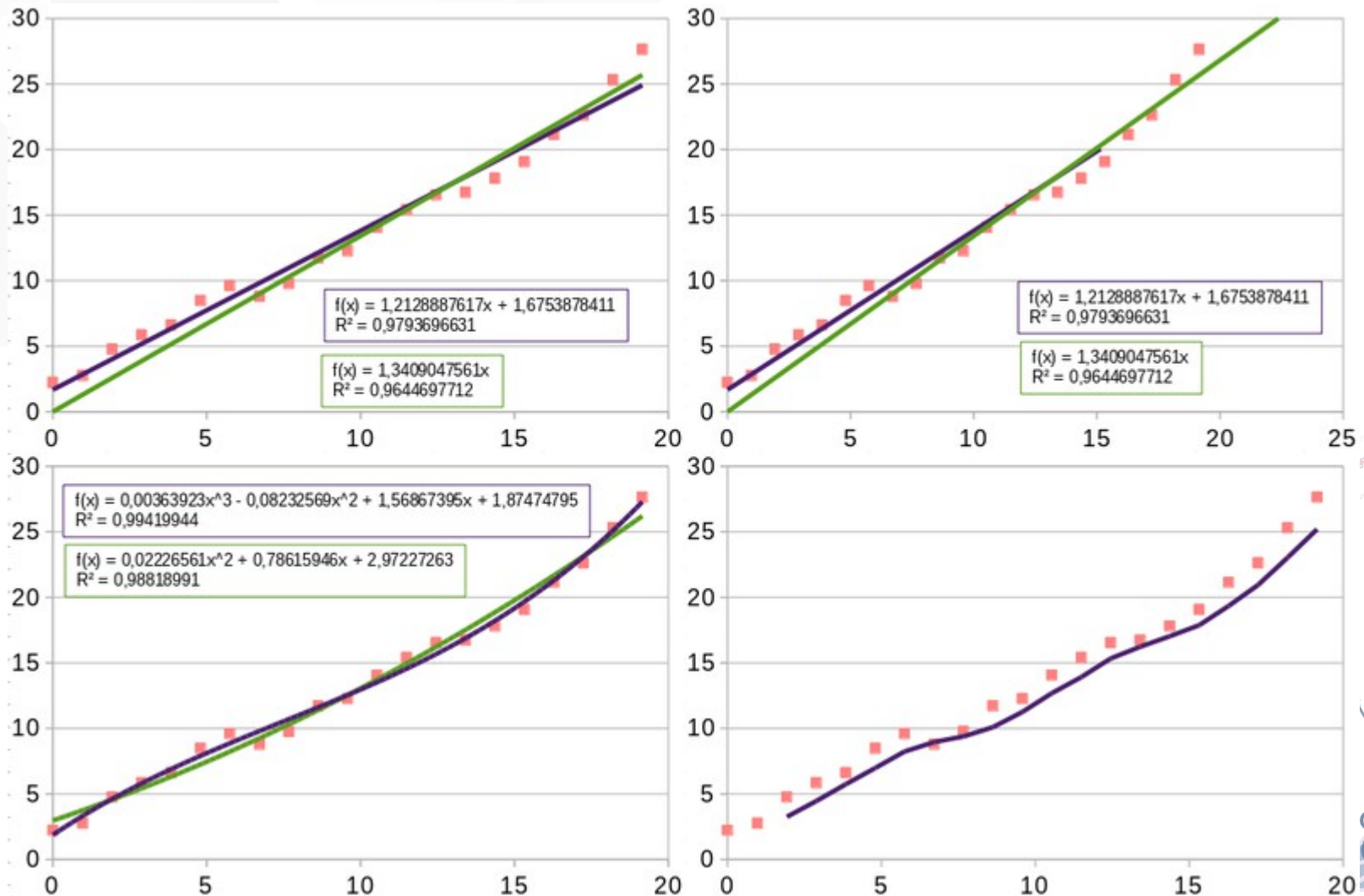


With thanks  
to *Cuong  
Cao Ngo*  
and  
*Cedric  
Bosdonnat*



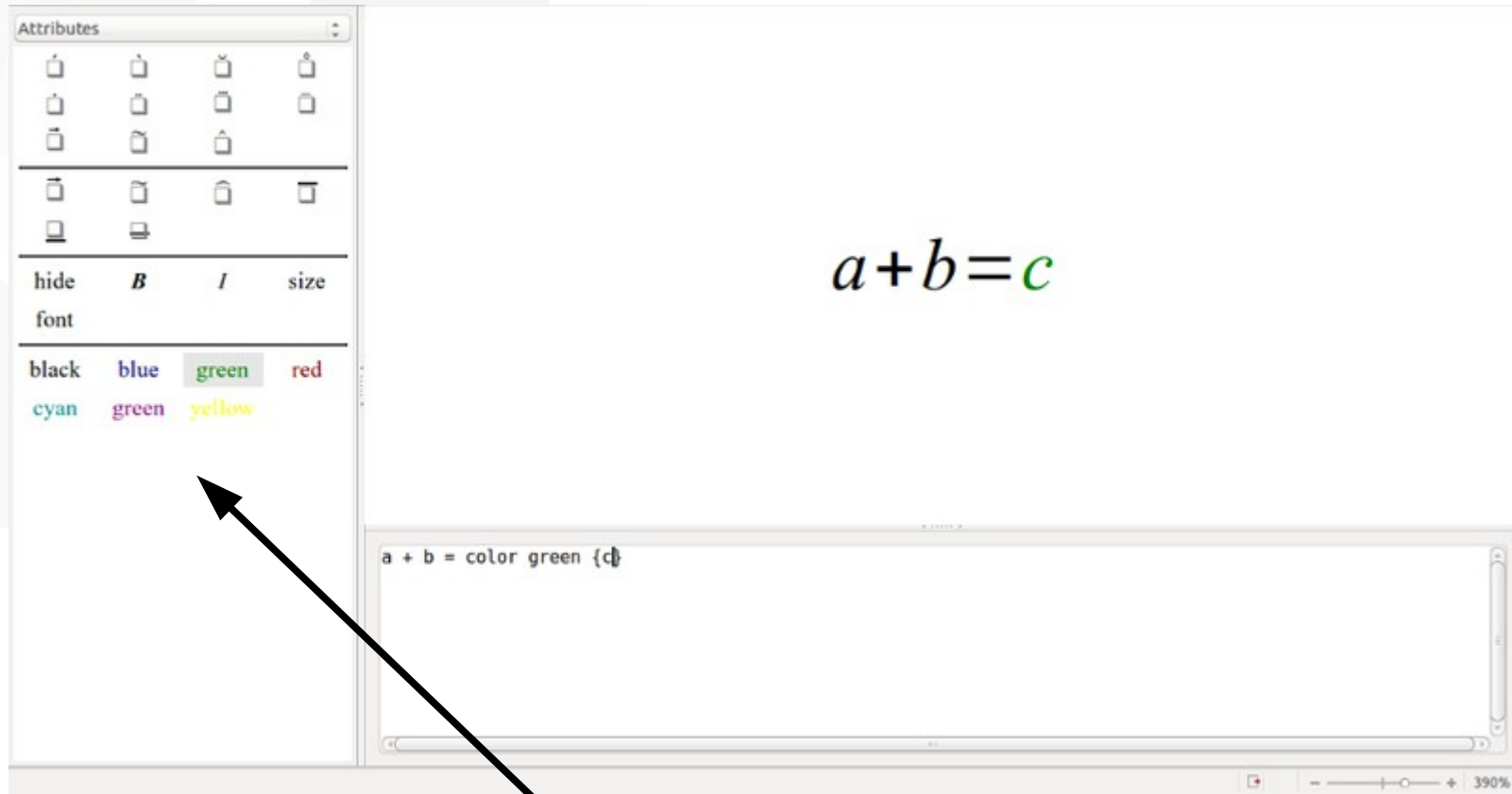
# Chart: new Trend lines

Thanks to *Tomaž Vajngerl*





# New Feature: Math



New Math Panel with colors

Thanks to: *Marcos Paulo de Souza*



# Writer: New Default Template

## Writer

A Word Processor for Every Kind of Document

### Introduction

Writer has all the features you need from a modern, full-featured word processing and desktop publishing tool.

### Features

There are hundreds of features.....

### Wizards

Takes all the hassle out of producing standard documents such as letters, faxes, agendas and minutes, and makes short work of more complex tasks such as mail merges.

Thanks to:

*Ahmad Harthi (KACST)*

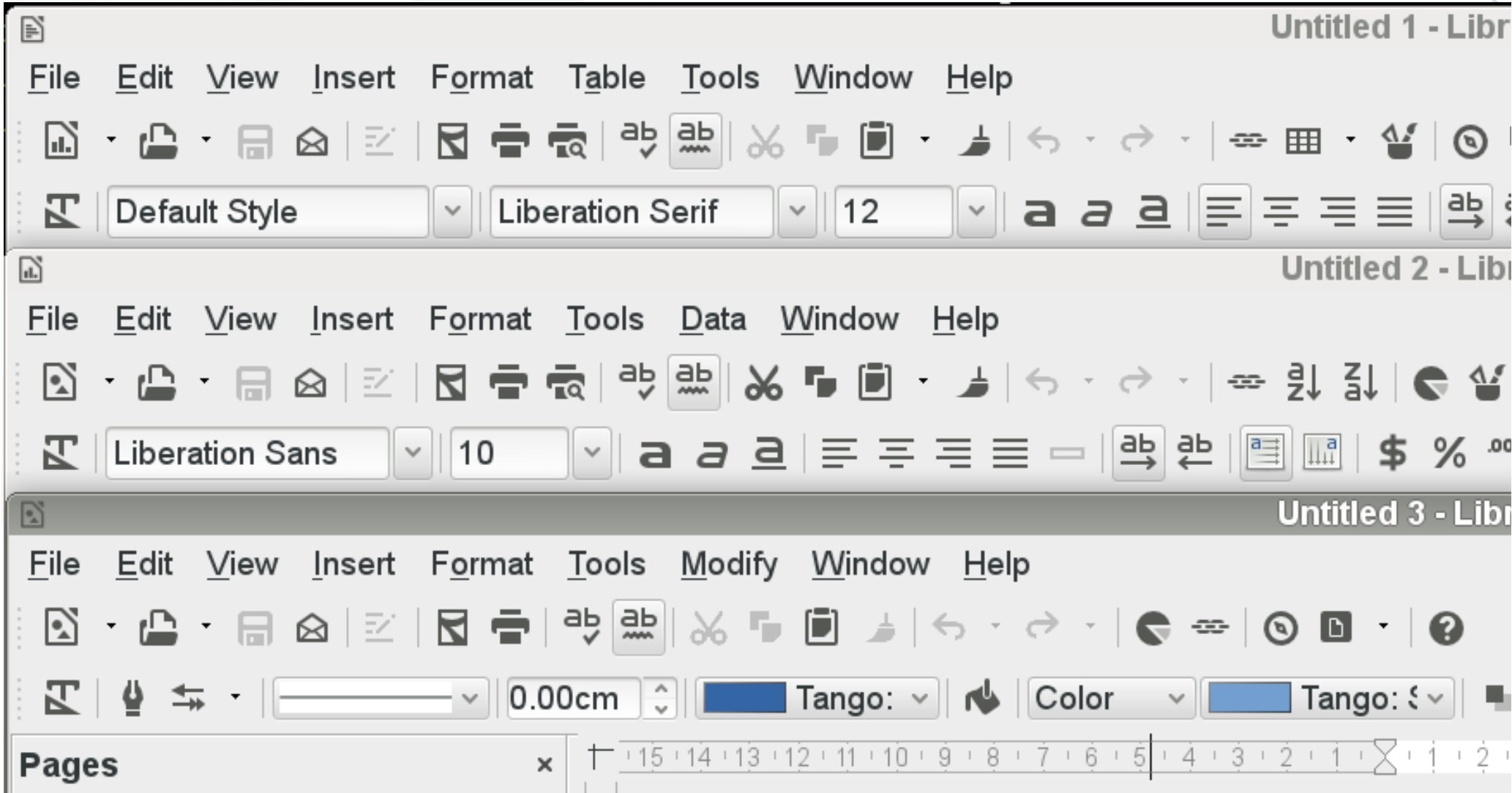
*Faisal M. Al-Otaibi (KACST)*







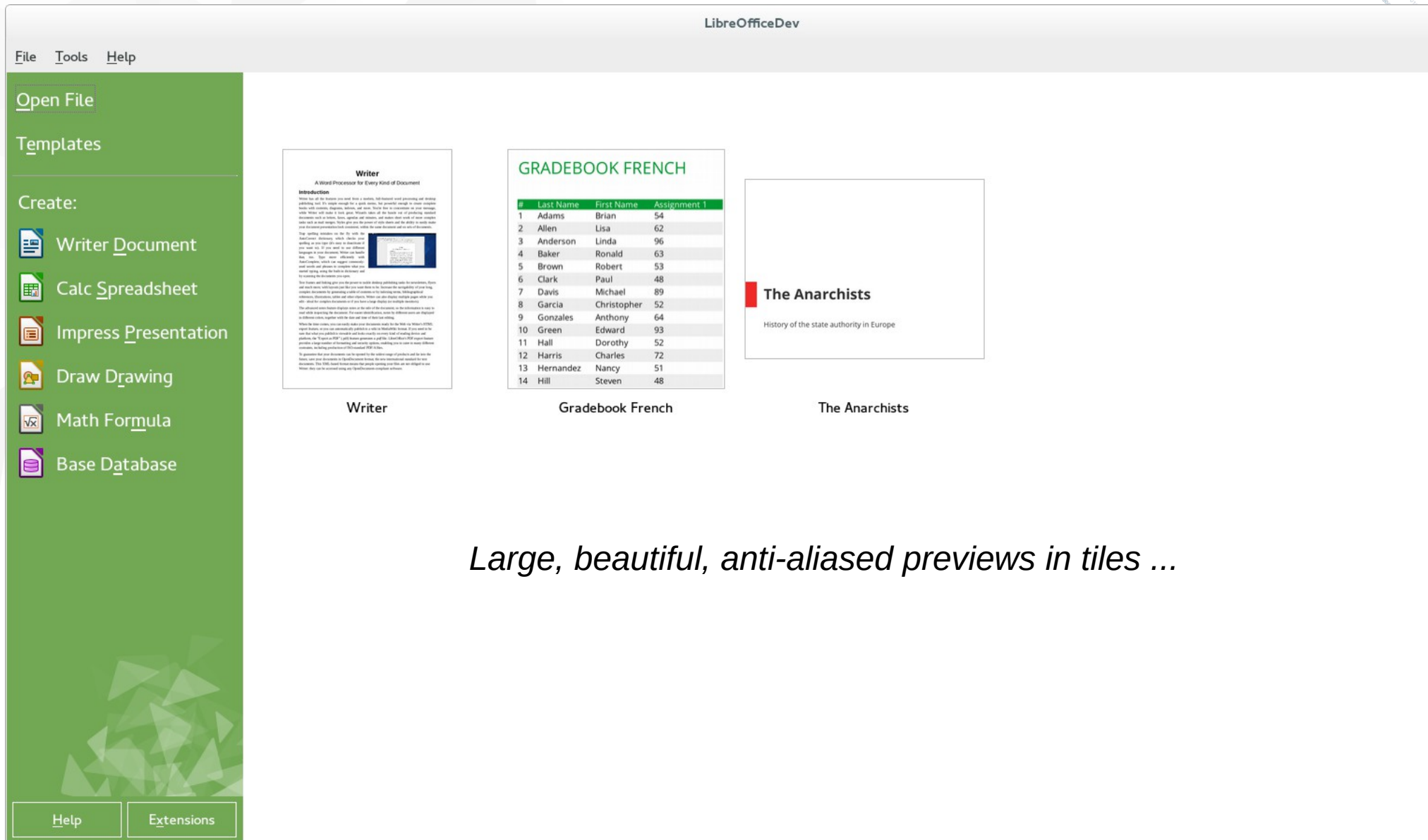
# New Feature: Sifr Icon Set



Thanks to *Issa Alkurtass (KACST)*, *Norah A. Abanumay (KACST)*



# New Feature: Start Screen

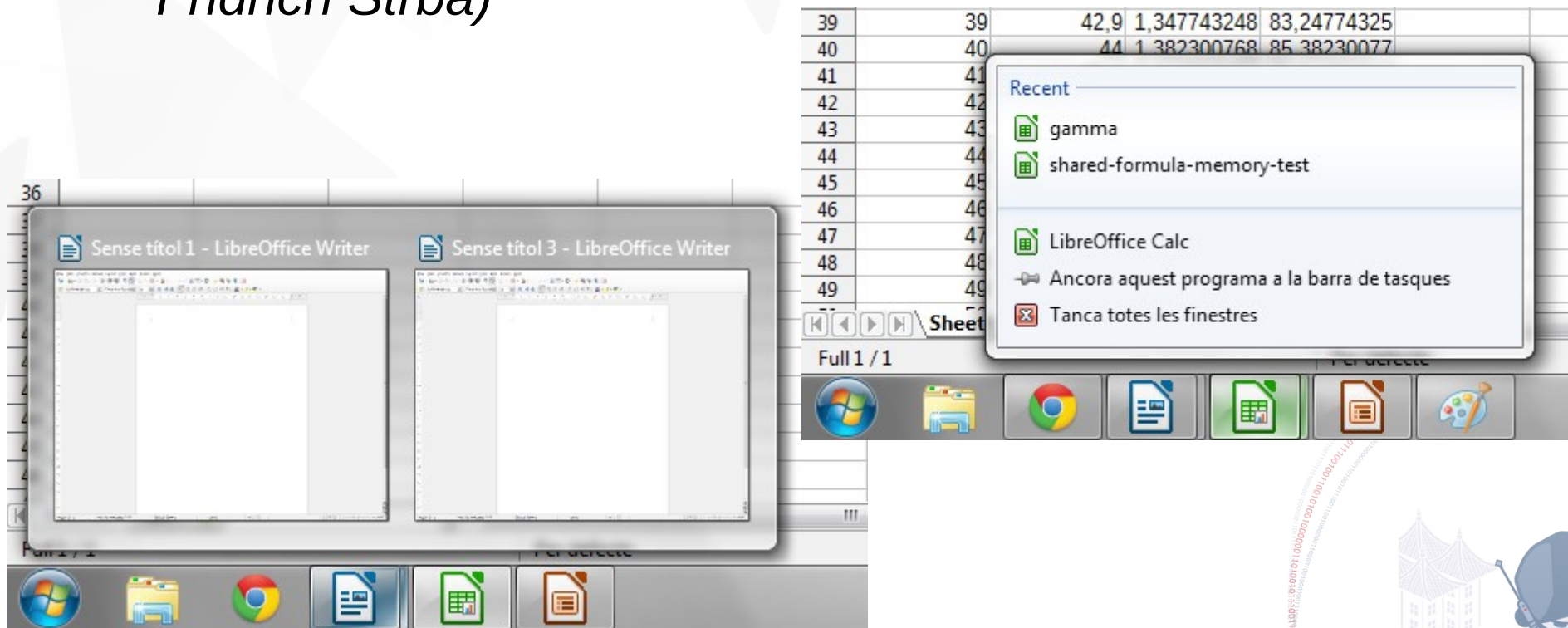


*Large, beautiful, anti-aliased previews in tiles ...*



# 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 ...

What do you want to do?

Create a new database

Embedded Database:

Open an existing database file

Recently used:



# 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*)
  - Acta Mac Classic Document, Beagle Works / WordPerfect Works Document, Great Works Document, MacDoc Document, More Mac v2-3 Document





# AbiWord Import

Thanks to Fridrich Strba

guadec-4.abw

File Edit View Insert Format Tools Table Collaborate Documents Help

Normal Times New Roman 12

## AbiWord 2.0 - The Next Step

(Toward World Domination)  
Martin Sevier and Dominic Lachowicz

### New features in 2.0

AbiWord 2.0<sup>1</sup> 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.

Feature	Imported from	Exported to
Tables	MS Word, RTF, WordPerfect, HTML, OpenOffice.org, DocBook, (other XML formats)	RTF, HTML, Latex, (other XML formats)
Footnotes <sup>2</sup>	RTF, MSWord, WordPerfect	RTF, Latex
Endnotes	RTF, MSWord, WordPerfect	RTF, Latex
Revision Marks <sup>3</sup>		
MailMerge		

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 heights and widths can be adjusted interactively by dragging ruler controls or table lines. The unix build includes a powerful automatic table insert widget which allows tables dimensions to be created interactively. We also have the ability to nest tables to arbitrary depth. This feature is not available in either Word Perfect or Open Office.

Feature	Platform
Automatic Font detection	New to Unix for 2.0
Anti-aliased text	New to Unix for 2.0
Gtk 2.0 GUI elements	New to Unix for 2.0
Gnome 2 integration	The Gnome Intergration features: <ul style="list-style-type: none"> <li>• Drag and drop images</li> <li>• Embed AbiWord in Nautilus</li> <li>• Drag and drop URL's and files into AbiWord</li> </ul>

1) Not all of AbiWord's table features (like nested tables) are available on these products.  
2) Footnotes can be inserted within tables and have a large variety of styles.  
3) Hopefully we'll have filters to and from at least RTF for Revision Marks before 2.0

Page 1 / 15 3618 words, 22208 characters Page Style 1 English (UK)

guadec-4.abw - LibreOfficeDev Writer

File Edit View Insert Format Table Tools Window Help

Default Style Luxi Sans 17

## AbiWord 2.0 - The Next Step

(Toward World Domination)  
Martin Sevier and Dominic Lachowicz

### New features in 2.0

AbiWord 2.0<sup>1</sup> 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.

Feature	Imported from	Exported to
Tables	MS Word, RTF, WordPerfect, HTML, OpenOffice.org, DocBook, (other XML formats)	RTF, HTML, Latex, DocBook, (other XML formats)
Footnotes <sup>2</sup>	RTF, MSWord, WordPerfect	RTF, Latex
Endnotes	RTF, MSWord, WordPerfect	RTF, Latex
Revision Marks <sup>3</sup>		
MailMerge		

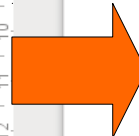
The Table feature in AbiWord is very powerful. Cells within tables can be merged either horizontally or vertically via an easy-to-use non-modal dialog. Rows and Columns heights and widths can be adjusted interactively by dragging ruler controls or table lines. The unix build includes a powerful automatic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

Feature	Platform
Automatic Font detection	New to Unix for 2.0
Anti-aliased text	New to Unix for 2.0

1) Not all of AbiWord's table features (like nested tables) are available on these products.  
2) Footnotes can be inserted within tables and have a large variety of styles.  
3) Hopefully we'll have filters to and from at least RTF for Revision Marks before 2.0

Presented at GUADEC 4, Dublin, Ireland, 2003

Page 1 / 15 3618 words, 22208 characters Page Style 1 English (UK)



# Initial Keynote Filter



preview.key - LibreOfficeDev Impress 4.2 [5eb37896ca344276cc0753ad93b822d7be9b21d6]

File Edit View Insert Format Tools Slide Show Window Help

0.00" Tango: Color

Slides

Normal Outline Notes Handout Slide Sorter


1 **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" idea :-)

Samples:

I  LibreOffice

hello

2

Properties

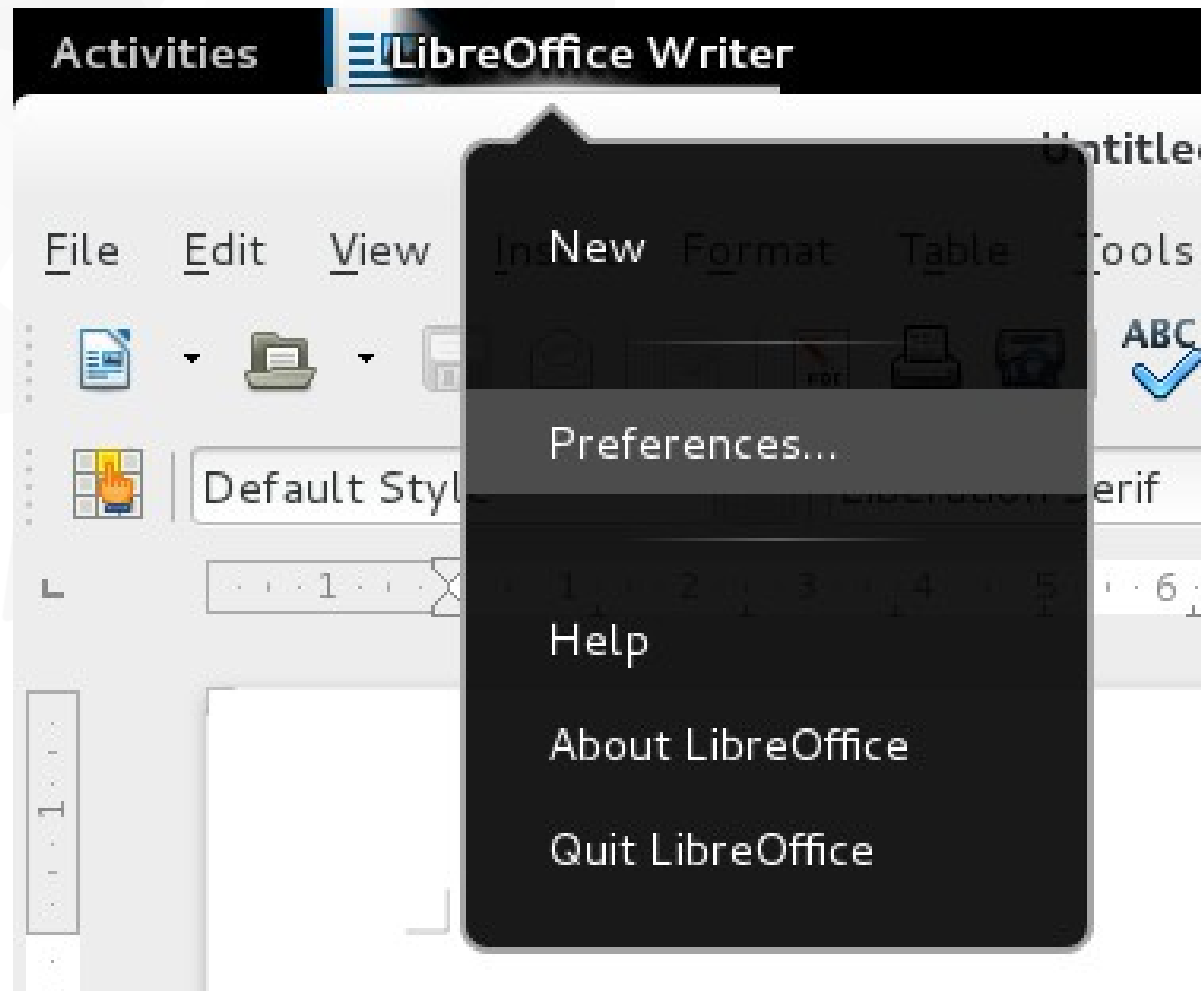
Layouts

Thanks to David Tardon (RedHat)

1.65 / 5.05 0.00 x 0.00 Slide 1 / 2 Default 46%

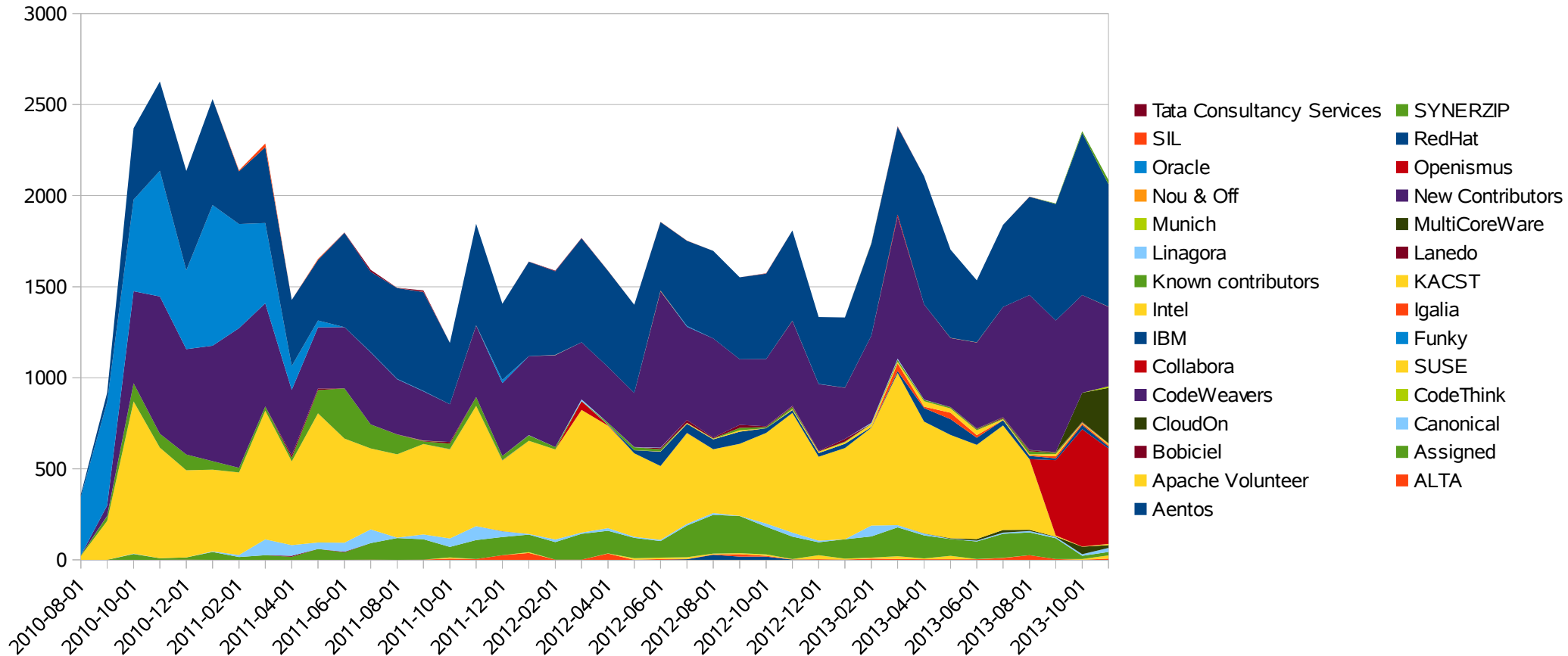


# GNOME 3.0 Menu



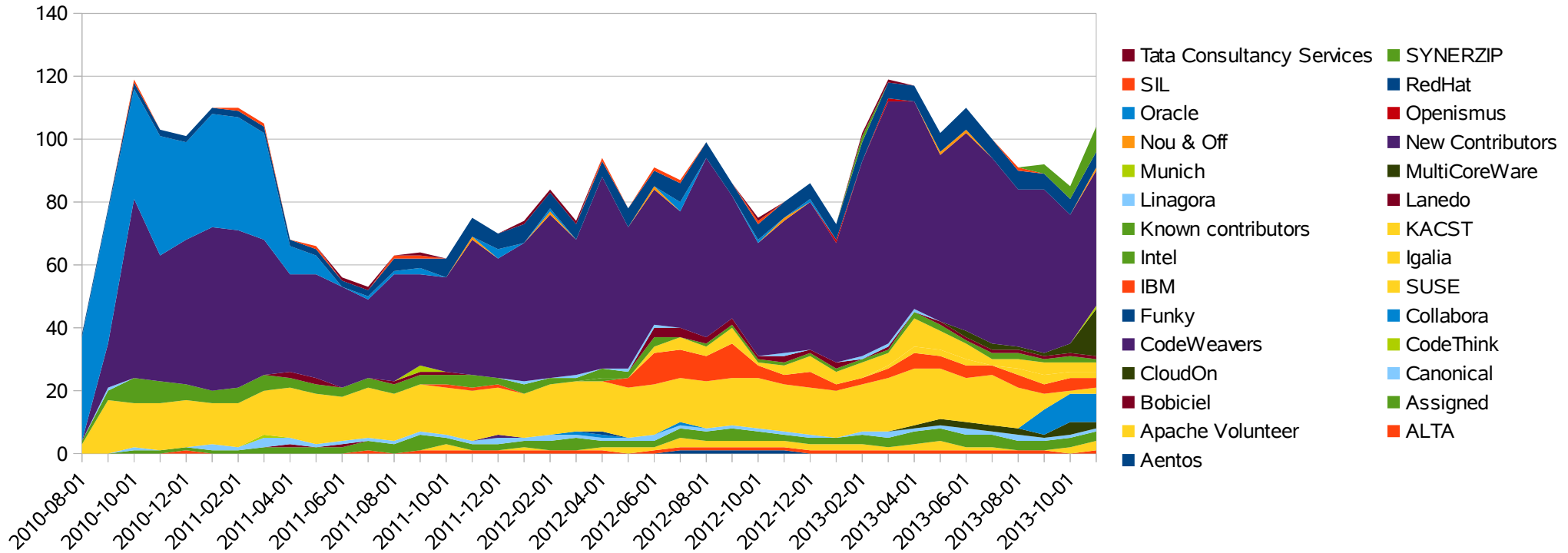
# Commits per month

Code commits per month by affiliation



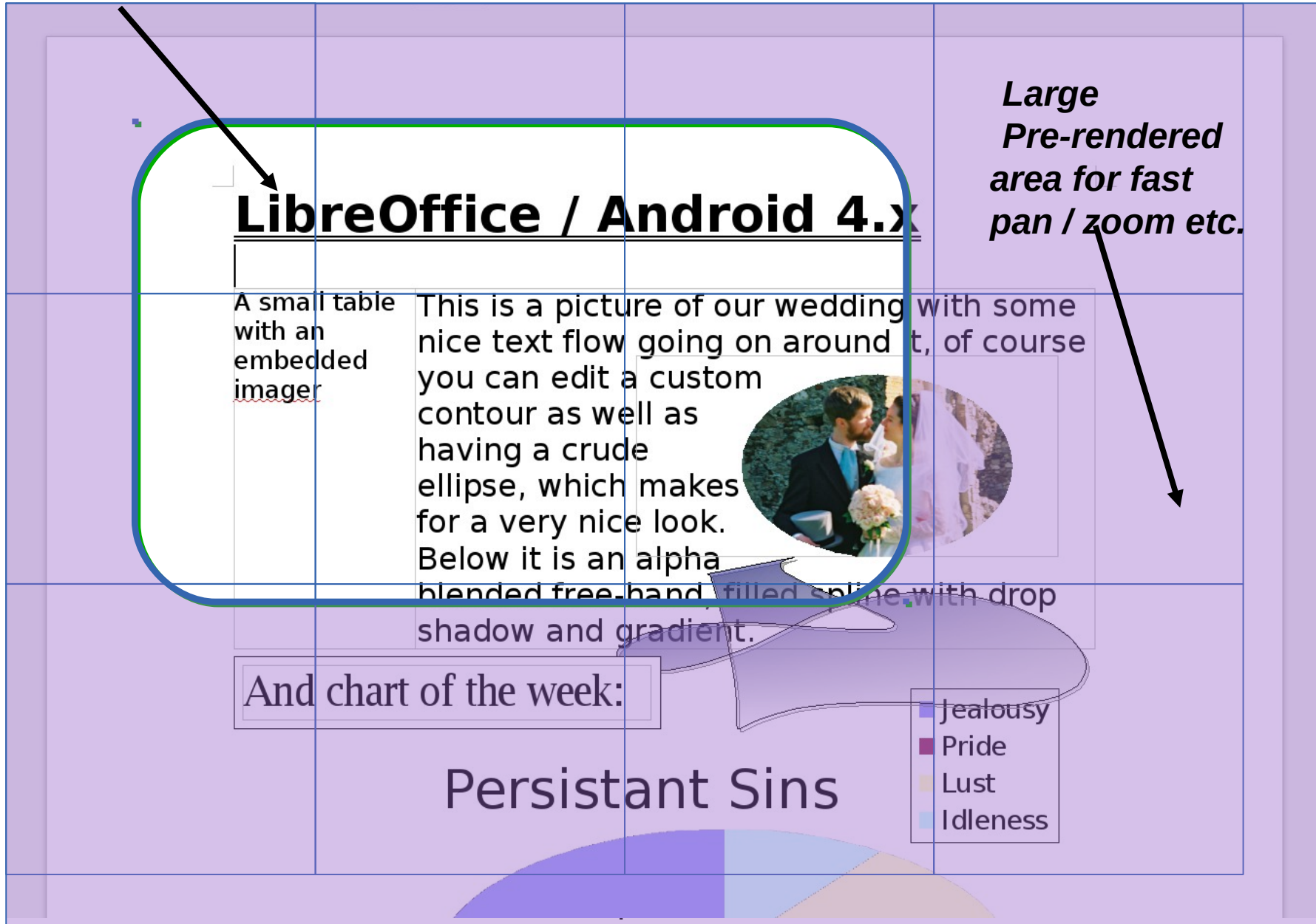
# Active developers per month

Active developers each month by affiliation



# Fast Tiled Rendering in textures

*Tablet display*



# Misc. Project Bits Recently

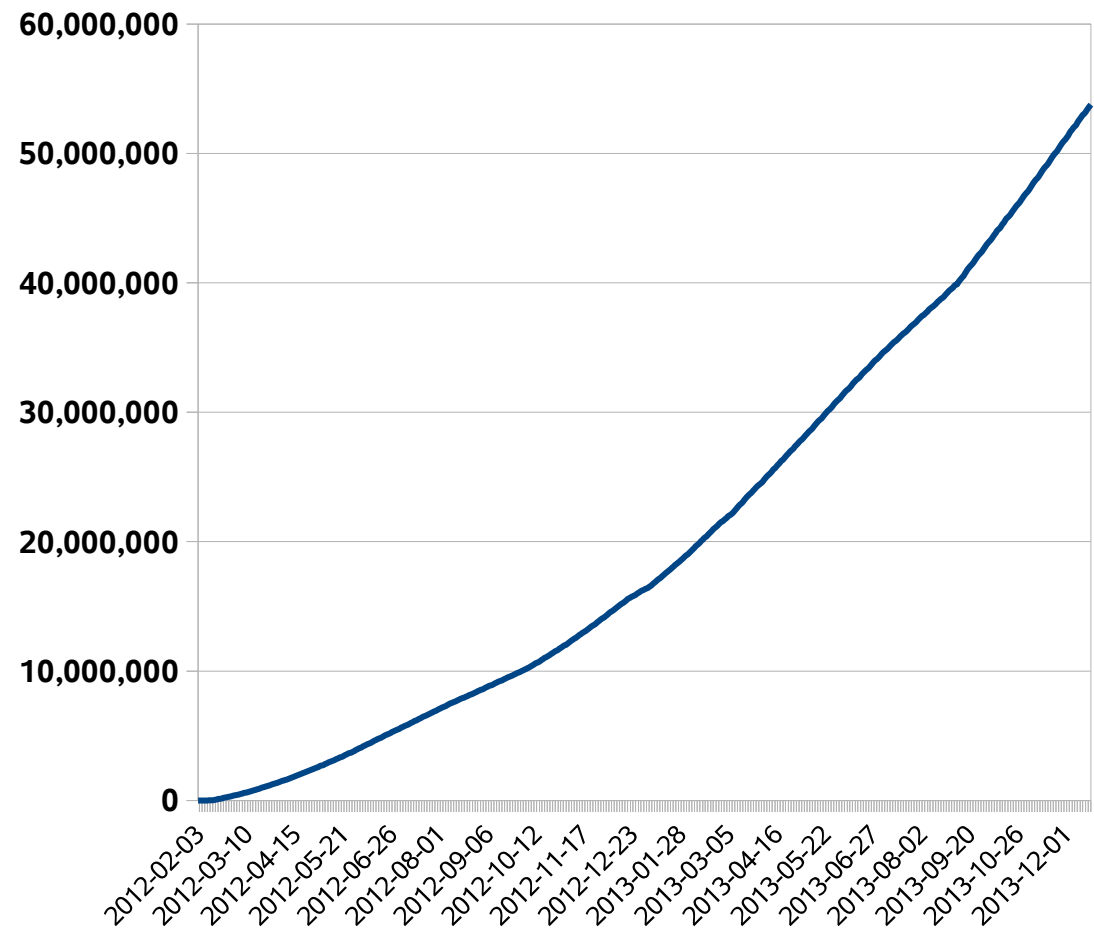


# LibreOffice Project & Software

- 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/>

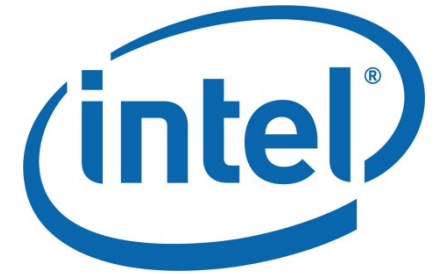
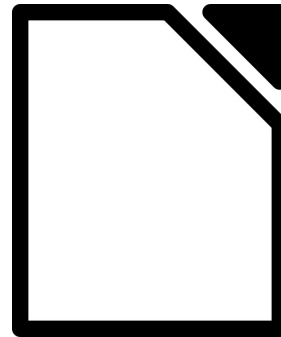
Cumulative unique IP's for updates vs. time

not counting any Linux / vendor versions



# Advisory Board Members

*This slide's layout is a victim of our success here ...*



مدينة الملك عبد العزيز  
للعلوم و التقنية KACST







# 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*

