JogAmp Fast Media & Processing

regardless whether the JVM is slow or not.

FOSDEM 2013 – Universite Libre de Bruxelles

February 3, 2013

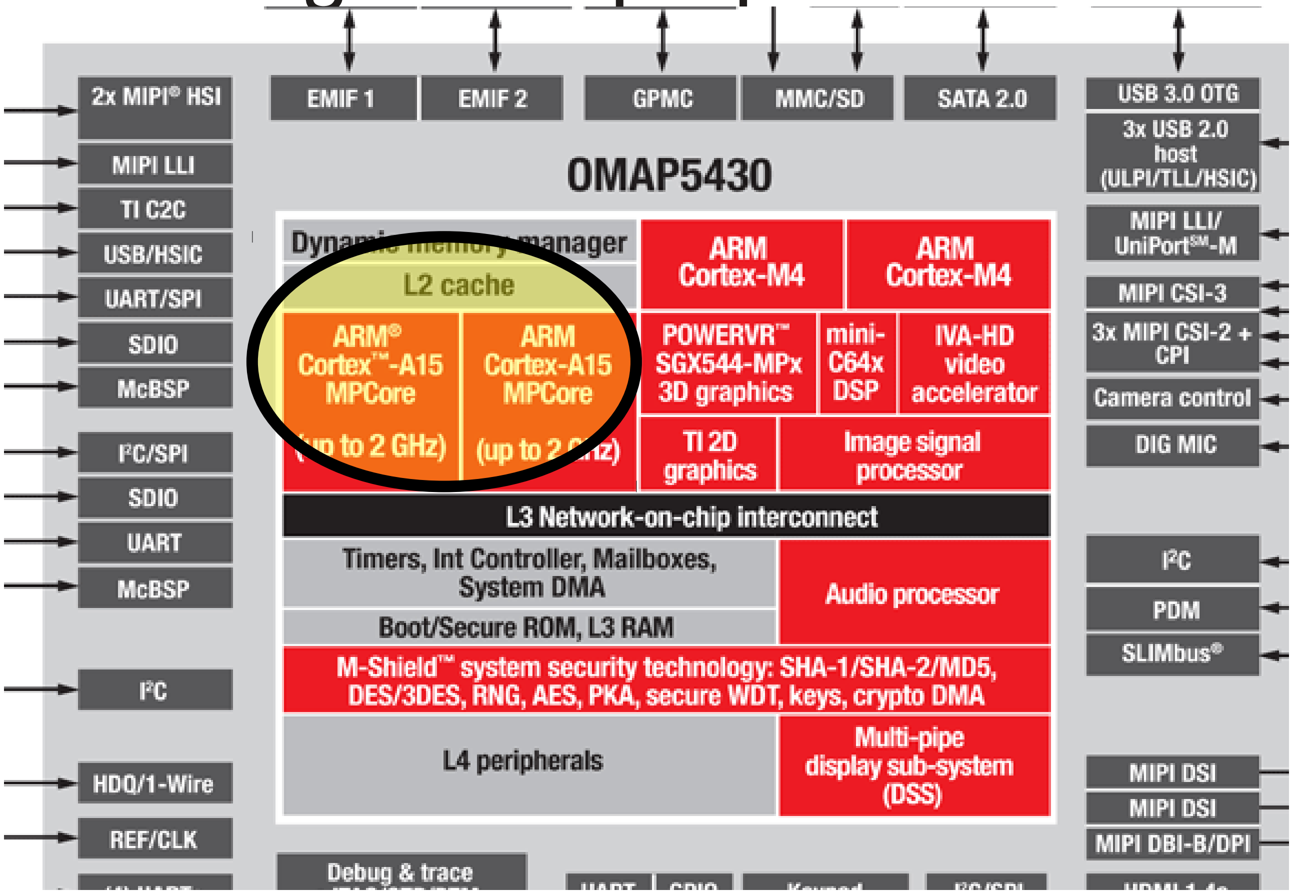
Presented by: Sven Gothel
Xerxes Ranby
Julien Gouesse

A Love Story ...

- .. whole project Lifecycle
- .. love across devices and boundaries
- .. free love
  - As in Beer
  - As in Freedom – Accessibility == Speech

General Love

- Agnostic CPU for general purpose

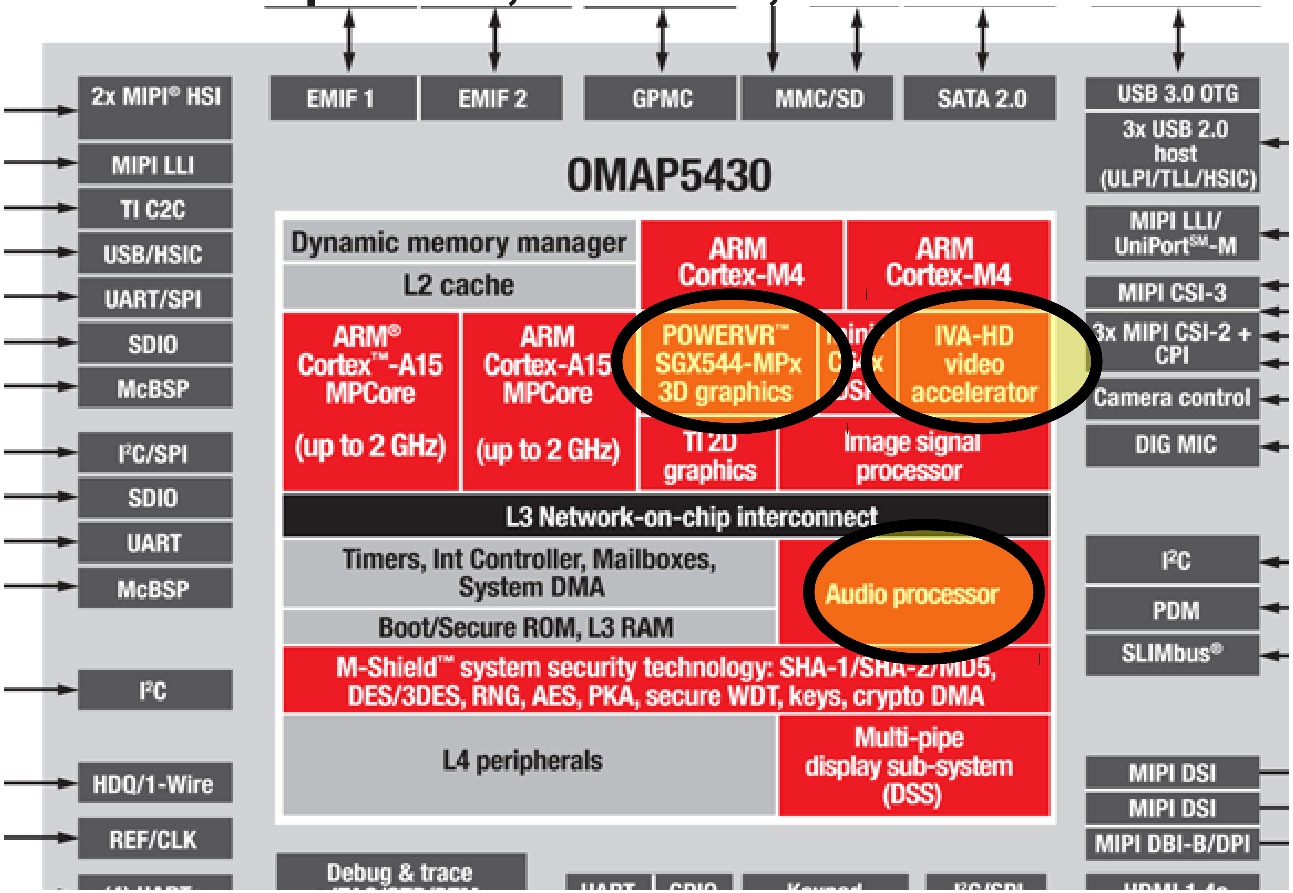


General Love

- Intel x32, amd64
- ARM 32 and 64 bit
- SuperH
- PowerPC
- Sparc
- MIPS

Dedicated Love

- **Fast** Discrete Graphics, Audio, Video ...



OMAP5430

2x MIPI® HSI

MIPI LLI

TI C2C

USB/HSIC

UART/SPI

SDIO

McBSP

I²C/SPI

SDIO

UART

McBSP

I²C

HDQ/1-Wire

REF/CLK

EMIF 1

EMIF 2

GPMC

MMC/SD

SATA 2.0

USB 3.0 OTG

3x USB 2.0 host (ULPI/TLL/HSIC)

MIPI LLI/ UniPort℠-M

MIPI CSI-3

3x MIPI CSI-2 + CPI

Camera control

DIG MIC

I²C

PDM

SLIMbus®

MIPI DSI

MIPI DSI

MIPI DBI-B/DPI

Dynamic memory manager

L2 cache

ARM Cortex-M4

ARM Cortex-M4

ARM® Cortex™-A15 MPCore (up to 2 GHz)

ARM Cortex-A15 MPCore (up to 2 GHz)

POWERVR™ SGX544-MPx 3D graphics

IVA-HD video accelerator

TI 2D graphics

Image signal processor

L3 Network-on-chip interconnect

Timers, Int Controller, Mailboxes, System DMA

Boot/Secure ROM, L3 RAM

Audio processor

M-Shield™ system security technology: SHA-1/SHA-2/MD5, DES/3DES, RNG, AES, PKA, secure WDT, keys, crypto DMA

L4 peripherals

Multi-pipe display sub-system (DSS)

Debug & trace

Dedicated Love

- Immaginative Technologies PowerVR SGX
- ARM Mali
- Qualcomm Andreno
- Vivante
- BCM VideoCore
- Nvidia Tegra
- Special DSPs (Audio/Video)
- Heterogenous CPUs (big.LITTLE)

Which to choose ?

- General Purpose CPU
  - PRO: Homogeneous Code / Easy to use
  - CON: Not fast for all Problems
- Dedicated Silicon
  - PRO: **FAST**
  - CON: Customized code, BLOBS, incompatible instruction set

Love them all!

- General Purpose CPU
  - CON: Not fast for all Problems
  - REMEDY: Use dedicated solution below
- Dedicated Silicon
  - CON: Customized code, BLOBS, incompatible instruction set
  - REMEDY: Open Interfaces (API & ABI)
    - OpenGL, OpenCL, OpenAL, OpenMAX, ...

Practical Love

- Pick something that works!
- Reliability / Transparency
  - Communication
  - Bugtracker
  - Forum

Community

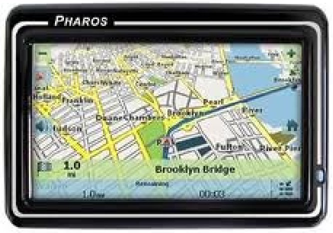
Practical Love

- JOGAMP makes the ***bare metal*** loveable
  - No restrictions to original API
  - Platform neutrality
  - Hard to use for domain specific problems
    - CAD, Games, ..
- Domain Specific APIs close the Circle
  - SciLab, Java3D, Ardor3D, jME3, libGDX ..
  - Specialized Solutions

Love in all shapes & Colors ..

JogAmp

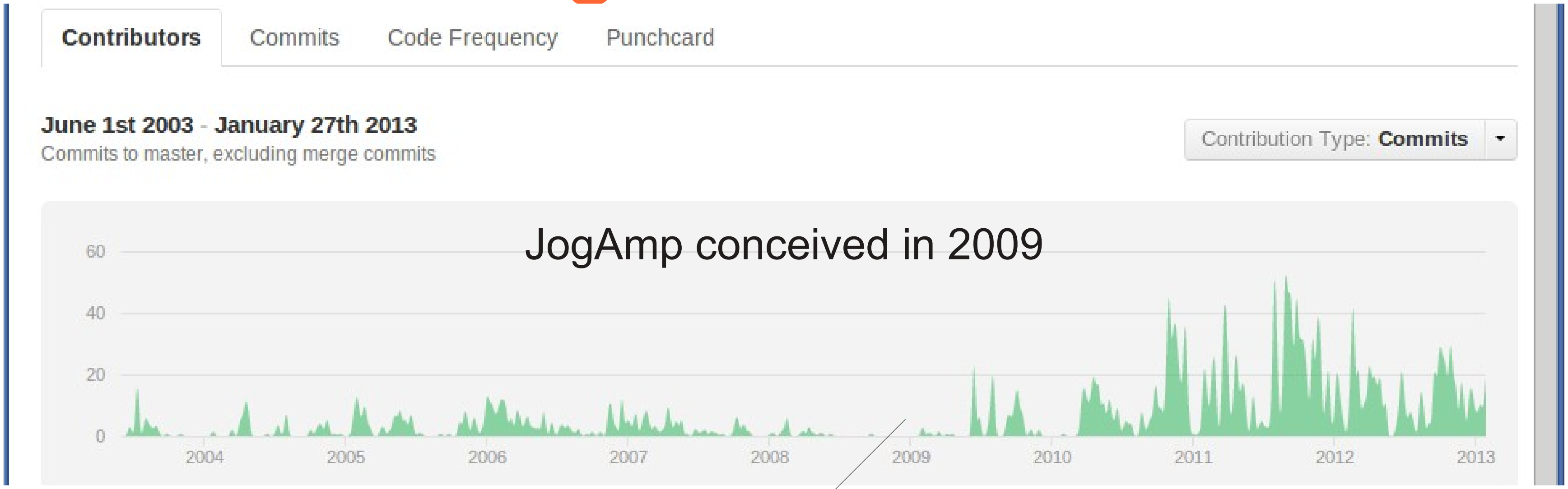
JOGL – JOCL – JOAL ...



About US

- Open & Vendor Independent
- BSD License
- Java Graphics, Audio, Media & Processing High Performance Bindings
- One Stop Community Platform
  - SCM, Bugtracking, Build Server, Mailinglist/Forum,..
- Commercial Support
- http://jogamp.org

Progressive Love

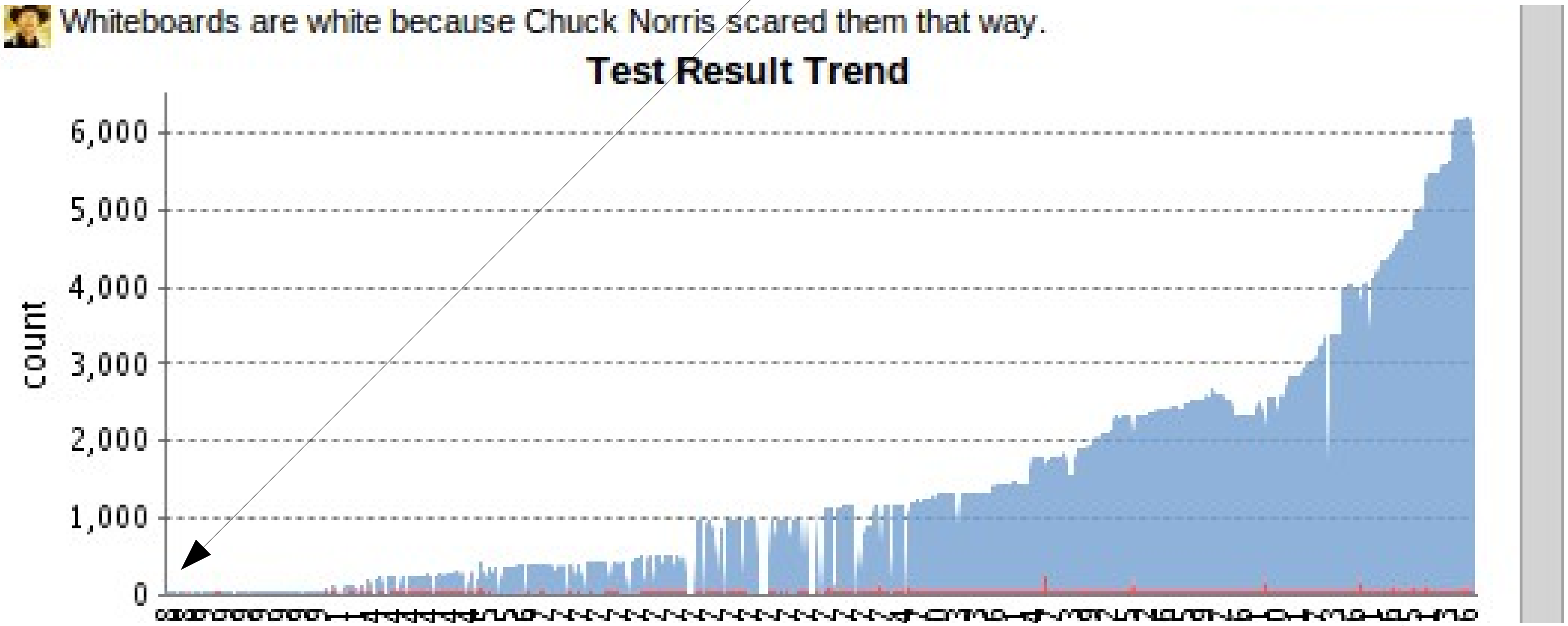


Contributors | Commits | Code Frequency | Punchcard

June 1st 2003 - January 27th 2013
Commits to master, excluding merge commits

Contribution Type: **Commits**

JogAmp conceived in 2009



Whiteboards are white because Chuck Norris scared them that way.

Test Result Trend

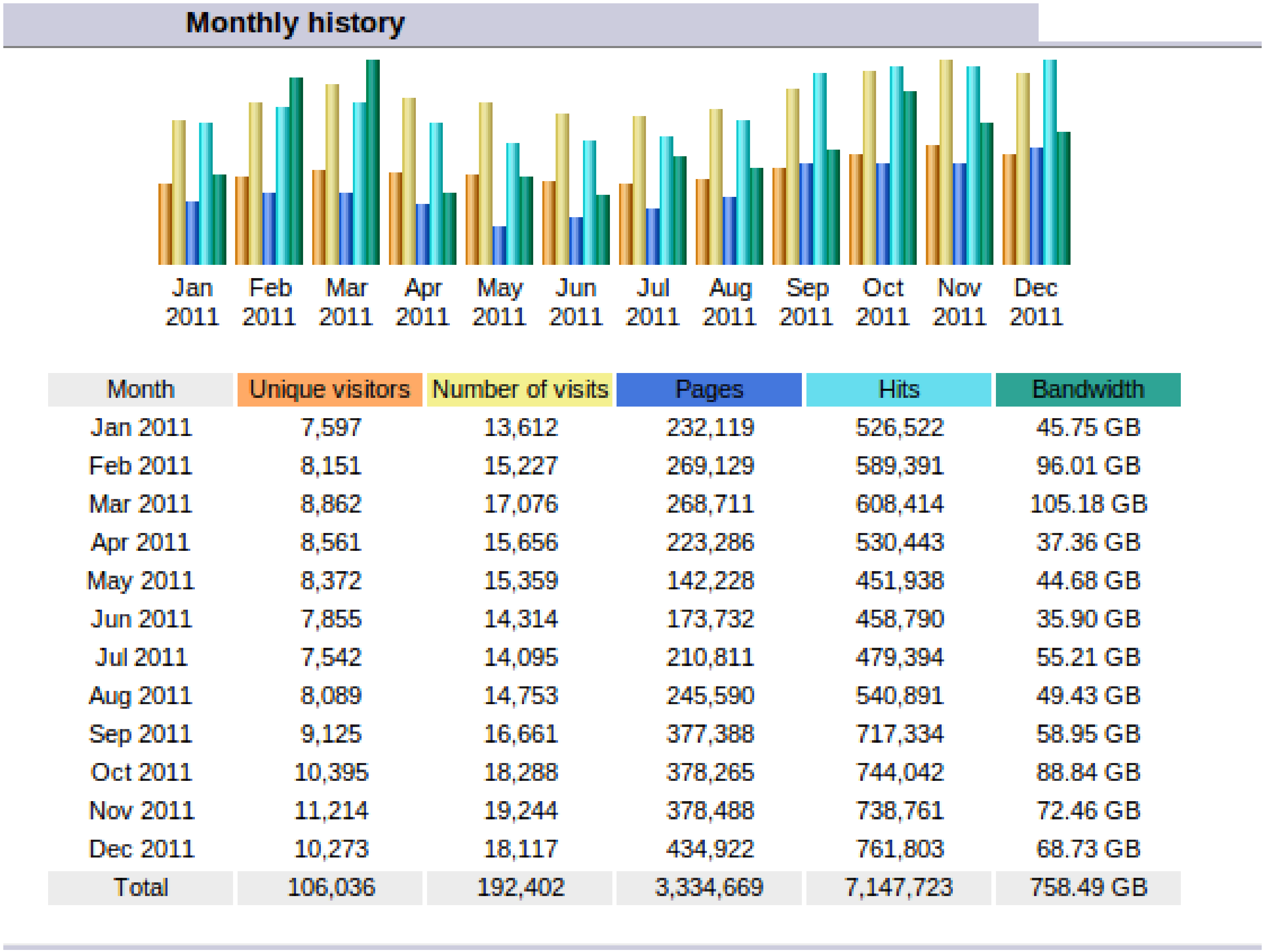
JogAmp Continuity / Maturity

- Maturity
  - Version 1
    - JSR-231
  - Version 2
    - OpenGL Profiles (ES 1+2, GL 2 + 3 + 4)
    - Windowing Toolkit Abstraction
    - Continuity Build/Test Server http://jogamp.org/chuck/
    - 90 GlueGen + 529 JOGL Unit Tests
- Community Effort
  - Ports [FreeBSD, ARM-HF, ..]
  - Engine & Device Support
  - Bug Entries, Test Cases & Fixes
  - Code Reviews, Communication & General Help

JogAmp Deployment

- Preinstalled Bundles
  - Modularized JARs
  - Android APKs (modular, or all-in-one)
  - Maven
- Online / Cached
  - Automatic Native-JAR loading support
  - Applet
    - Classical
    - JNLP
  - Webstart (JNLP)

.. and Voyeurism - 2011

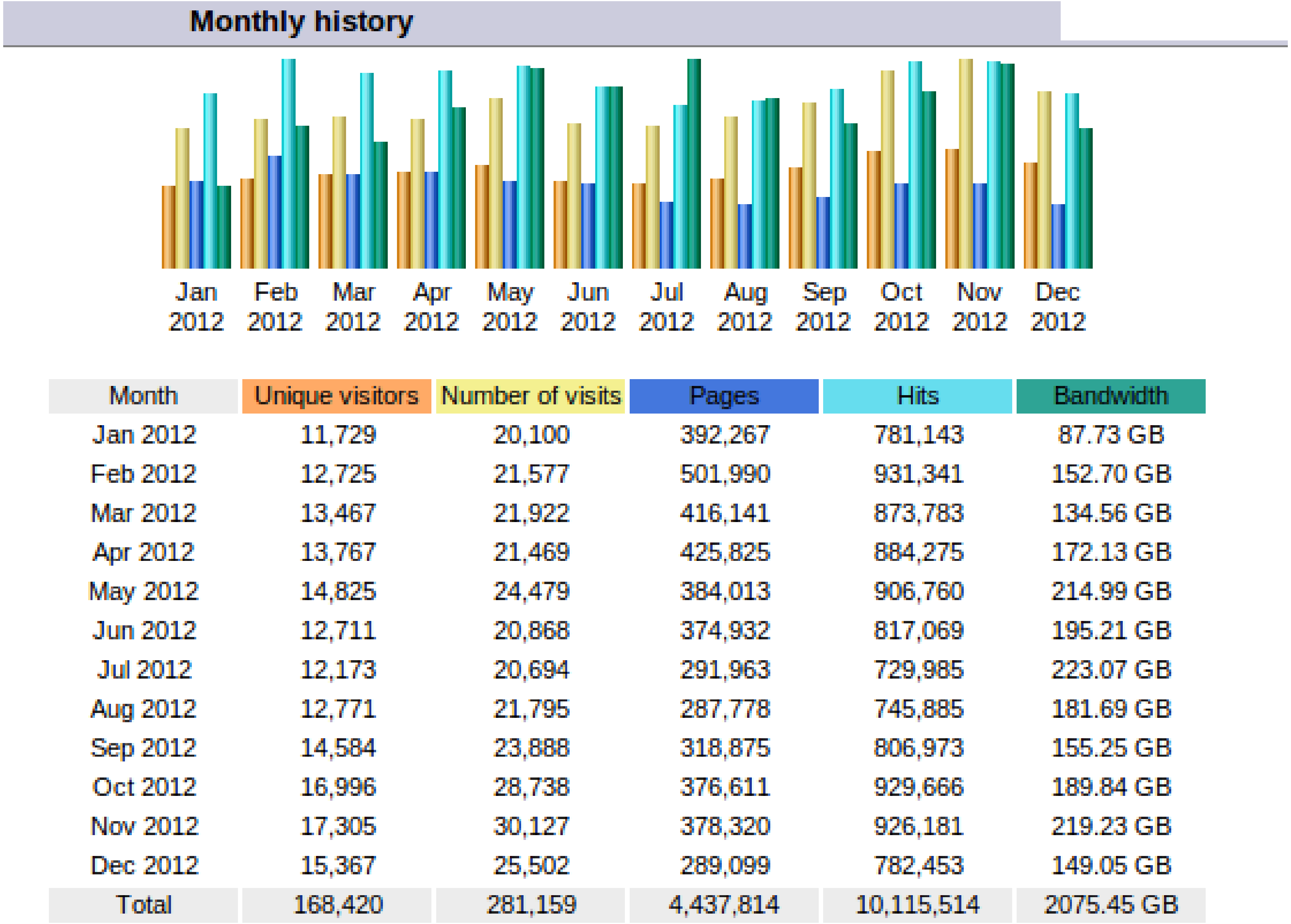


Monthly history

Month	Unique visitors	Number of visits	Pages	Hits	Bandwidth
Jan 2011	7,597	13,612	232,119	526,522	45.75 GB
Feb 2011	8,151	15,227	269,129	589,391	96.01 GB
Mar 2011	8,862	17,076	268,711	608,414	105.18 GB
Apr 2011	8,561	15,656	223,286	530,443	37.36 GB
May 2011	8,372	15,359	142,228	451,938	44.68 GB
Jun 2011	7,855	14,314	173,732	458,790	35.90 GB
Jul 2011	7,542	14,095	210,811	479,394	55.21 GB
Aug 2011	8,089	14,753	245,590	540,891	49.43 GB
Sep 2011	9,125	16,661	377,388	717,334	58.95 GB
Oct 2011	10,395	18,288	378,265	744,042	88.84 GB
Nov 2011	11,214	19,244	378,488	738,761	72.46 GB
Dec 2011	10,273	18,117	434,922	761,803	68.73 GB
Total	106,036	192,402	3,334,669	7,147,723	758.49 GB

FOSDEM '13

.. and Voyeurism - 2012



Monthly history

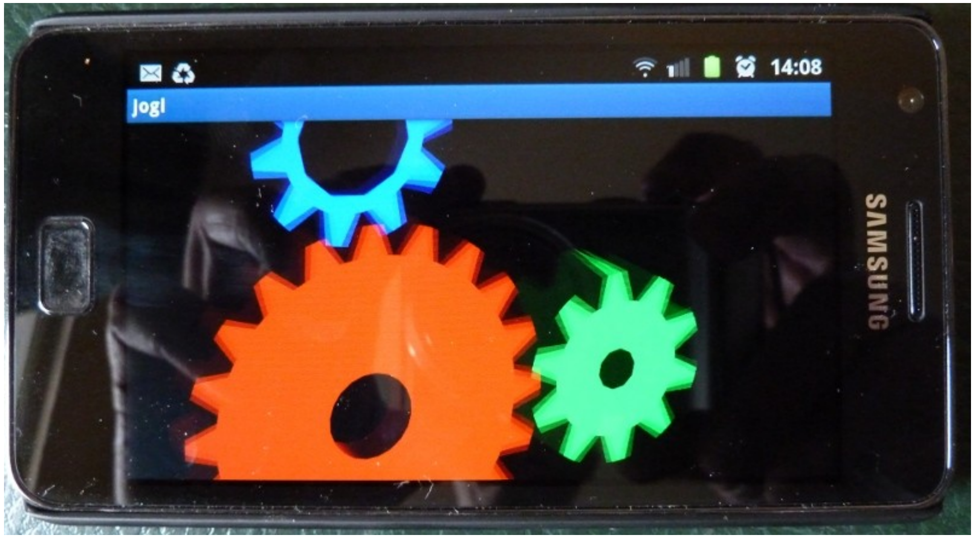
Month	Unique visitors	Number of visits	Pages	Hits	Bandwidth
Jan 2012	11,729	20,100	392,267	781,143	87.73 GB
Feb 2012	12,725	21,577	501,990	931,341	152.70 GB
Mar 2012	13,467	21,922	416,141	873,783	134.56 GB
Apr 2012	13,767	21,469	425,825	884,275	172.13 GB
May 2012	14,825	24,479	384,013	906,760	214.99 GB
Jun 2012	12,711	20,868	374,932	817,069	195.21 GB
Jul 2012	12,173	20,694	291,963	729,985	223.07 GB
Aug 2012	12,771	21,795	287,778	745,885	181.69 GB
Sep 2012	14,584	23,888	318,875	806,973	155.25 GB
Oct 2012	16,996	28,738	376,611	929,666	189.84 GB
Nov 2012	17,305	30,127	378,320	926,181	219.23 GB
Dec 2012	15,367	25,502	289,099	782,453	149.05 GB
Total	168,420	281,159	4,437,814	10,115,514	2075.45 GB

DEMOS

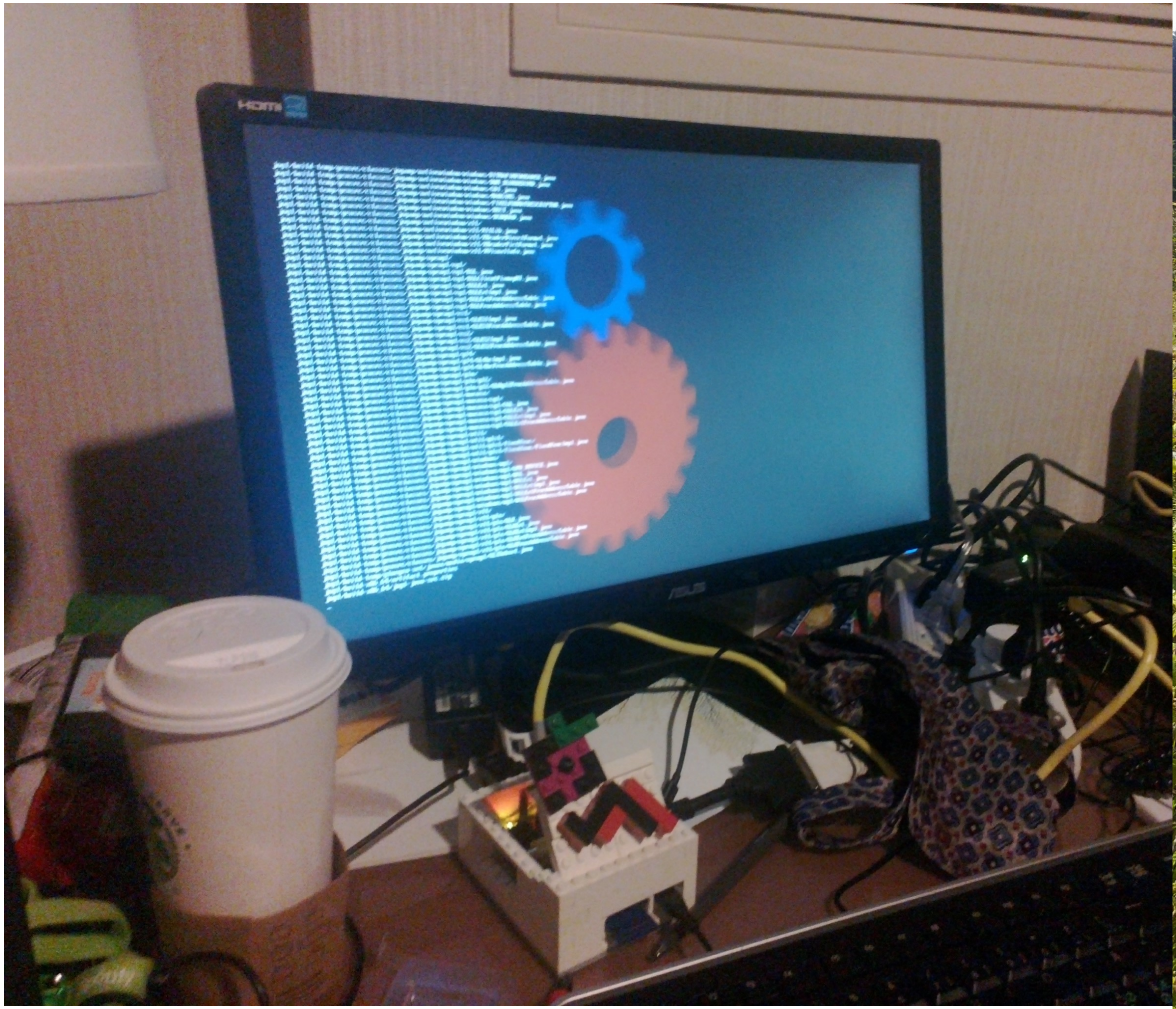
- JOGL/JOCL
  - Desktop (Any)
  - Android (Any)
  - AC 100 (tegra2)
- Jake2
  - AC100 (tegra2)
  - MeeGo (PowerVR SGX)
- LibGDX
  - Raspberry Pi (BCM)
  - AC 100 (tegra2)
- JME3
  - Desktop (Any)

JOGL Android Binding

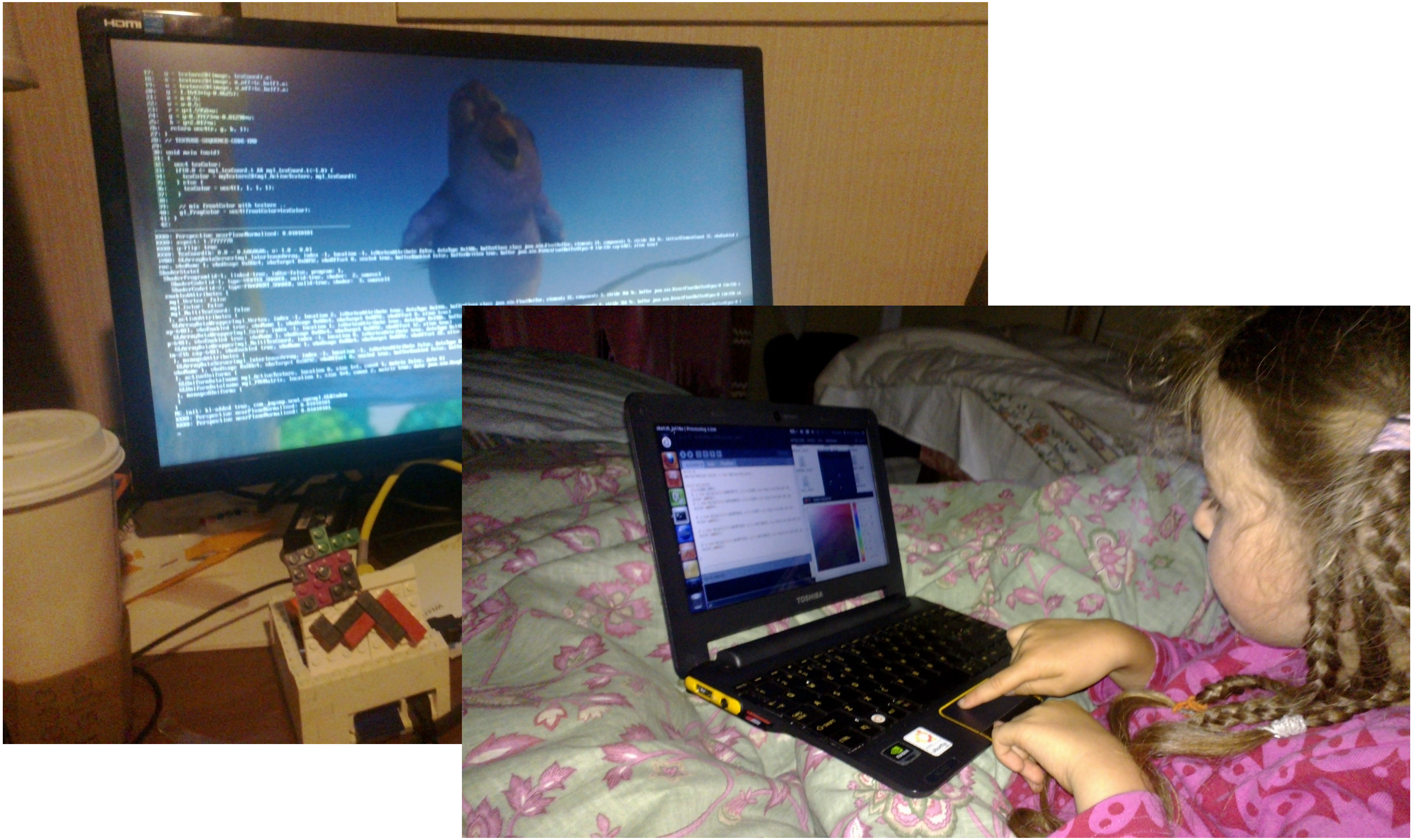
- http://www.youtube.com/watch?v=VHxtVT4tWjM



JOGL Embedded / R.-Pi



JOGL Embedded / ..



Cross Platform & Device: Use Case

C3D Visual Project Control

C3D Studio/Planner

Scenario Creation
Data Integration
...

C3D Viewer

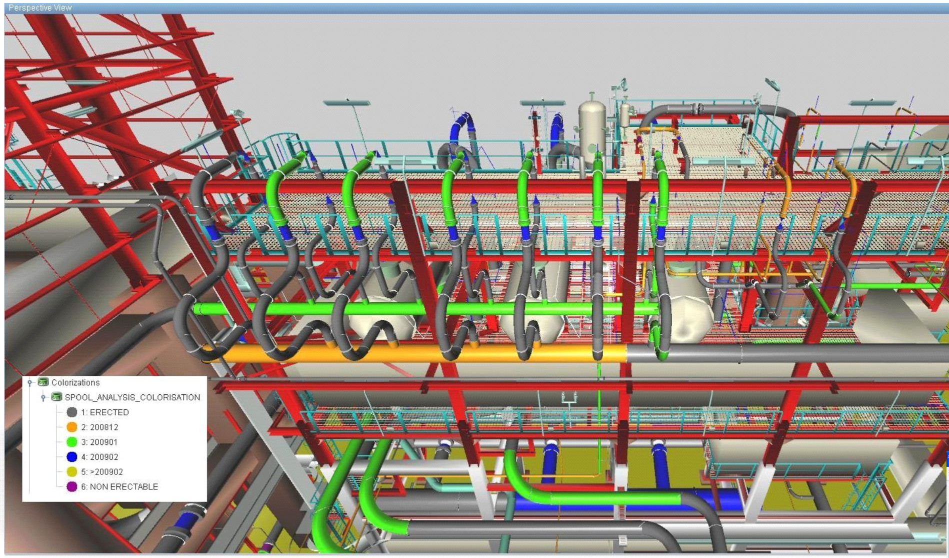
Model Visualization
Project Progress Update
4D Animation
Report Generation
Design Review

C3D Mobile

BIM Model Visualization
just-in-time progress update
...

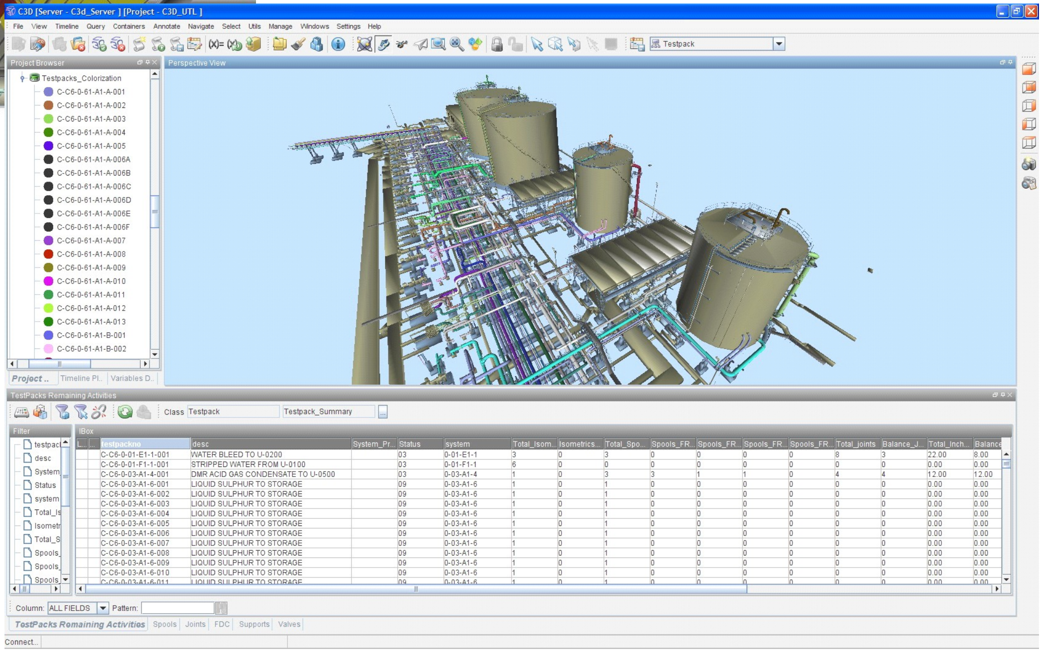
http://c3d.com

C3D - Visual Project Controls



Colorizations
SPOOL_ANALYSIS_COLORISATION
● 1: ERECTED
● 2: 200812
● 3: 200901
● 4: 200902
● 5: >200902
● 6: NON ERECTABLE

Sample usecase: Colorize by Material Delivery Date, highlighting conflicts with plan...



Testpacks_Colorization
● C-C6-0-61-A1-A-001
● C-C6-0-61-A1-A-002
● C-C6-0-61-A1-A-003
● C-C6-0-61-A1-A-004
● C-C6-0-61-A1-A-005
● C-C6-0-61-A1-A-006A
● C-C6-0-61-A1-A-006B
● C-C6-0-61-A1-A-006C
● C-C6-0-61-A1-A-006D
● C-C6-0-61-A1-A-006E
● C-C6-0-61-A1-A-006F
● C-C6-0-61-A1-A-007
● C-C6-0-61-A1-A-008
● C-C6-0-61-A1-A-009
● C-C6-0-61-A1-A-010
● C-C6-0-61-A1-A-011
● C-C6-0-61-A1-A-012
● C-C6-0-61-A1-A-013
● C-C6-0-61-A1-B-001
● C-C6-0-61-A1-B-002

Class: Testpack Testpack_Summary

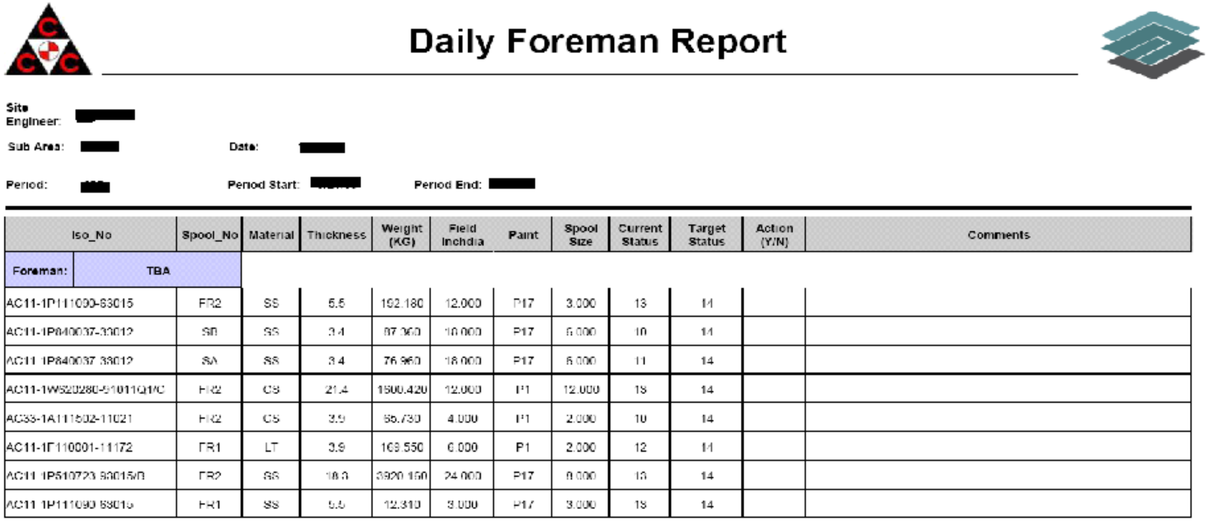
L...	...	testpackno	desc	System_Pr...	Status	system	Total_Isom...	Isometrics...	Total_Spo...	Spools_FR...	Spools_FR...	Spools_FR...	Spools_FR...	Total_joints	Balance_J...	Total_Inch...	Balance
		C-C6-0-01-E1-1-001	WATER BLEED TO U-0200		03	0-01-E1-1	3	0	3	0	0	0	0	8	3	22.00	8.00
		C-C6-0-01-F1-1-001	STRIPPED WATER FROM U-0100		03	0-01-F1-1	6	0	0	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-4-001	DMR ACID GAS CONDENSATE TO U-0500		03	0-03-A1-4	1	0	3	3	1	1	0	4	4	12.00	12.00
		C-C6-0-03-A1-6-001	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-002	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-003	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-004	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-005	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-006	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-007	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-008	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-009	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-010	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00
		C-C6-0-03-A1-6-011	LIQUID SULPHUR TO STORAGE		09	0-03-A1-6	1	0	1	0	0	0	0	0	0	0.00	0.00

Column: ALL FIELDS Pattern:

Sample usecase: Visualize remaining activities to mark testpack as done

C3D - Visual Project Controls

Sample Usage: Generate Forman daily report and task list



Daily Foreman Report

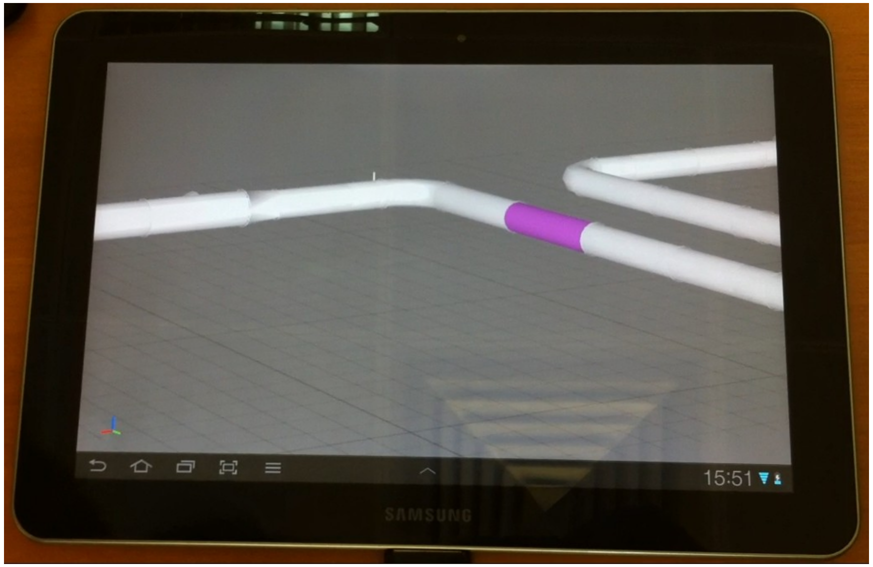
Site Engineer: [redacted]

Sub Area: [redacted] Date: [redacted]

Period: [redacted] Period Start: [redacted] Period End: [redacted]

Iso_No	Spool_No	Material	Thickness	Weight (KG)	Field Inchdia	Paint	Spool Size	Current Status	Target Status	Action (Y/N)	Comments
Foreman:	TBA										
AC11-1P111090-63015	FR2	SS	5.5	192.180	12.000	P17	3.000	13	14		
AC11-1P840037-33012	SB	SS	3.4	87.360	10.000	P17	6.000	10	14		
AC11 1P840037 33012	SA	SS	3.4	76.960	16.000	P17	6.000	11	14		
AC11-1W520280-51011Q1/C	FR2	CS	21.4	1500.420	12.000	P1	12.000	13	14		
AC33-1A111502-11021	FR2	CS	3.9	55.730	4.000	P1	2.000	10	14		
AC11-1F110001-11172	FR1	LT	3.9	169.550	6.000	P1	2.000	12	14		
AC11 1P510723 93015/B	FR2	SS	18.3	3920.150	24.000	P17	8.000	13	14		
AC11 1P111090 63015	FR1	SS	5.5	12.310	3.000	P17	3.000	13	14		

C3D Mobile: Instead of a paper; generate a BIM model for each forman



Why JogAmp on Java?

- Availability:
  - Java, OpenGL, OpenCL, OpenAL, ..
  - Multiple Vendors
    - OpenJDK / IcedTea
    - Oracle JDK
    - IBM J9, ..
    - PhoneME
    - JamVM
    - CacaoVM
    - Dalvik
  - x86, arm, ppc, sh4, ..
  - GNU/Linux, Android, BSD, Mac OSX, Solaris/OpenIndiana, MS Windows

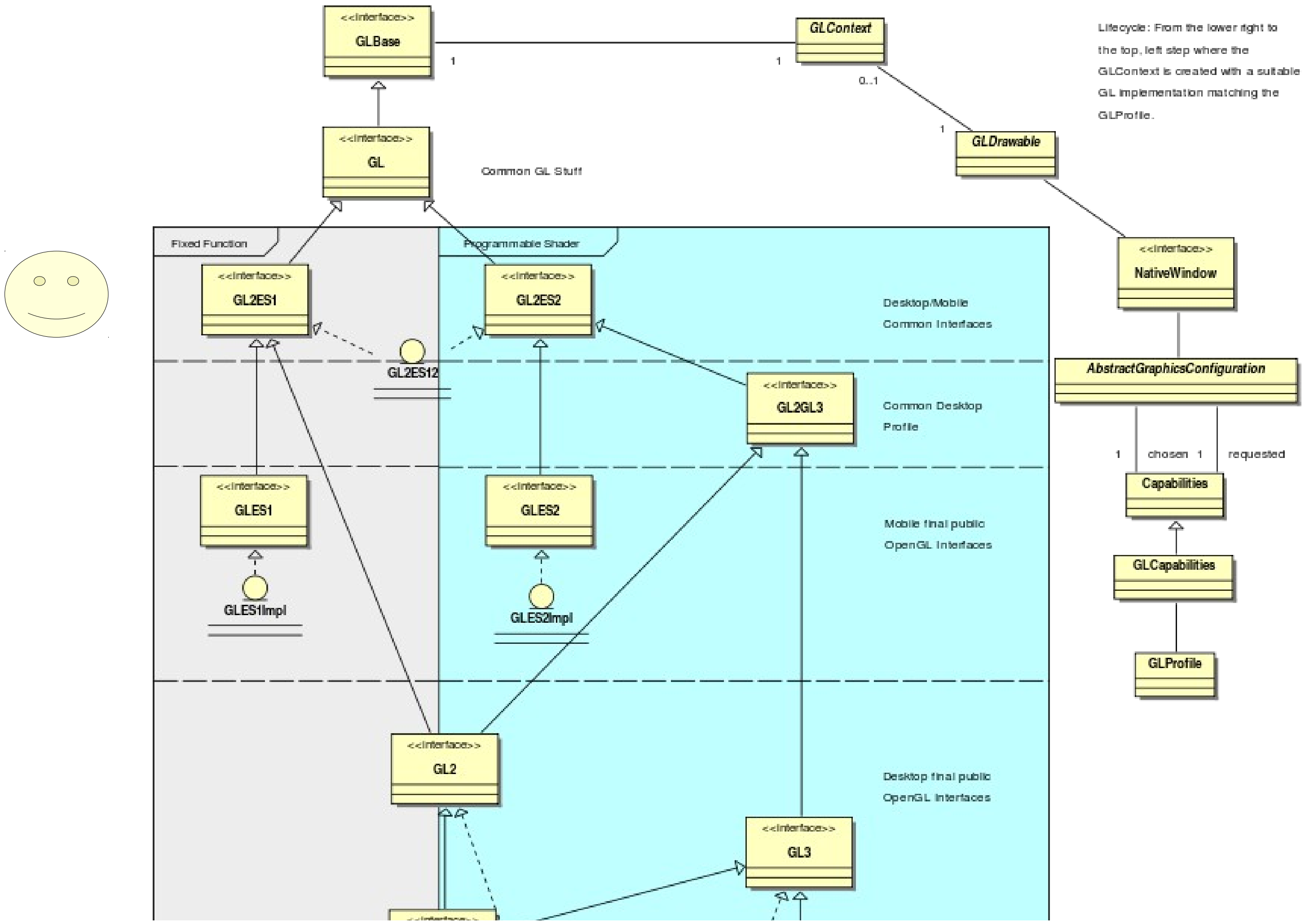
Why JogAmp on Java?

- Managed Code
  - Common API for
    - Windowing
    - GLDrawable / GLContext / GLSL
    - I/O, Resource Handling (Texture, Code, ..)
    - Rendering
  - OpenGL Pipelining / Debugging / Trace
  - Access to vast number of API / Middleware

JogAmp Continuity / Usage

- Usage http://jogamp.org
  - Ardor3D
  - C3D Studio http://c3d.com
  - Elflight Engine
  - Processing
  - Gephi
  - NASA Worldwind
  - Java3D
  - ...

OpenGL Profiles



Lifecycle: From the lower right to the top, left step where the GLContext is created with a suitable GL implementation matching the GLProfile.

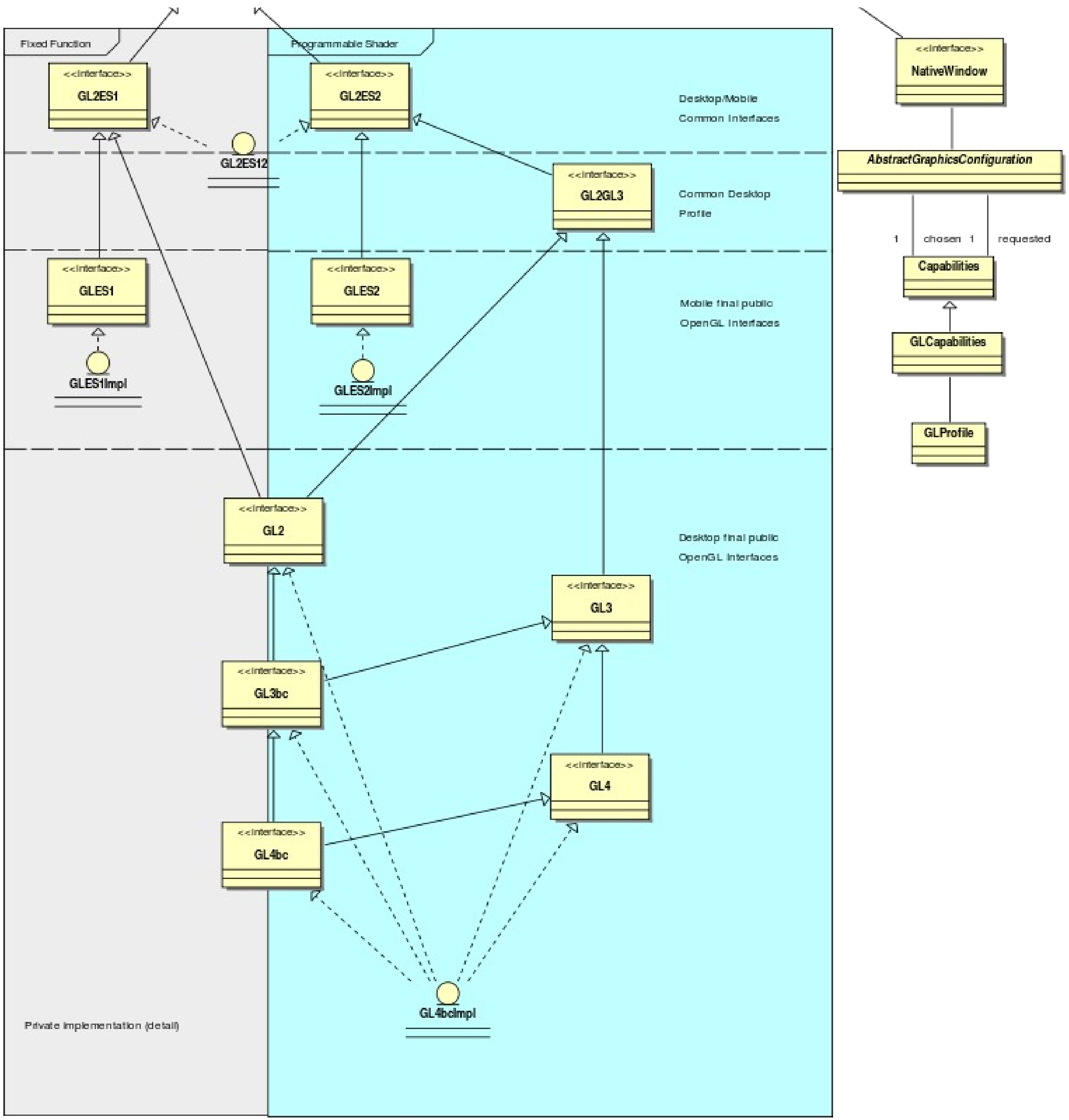
Common GL Stuff

Desktop/Mobile Common Interfaces

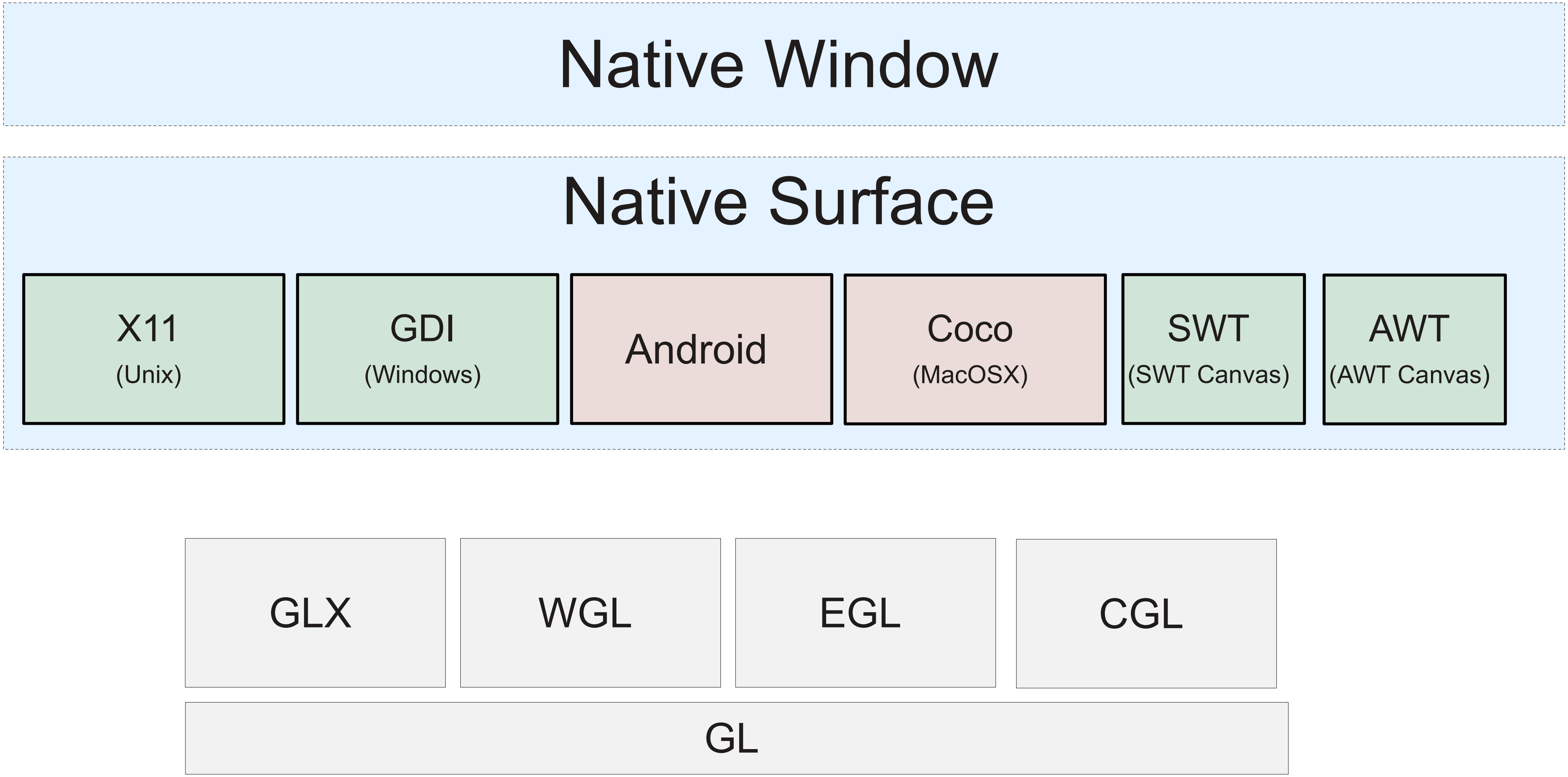
Common Desktop Profile

Mobile final public OpenGL Interfaces

Desktop final public OpenGL Interfaces


Fixed Function
Programmable Shader
<<interface>>
GL2ES1
<<interface>>
GL2ES2
Desktop/Mobile
Common Interfaces
GL2ES12
<<interface>>
GL2GL3
Common Desktop
Profile
<<interface>>
GLES1
<<interface>>
GLES2
Mobile final public
OpenGL Interfaces
GLES1Impl
GLES2Impl
<<interface>>
GL2
Desktop final public
OpenGL Interfaces
<<interface>>
GL3
<<interface>>
GL3bc
<<interface>>
GL4
<<interface>>
GL4bc
GL4bcImpl
Private implementation (detail)
<<interface>>
NativeWindow
AbstractGraphicsConfiguration
1
chosen
1
requested
Capabilities
GLCapabilities
GLProfile

Windowing Toolkits



2011 – 2012 Enhancements

- GLMediaPlayer
  - Uses OpenMAX on Android via ICS's MediaPlayer / libstagefright
  - Uses libav/libffmpeg where available
    - Missing [OpenAL] audio output
  - Missing native implementation for Win32 / OSX
- Graph API for Curve & Text rendering via GPU
  - Experimental UI
- Mobile Bindings (Android Intel/ARM, Linux ARM)
- Stability
- NEWT AWT / SWT Enhancement
- Documentation & Tutorials
- ***Higher Community Participation***

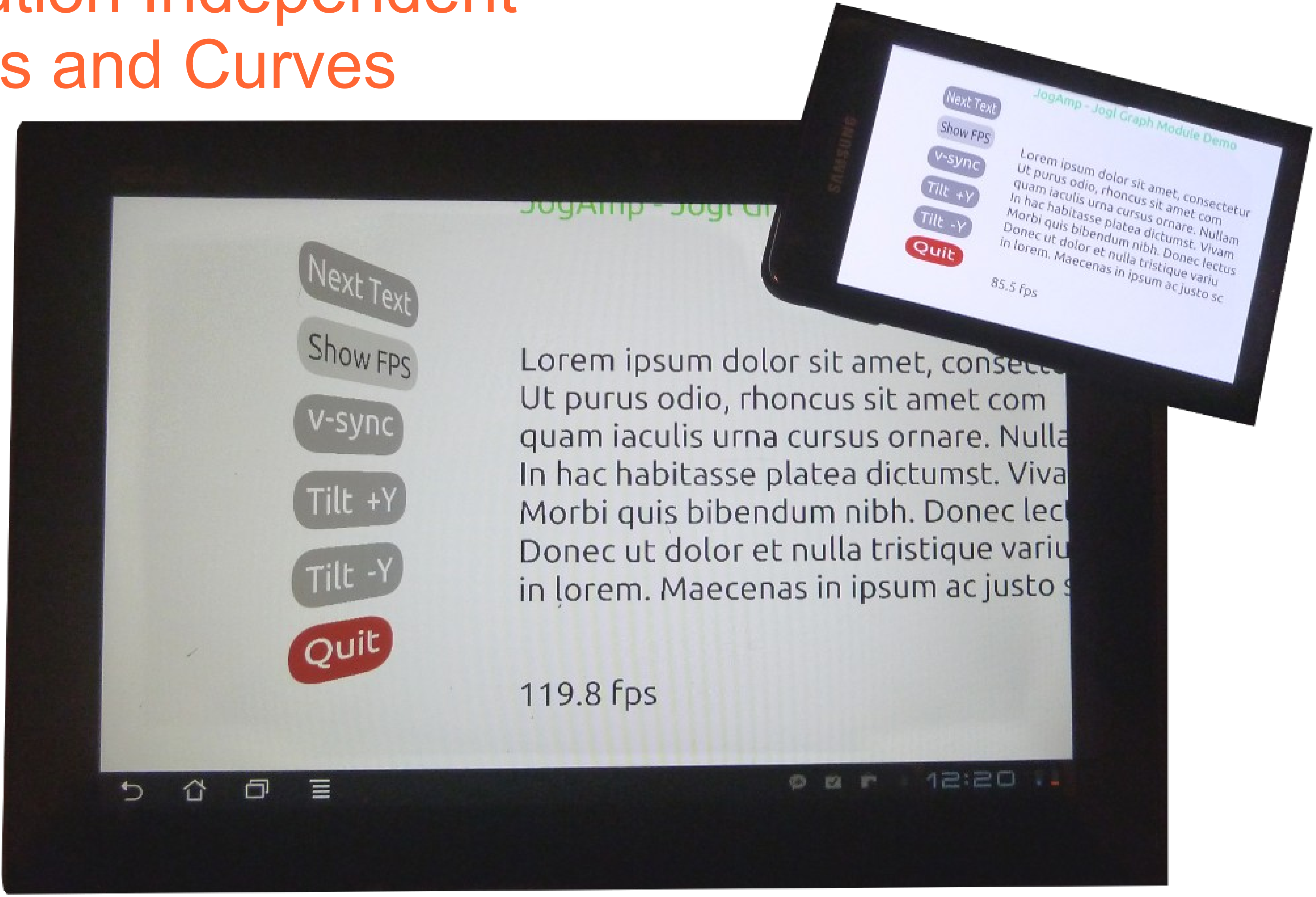
NEWT

- Seamless integration w/ native Windowing System
  - Multithreaded Access to Window Surface
  - Lock free event handling
  - Transparency, decoration and offscreen control
  - Screen Mode API (fullscreen, resolution & rotation)
  - X11, Win32, OSX, Android, OpenKD, .. implementation
  - AWT and SWT integration via native re-parenting
  - Desktop & Mobile

JOGL Android Binding

- Why?
  - Short Development Cycles
  - No device specific development
  - Supports NEWT (Multitouch, Surface, ...)
  - Same code compiled for all – almost Android agnostic.
- Deployment:
  - adb install jogl.apk
  - adb install myFancyapplication.apk
  - Manual Daisy Chained ClassLoader, if desired.

Graph API
Resolution Independent
Shapes and Curves



Lorem ipsum dolor sit amet, consectetur
Ut purus odio, rhoncus sit amet com
quam iaculis urna cursus ornare. Nullam
In hac habitasse platea dictumst. Vivam
Morbi quis bibendum nibh. Donec lectus
Donec ut dolor et nulla tristique variu
in lorem. Maecenas in ipsum ac justo sc

85.5 fps

Lorem ipsum dolor sit amet, conse
Ut purus odio, rhoncus sit amet com
quam iaculis urna cursus ornare. Nulla
In hac habitasse platea dictumst. Viva
Morbi quis bibendum nibh. Donec lec
Donec ut dolor et nulla tristique variu
in lorem. Maecenas in ipsum ac justo s

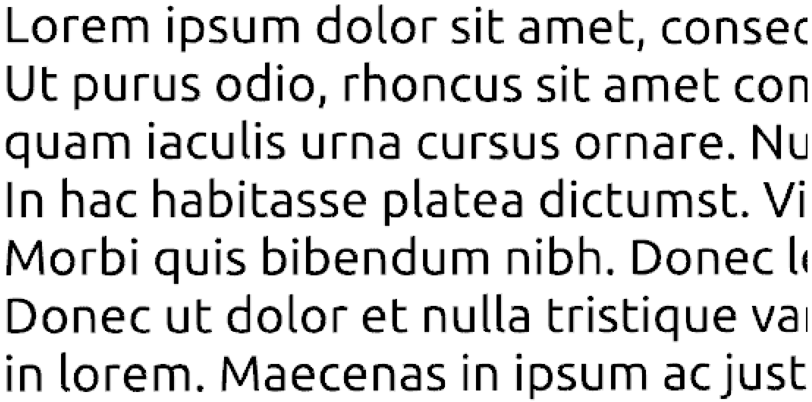
119.8 fps

Resolution Independent Curve Rendering API

- Based on Paper:
  - R Santina, “Resolution Independent NURBS Curve Rendering using Programmable Graphics Pipeline”, presented in GraphiCon2011.
- **NOT** Loop/Blinn
- Patent Free
- Can Render Bezier, Bsplines, NURBS

Resolution Independent Curve Rendering API

- Why?
  - Resolution Independent Text Rendering
  - GPU based - Fast
  - Seamless integration into Renderer (Scenegraph,...)
  - New User Interface – across devices
  - http://jogamp.org/deployment/jogamp-current/jogl-test-applets.html
  - http://www.youtube.com/watch?v=Rqsu46ifMaw


Lorem ipsum dolor sit amet, consec
Ut purus odio, rhoncus sit amet con
quam iaculis urna cursus ornare. Nu
In hac habitasse platea dictumst. Vi
Morbi quis bibendum nibh. Donec l
Donec ut dolor et nulla tristique va
in lorem. Maecenas in ipsum ac just

JOGL Graph API

- Outline → OutlineShapes → GLRegion
- Renderer
  - RegionRenderer
  - TextRenderer (same as RegionRender)
    - Helper methods for texts and fonts.

```
outline.addVertex(x, y, z, w, onCurve);
 ….
 outlineShape.addOutline(outline);
 outlineShape.addOutline(outline2);
 region = GLRegion.create(outlineShape, getRenderModes());
 region.render(gl, outlineShape,...);
```

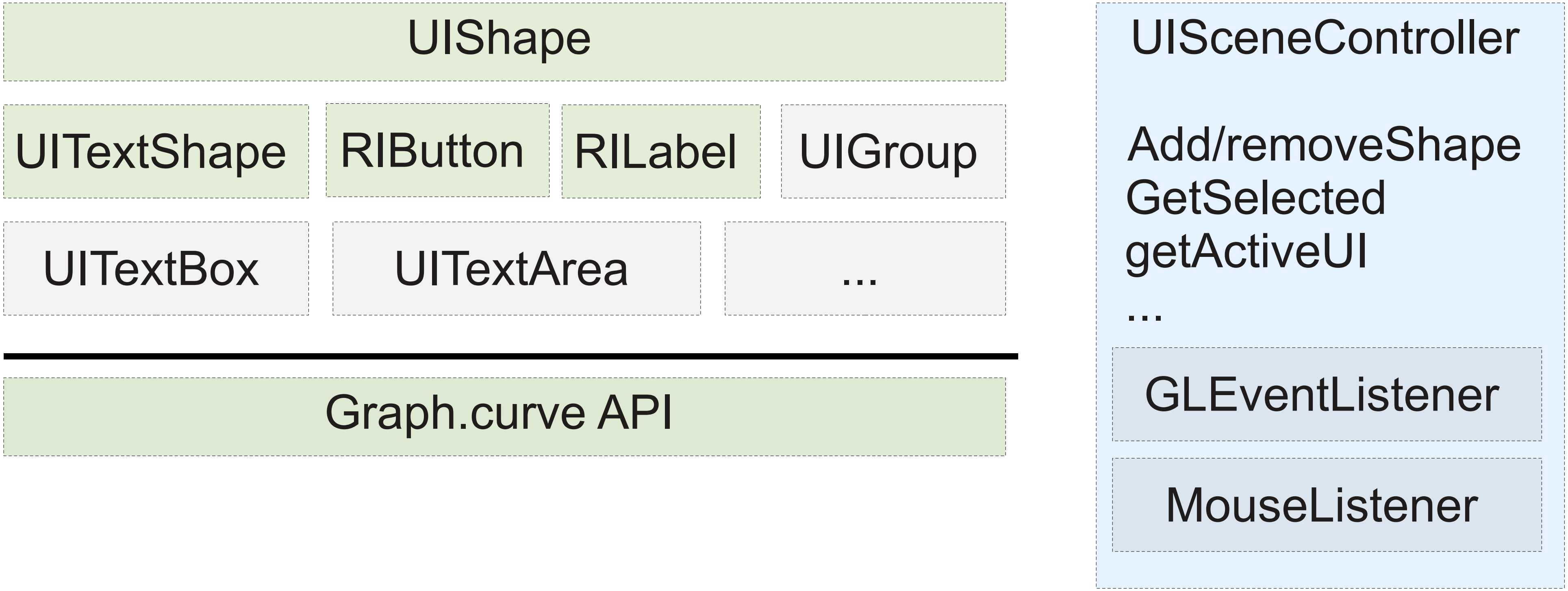
JOGL Graph API

- Initializing:
  - Read Outlines (from font, svg, application, ...)
  - Modified Constrained Delaunay Triangulation
  - Generate Region
- Rendering:
  - VBO buffers
  - Realtime manipulation – weights
  - Transformation....

GPU based Resolution Independent UI

- Abstracted from the windowing toolkit
- Support multithreading
- Seamless integration into
  - A native window (HUD)
  - A custom Scenegraph (2D plane within 3D)
- High Quality rendering
- Super Fast

JOGL Graph.UI API



UIShape

UITextShape

RIButton

RILabel

UIGroup

UITextBox

UITextArea

...

Graph.curve API

UISceneController

Add/removeShape
GetSelected
getActiveUI
...

GLEventListener

MouseListener

UI Requirements *(WIP)*

- Generic UI Rendering
  - Rendering shall be performed using native rendering TKs (JOGL, ..)
  - Render primitives on an offscreen 2D plane to be
    - integrated into a custom 3D scenegraph
    - rendered as a HUD.
- Generic User Input
  - Input events should be delegated from the custom scenegraph to the UI input module.

JOGL on Embedded Devices

- Development Env:
  - Beagleboard / Pandaboard w/ ARM7l / PowerVR
    - Linux
    - Android
  - Platform based Unit tests
  - Continuous Integration with auto-builds.
  - Cross platform compilation/building
  - Utilizing HW accelerated GL if available (EGL/ES)

JOGL Android Binding

- Details:
  - Enhanced EGL binding
  - Exposing GLES1 and GLES2 native profiles
  - GL2ES1 and GL2ES2 profiles for Desktop/Mobile
  - Using Android SDK/NDK
    - Requires SDK Level 9, Android 2.3 Gingerbread for NIO Surface access
  - Tested with:
    - Pandaboard - PowerVR
    - Samsung Galaxy S2 – Arm/Mali
    - Samsung Galaxy S – PowerVR
    - Samsung Tablet / ASUS TF2 – Tegra2
    - ASUS TF3 - Tegra3

JOGL Android Binding

- Cross platform builds/tests with Linux host
- Scripts provided in source code repository
- NEWT Helper class (NewtActivity)
  - Android Surface / NEWT Window mapping
  - Android Input Event / NEWT translation

JogAmp's Ecosystem

- Middle and high level APIs
  - Scenegraphs: Ardor3D, Java3D, JMonkeyEngine, JReality, Aviatrix3D, 3DzzD, Avengina, Xith3D, MSG
  - UI frameworks: FengGUI, Nifty GUI
  - Visualization frameworks: LibGDX, Jzy3D, GLG2D, Gephi, …
  - Sound framework: Paul Lamb Sound Library
- Low level APIs & bindings
  - JOGL, JOCL, JOAL, JInput for JogAmp

Ardor3D

- Java based retained mode 3D engine
- Runs on top of JOGL, SWT OpenGL binding…
- Supports GLSL
- Skeletal animation
- Supports Android
- Hardware accelerated UI
- Terrain system (with geometry and texture clipmaps, level of details, ...)

Ardor3D

- Pros:
  - Actively maintained
  - Most reliable JOGL based renderers
  - Abstracts rendering details but does not prevent you from extending its features with or without renderer independence
  - Render delegates used for legacy OpenGL code
  - Supports shaders (but still supports OpenGL 1.3)
  - Both community and paid support

Ardor3D

- Cons:
  - Focused on rendering (no sound, no physics, no networking, no state machines)
  - Lacks tutorials and very elaborated examples
  - Lacks importers (only Collada, OBJ and MD2)
  - Not yet any fully shader-based architecture (planned in Ardor3D 2.0)
  - No integrated game development environment
  - No build-in spatial partitioning

Q&A

- Whats Next?
- Why is neither Swing nor AWT recommended?
- What are the supported IDEs?

Thank You & Love You

Rami Santina

Sven Gothel

Xerxes Ranby

Julien Gouesse

Wade Walker

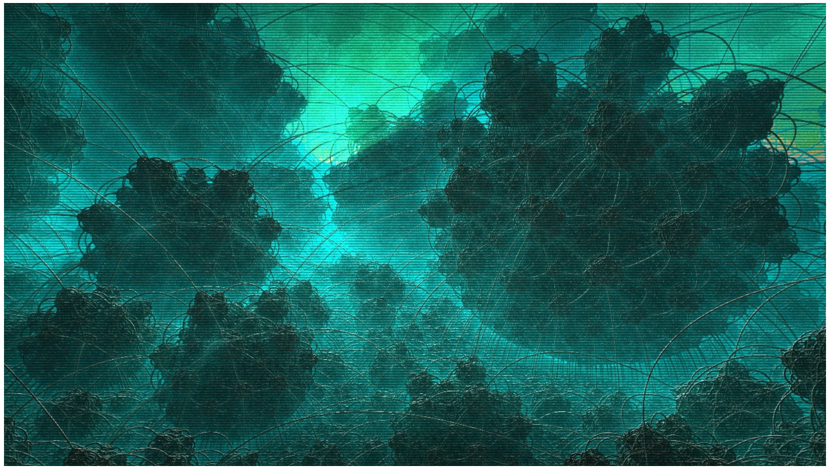
Demoscene Passivist

Mark Raynsford

Michael Bien

… all the many contributors & users

4096 bytes - *Hartverdrahtet*

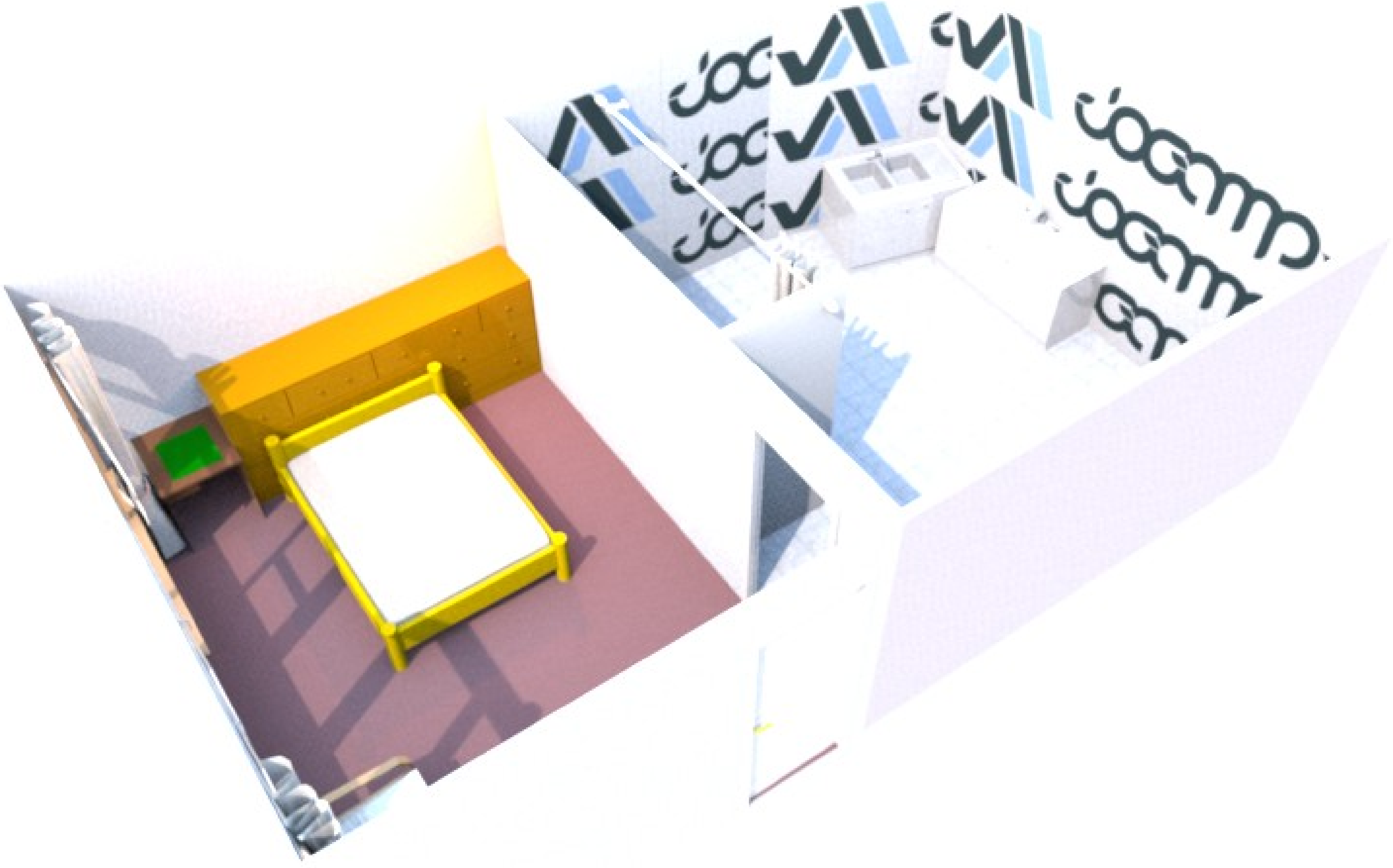


- “Demoscene” production developed with JOGL
- Total executable size including music+visuals must be <=4096 bytes (running length ~3 mins)
- *Hartverdrahtet* placed 1st at Revision 2012 (worlds biggest demoscene event)
- 1:1 JOGL port ofc with sourcecode available: http://copypastaresearch.tumblr.com/

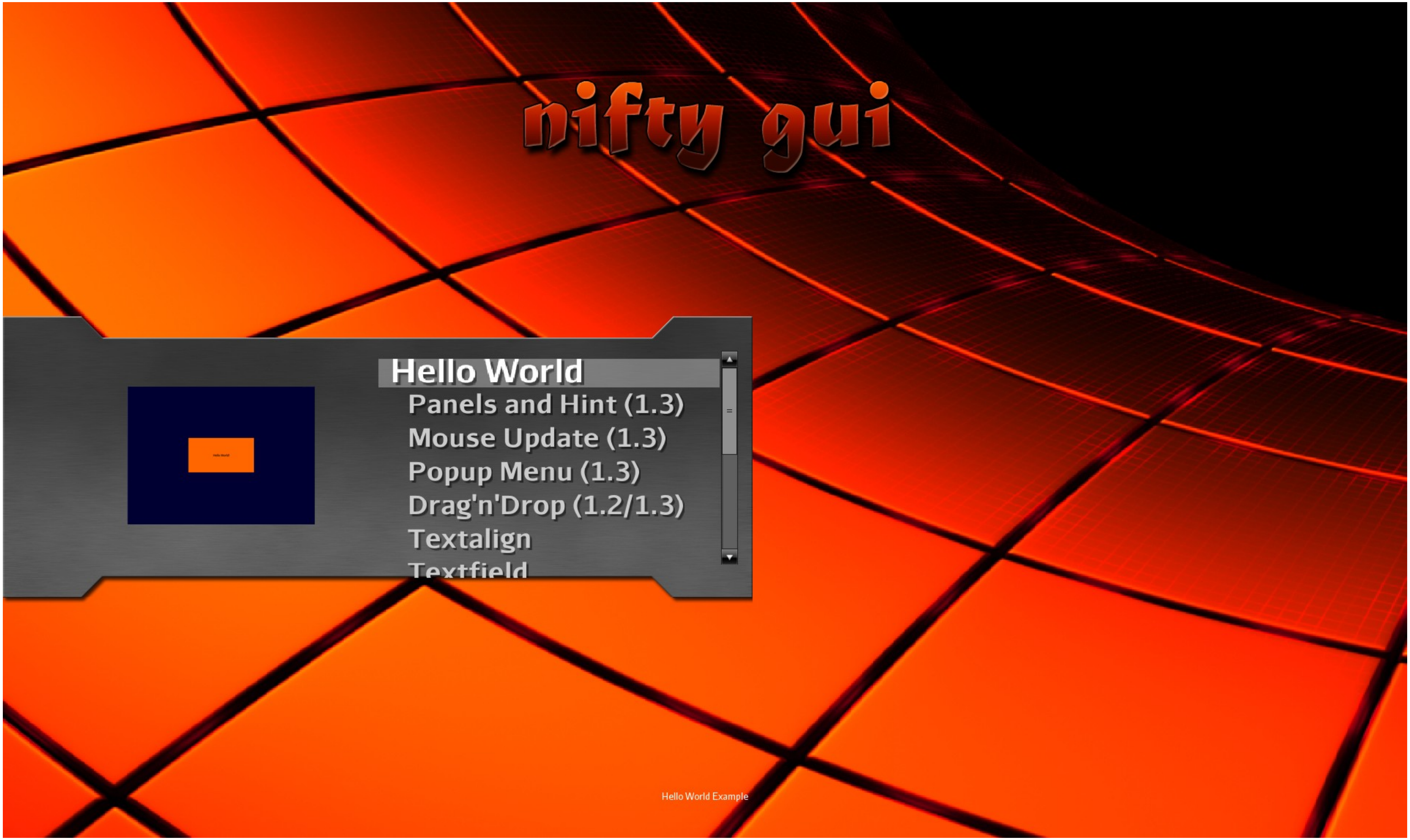
Hartverdrahtet Visuals in a Nutshell

- Single fragment shader (fullscreen billboard)
- Zero polygons! Analytical estimated surface of multidimensional IFS (fractal) volume
- Implements real-time raytracing (sphere-tracing based) with global illumination features (e.g. ambient occlusion)
- Everything is generated per pixel on-the-fly (no precalc)
- Postprocessing pass finishes the look adding volumetric lighting, noise and analog distortions
- Complete fragment shader including the raytracer, fractal, camera pathes for 10 scenes and post-effects <1500 bytes

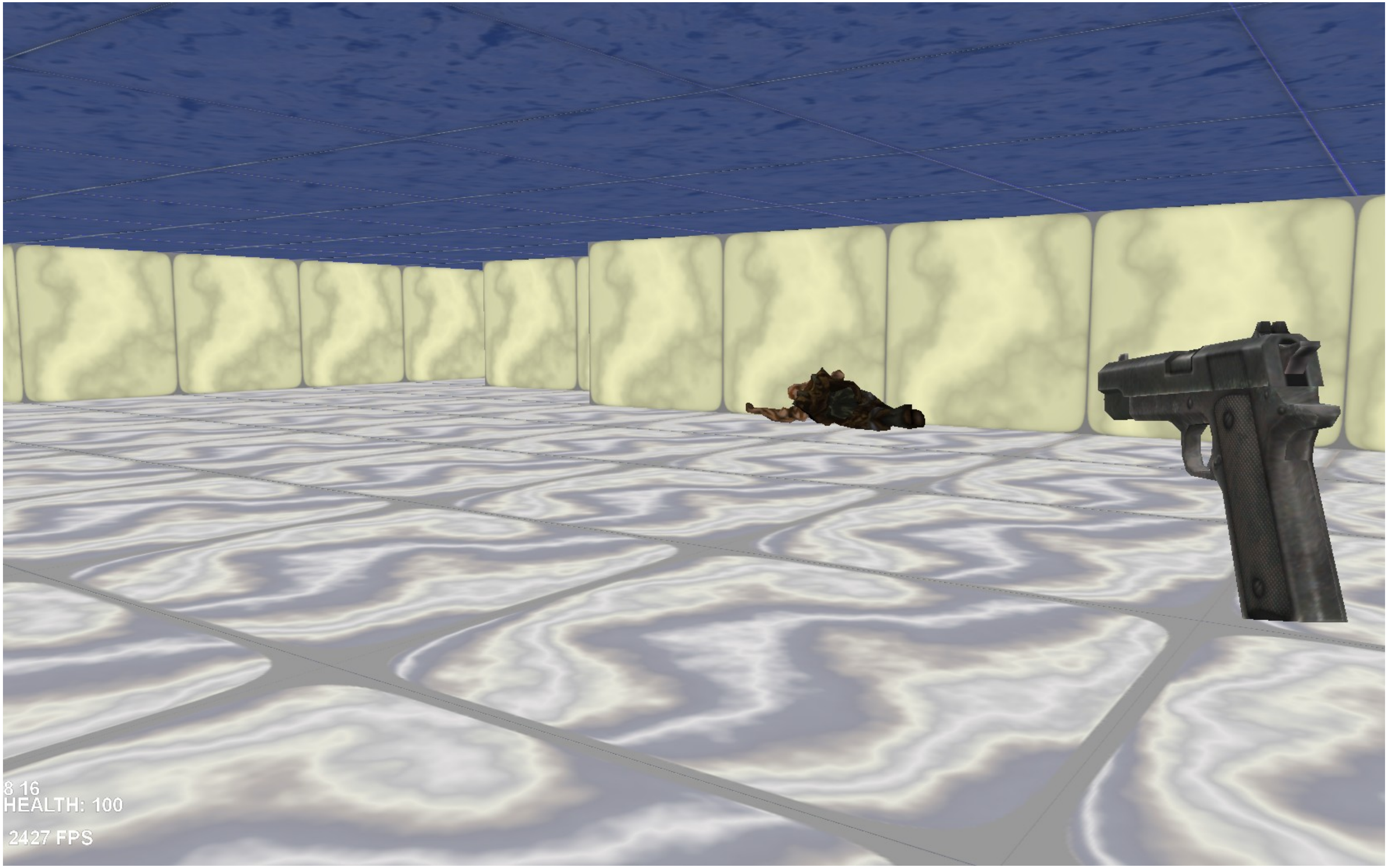
Sweet Home 3D / Java3D / Engines



NiftyGUI / Engines



Tuer / Ardor3D / Engines


8 16
HEALTH: 100
2427 FPS