



GStreamer Conference 2018

Summerhall
Edinburgh, Scotland, UK



25th and 26th October 2018

Platinum Sponsors



COLLABORA



Gold Sponsors



igalia



Fluendo



Centricular

facebook

Media Partner



UBICAST

Silver Sponsor



Welcome to the GStreamer Conference 2018 in Edinburgh!

This is now already our 9th annual GStreamer conference, and we hope we can continue with the successes of previous years.

Since its inception the GStreamer project has steadily grown in scope and popularity. Over the years GStreamer has become the standard multimedia framework for Linux-based systems and thanks to its ever improving cross-platform support it is also rapidly emerging as a standard for cross-platform multimedia development, which has become increasingly important in recent years.

We are excited to have you here and hope you enjoy the presentations, as well as the social event we have planned for Thursday evening, and of course also the informal “hallway track”.

We would like to thank all our sponsors: **Platinum sponsors Collabora and Pexip. Gold sponsors Facebook, Fluendo, Igalia, and Centricular. Silver sponsor Zeiss**, as well as our **media partner Ubcast**, without whom this event would not have been possible in this form. Thank you all very much for sponsoring the GStreamer project and the conference!

Conference Venue (Thu + Fri) Hackfest Venue (Sat + Sun)

Summerhall
1 Summerhall
Edinburgh
EH9 1PL

<http://www.summerhall.co.uk>

Wifi: Summerhall Events
Password: \$caryPanda94

CodeBase
38 Castle Terrace
Edinburgh
EH3 9DZ

<http://www.thisiscodebase.com/>

Social Event

There will be an informal social / networking event with drinks and food on Thursday evening from 19.30h onwards until late, at *Lebowski's Bar* at 18 Morrison Street, Edinburgh EH3 8BJ.

Both drinks and food will be provided (many thanks to our sponsors!), and vegetarian and vegan options will be available. <http://www.lebowskis.co.uk/edinburgh/>

Video Recordings and Slides

Talks will be recorded by Ubcast and will be available at <http://gstconf.ubicast.tv> after the conference.

Speakers, please send your slides to gstreamer-conference@list.freedesktop.org after your talk, so we can make them available on the website, thanks!

Cover: Edinburgh. View from Calton Hill, by Oliver Bonjoch. CC-BY 3.0



GStreamer State of the Union

Tim-Philipp Müller ([_tim](#)), Centricular, Thursday 9:40-10:20,

This talk will take a bird's eye look at what's been happening in and around GStreamer in the last twelve months and look forward at what's next in the pipeline.

Tim Müller is a GStreamer core developer and maintainer, and backseat release manager. He works for Centricular, an Open Source consultancy with a focus on GStreamer, cross-platform multimedia and graphics, and embedded systems. Tim lives in Bristol, UK.

GStreamer for cloud-based live video handling

Matthew Clark & Luke Moscrop, BBC, Thursday 10:30-11:00, Main Hall

The BBC is increasing the amount of live streaming on its website and apps. From breaking news to music events, there is a huge amount the BBC can offer that's not on its TV channels. But a solution to do so must be simple, flexible, cost-effective, and highly scalable. And that's where GStreamer fits in perfectly.

We at the BBC have been investigating how GStreamer could play a central role in remotely handling live streams. Live video can be easily sent to the cloud, and from there distributed to an audience. The key requirement is control - being able to see, preview, mix, and forward the live stream, without the need for traditional broadcast kit. If this can be achieved on the cloud, then flexible and highly scalable stream manipulation becomes possible. This talk is our experiences of using GStreamer to do this, and our hopes to soon do so at the BBC in production.

Matthew Clark leads the architecture for many of the BBC's websites and apps. He's overseen the design and operation of some of the BBC's biggest online events, including the Olympic Games and UK elections. He lives in Manchester, England.

Luke Moscrop is a software engineer within the BBC Live team. Having first started combing video and software engineering during university student TV and has now doing the same but at one of the worlds largest broadcasters. He is also based at the BBC's Media City offices in Salford, Greater Manchester.

Bringing Deep Neural Networks to GStreamer

Stian Selnes, Pexip, Thursday 10:30-11:00, Anatomy Lecture Theatre

Over the last decade, there's been a tremendous progress in deep learning. The field has shifted from a niche reserved only to researchers, to a mainstream technology that I've observed being discussed at a family tea party (seriously). There's now a jungle of frameworks that enable a layman to train complex models and make accurate predictions about almost anything. If you're able to navigate this jungle, you will be able to make useful applications and impressive demos with relatively little effort. And it just got even easier! Last year, OpenCV released a DNN module to run trained models. So what's more natural than to make use of this and bring the power of deep neural networks to GStreamer?!

Stian Selnes is a software engineer developing video conferencing systems at Pexip. He's been working with GStreamer, video codecs and other types of signal processing for 11 years. In an effort to be hip and trendy he's now dipping his toes into neural networks and deep learning.

When adding more threads adds more problems - Thread-sharing between elements in GStreamer

Sebastian Dröge (slozo), Centricular, Thursday 11:30-12:15, Main Hall

In GStreamer we liberally use threads everywhere: for parallelizing processing on multiple cores, for decoupling processing of different pipeline parts, for timers and for blocking operations. A normal GStreamer pipeline can easily have dozens of threads interacting with each other.

In this presentation the topic is not that using threads is hard and about all the thread-safety problems that potentially exist: by the design of GStreamer this is relatively well-handled already.

Instead the topic is the bad scalability of using a new thread for everything. If every pipeline you create uses 10 threads, and you'd like to run 100s of them on the same machine, you can easily end up using all your system resources (both CPU and memory) just for these threads while not being able to do the actual data processing fast enough or in time.

An experimental set of GStreamer elements will be introduced to show a potential solution to this problem for specific use-cases by sharing a fixed number of threads between different elements, and similar approaches to other existing elements (e.g. the RTP jitterbuffer).

Afterwards results of that approach in one specific scenario will be presented, and in the end potential future development will be discussed and what could be changed in GStreamer core for 2.0 to integrate such approaches in a cleaner and lower-overhead way to make GStreamer more scalable.

Sebastian Dröge (slozo) is a Free Software developer and one of the GStreamer maintainers and core developers. He has been involved with the project since more than 10 years now. He also contributes to various other Free Software projects, like Debian, Rust, GNOME and WebKit. While finishing his master's degree in computer sciences at the University of Paderborn in Germany, he started working as a contractor for GStreamer and related technologies. Sebastian is one of the founders of Centricular, a company providing consultancy services, where he's working from his new home in Greece on improving GStreamer and the Free Software ecosystem in general.

Apart from multimedia related topics, Sebastian has an interest in digital signal processing, programming languages, machine learning, network protocols and distributed systems.



PraxisLIVE, PraxisCORE and the Java bindings for GStreamer

Neil C Smith, Thursday 11:30-12:00, Anatomy Lecture Theatre

PraxisLIVE is a hybrid-visual FLOSS IDE and actor-based runtime for live programming, with a particular emphasis on live creative coding. The recently released PraxisCORE is a modular JVM runtime for cyberphysical programming, supporting real-time coding of real-time systems. It is the heart of PraxisLIVE. With a distributed forest-of-actors architecture, runtime code changes and comprehensive introspection, PraxisCORE brings aspects of Erlang, Smalltalk and Extempore into the Java world ... a powerful platform for media processing, data visualisation, sensors, robotics, IoT, and lots more!

A key aspect of PraxisLIVE has always been its support for developing projects mixing live and pre-recorded video with OpenGL for media artists and VJs, and the use of GStreamer actually predates inclusion of Processing for live graphics. In late-2015, after a number of stalled attempts by the Processing project to create new Java bindings for GStreamer 1.x, I took on the task of forking and maintaining the existing 0.10 bindings to work with GStreamer 1.x for use in PraxisLIVE. Since then, various other people and projects have made use of and contributed to them.

This talk and demo will cover the current state of the GStreamer 1.x Java bindings, and showcase their use with PraxisLIVE.

Neil C Smith is an Artist & Technologist from Oxford, UK. An artist working with code, he builds interactive spaces & projections, and improvised & live-coded performances. A technologist with a creative edge, he is lead developer of PraxisLIVE, maintains various Java media libraries including the bindings for GStreamer, and is an Apache NetBeans committer.

D3Dx Video Game Streaming on Windows

Florian Nierhaus, Bebo, Thursday 12:30-13:00, Main Hall

How to be successful with GStreamer on windows. A reflection on 10 months of building a high performance D3Dx Video Game Streaming pipeline and overview of the D3D11/12 GStreamer infrastructure we built, as well as our GStreamer plugins for Chromium Embedded (CEF) and NW.js.

Florian Nierhaus is Director of Engineering and co-founder at Bebo. He loves engineering challenges and building real-time multimedia applications for fun and profit.

Closed Captions in GStreamer

Edward Hervey (bilboed), Centricular, Thursday 12:30-12:50, Anatomy Lecture Theatre

While GStreamer does have support for multiple subtitle formats, Closed Caption (CC) support was always missing until now.

This talk will explain what makes CC unique compared to other formats, what new API and elements have been added to help handle it, and what remains to be done to bring it to the same level of support as other subtitling formats.

Edward Hervey has been contributing to the GStreamer project for the past 14 years, from core components to high-level projects such as the pitivi video editor. Currently a Senior Engineer at Centricular, he has helped numerous clients in current and past companies to make the most out of GStreamer in various areas. He is currently in charge of Continuous Integration and overseeing QA infrastructure for the GStreamer project.

Video Editing with GStreamer, status update and future plans

Thibault Saunier (thiblahute), Igalia, Thursday 14:00-14:30, Main Hall

Pitivi 1.0, the main GStreamer based Video editing application, is around the corner, we have been working in the last few years on stabilizing key areas of GStreamer to have a solid back-end for Video Editing.

This talk will first present the various GStreamer components we heavily rely on in Pitivi and the new "features" that have been added recently to enhance the robustness of the application.

A long term goal of Pitivi is to support professional use cases, being based on GStreamer means that our backend is very flexible and already supports important pro editing technologies. Still, we have quite a long way to go to properly support that target audience and this talk will focus on the challenges we need to overcome to reach that goal.

Thibault Saunier is a Senior Software Engineer currently working at Igalia. He is a GStreamer developer who maintains GStreamer validate, the GStreamer Video Editing Stack as well as the Pitivi video editor.



Post Mortem GStreamer Debugging with Gdb and Python

Michael Olbrich, Pengutronix, Thursday 14:00-14:30, Anatomy Lecture Theatre

There are a lot of tools available to simplify debugging GStreamer applications at runtime. Unfortunately, the situation is quite different, once an application crashes. While gdb can be used to access all the data structures, interpreting the data can be difficult and time consuming.

This talk shows how the gdb Python API to simplify debugging GStreamer applications. It will show how Python can be used to determine the overall state of the pipeline and display the GStreamer object and related data structures in a more readable format.

Michael Olbrich is an open-source developer with a focus on platform integration on embedded Linux. He works as a full-time Linux developer for Pengutronix. His job is to provide a smooth Linux experience on embedded devices from init systems to graphics and multimedia frameworks. He is the main maintainer for PTXdist, an embedded Linux distribution.

GStreamer CI for embedded platforms

Olivier Crête (ocrete), Collabora, Thursday 14:40-15:10, Main Hall

Many people at this conference use GStreamer on embedded systems, yet our Continuous Integration (CI) only runs on Intel based systems. Earlier, this year we tried to tackle this. We created a prototype CI that builds GStreamer for the Raspberry Pi platform, and then, using Collabora's LAVA infrastructure, runs gst-validate on the actual board.

During this talk, I will describe the approach we've chosen, why we chose it, how it works, and our plan to bring this to GStreamer's infrastructure once the move to GitLab is complete.

Olivier Crête has been involved in free software since 2000. He has been involved in GNOME since 2003 and in Gentoo from 2003 to 2012. He currently works for Collabora, where he leads the multimedia team. He's been an active GStreamer developer since 2007, first working on VoIP and video calls, but lately he's been working on all kinds of multimedia projects.

PipeWire wants to be your audio server too

Wim Taymans (wtay), RedHat, Thursday 14:40-15:10, Anatomy Lecture Theatre

In the last year, a solid plan has emerged to bring pro-audio and desktop audio together in PipeWire. In this talk I want to present the plans and give an overview of the work that has been done.

Wim Taymans has a computer science degree from the Katholieke Universiteit Leuven, Belgium. He co-founded the GStreamer multimedia framework in 1999. Wim Taymans is a Principal Software Engineer at Red Hat, responsible for various multimedia packages and is currently working on PipeWire.

Introduction to DeepStreamSDK

Tushar Khinvasara, Nvidia, Thursday 15:40-16:10, Main Hall

For more information about the DeepStreamSDK see:

- Intro - <https://developer.nvidia.com/deepstream-sdk>
- Details - <https://devblogs.nvidia.com/accelerate-video-analytics-deepstream-2>

Tushar is working as technical architect at Nvidia Pune center for Video Analytics Development with DeepStream. He has total 15+ yrs of SW development experience in embedded system and has extensively worked on various multimedia frameworks from NXP semiconductor's, Symbian and Android.

Multimedia support in WebKitGTK and WPE, current status and plans

Philippe Normand (philn), Igalia, Thursday 15:40-16:10, Anatomy Lecture Theatre

This talk is about multimedia support in the WPE and GTK+ WebKit ports. I will give a status update about the HTML5 features currently supported by our GStreamer backend, such as WebRTC, MSE, MediaCapabilities support. The talk would also include a brief case study about using WPE and its Cog browser on IMX6 platforms.

Philippe Normand is a software engineer working for Igalia. His expertise spans between GStreamer and WebKit, where he has been improving the multimedia backends required for the HTML5 Living Standard.

Thursday, 25th October 2018

	Main Hall	Anatomy Lecture Theatre
9:00	Registration	
9:30	<i>Opening and Welcome to the 2018 GStreamer Conference</i>	
9:40	GStreamer State of the Union <i>Tim-Philipp Müller, Centricular</i>	
10:20	mini break	
10:30	GStreamer for Cloud-based Live Video Handling <i>Matthew Clark & Luke Moscrop, BBC</i>	Bringing Deep Neural Networks to GStreamer <i>Stian Selnes, Pexip</i>
11:00	Coffee Break	
11:30	When adding more threads adds more problems - Thread-sharing between Elements <i>Sebastian Dröge (slomo), Centricular</i>	PraxisLIVE, PraxisCORE and the Java bindings for GStreamer <i>Neil C Smith</i>
12:15	mini break	
12:30	D3Dx Video Game Streaming on Windows <i>Florian Nierhaus, Bebo</i>	Closed Captions in GStreamer <i>Edward Hervey (bilboed), Centricular</i>
13:00	Lunch	
14:00	Video Editing with GStreamer, status update and plans for the future <i>Thibault Saunier (thiblahute), Igalia</i>	Post-Mortem GStreamer Debugging with Gdb and Python <i>Michael Olbrich, Pengutronix</i>
14:30	mini break	
14:40	GStreamer CI for embedded platforms <i>Olivier Crête (ocrete), Collabora</i>	Pipewire wants to be your audio server too <i>Wim Taymans (wtay), RedHat</i>
15:10	Coffee Break	
15:40	Introduction to DeepStreamSDK <i>Tushar Khinvasara, Nvidia</i>	Multimedia support in WebKitGTK and WPE <i>Philippe Normand (phln), Igalia</i>
16:10	mini break	
16:20	Taming the three-headed monster <i>Nirbheek Chauhan (nirbheek), Centricular</i>	Non-interleaved audio in GStreamer <i>George Kiagiadakis (gkiagia), Collabora</i>
16:50	mini break	
17:00	Lightning Talks	
18:30	End of Day	
19:30	Social Event Drinks and Food at Lebowkis Bar 18 Morrison Street, Edinburgh EH3 8BJ 19.30-late. Bring your badge.	

Friday, 26 th of October 2018		
	Main Hall	Anatomy Lecture Theatre
9:00	Sleep in (note we're starting a bit later on Friday, at 10.30am!)	
10:30	<p>NNStreamer: Neural Networks as GStreamer Filters MyungJoo Ham, Samsung</p>	BoF / Workshop #1
11:15	Coffee Break	
11:30	<p>Streams and collections: we're not done yet! Edward Herve (bilboed), Centricular</p>	BoF / Workshop #2
12:15	mini break	
12:30	<p>Profiling GStreamer applications with HawkTracer and tracing subsystem Marcin Kolny, Amazon</p>	<p>In which the protagonist explores why ASIO is still alive Nirbheek Chauhan (nirbheek), Centricular</p>
13:00	Lunch	
14:00	<p>Trust but verify. Our road to robust multimedia and graphics stack verification Michał Budzyński, Samsung</p>	<p>Android Camera Source 2 - A Continuation Story Justin Kim (김정석), SK Telecom</p>
14:40	Coffee Break	
15:10	<p>Microsoft Teams Connector Håvard Graff (hgr), Pexip</p>	<p>GstInference: A GStreamer Deep Learning Framework Jose Manuel Jimenez, RidgeRun</p>
15:55	mini break	
16:10	<p>Discovering Video4Linux CODECS Nicolas Dufresne (ndufresne) & Ezequiel Garcia, Collabora</p>	<p>What's new with GStreamer & Rust Sebastian Dröge (slobo), Centricular</p>
16:40	mini break	
16:50	<p>Using GStreamer for Servo's WebAudio implementation in Rust Manish Goregaokar, Mozilla</p>	BoF / Workshop #3
17:25	<i>Closing Session</i>	
17:30	End of Conference	



Taming the three-headed monster

Nirbheek Chauhan (nirbheek), Centricular, Thursday 16:20-16:50, Main Hall

Cerbero is the build tool used by the GStreamer project to provide a standard way to build binaries for all supported platforms: macOS, iOS, Android, Linux, and Windows. The tool has been instrumental in ensuring cross-platform support for GStreamer, but few understand the arcane mysteries that embody its very essence.

In this talk I shall begin with a brief description of the structure of Cerbero, followed by some of the many improvements that have been made to Cerbero over the past year; particularly the port to Meson and MSVC/Windows and the move to Python 3.

Nirbheek Chauhan writes software and hacks on GStreamer for a living and for fun. In recent times and despite his best efforts, he accidentally became a build system expert and continues to contribute to the Meson build system as a core contributor. When not fixing broken builds, he works on interesting WebRTC applications using GStreamer and complains about how slow Rust adoption is.

Non-interleaved audio in GStreamer

George Kiagiadakis (gkiagia), Collabora, Thursday 16:20-16:50, Anatomy Lecture Theatre

Earlier this year I was working on enabling support for non-interleaved audio buffers in GStreamer. In this talk, first, I will shortly explain what non-interleaved audio means and I will show some of the challenges I faced while implementing support for it. Afterwards, I will go through the new API I introduced for handling audio buffers, in order to support non-interleaved as well as interleaved audio seamlessly, explaining why it should be adopted for writing new elements that handle audio.

George Kiagiadakis is a computer science graduate from the University of Crete and a free software contributor since 2008. He got involved with GStreamer in 2009 with a Summer of Code project in KDE, from which QtGStreamer later emerged. Since 2010, he is working at Collabora where he is assisting customers with the integration of GStreamer in their products and researching new features.

Lightning Talks

Thursday 17:00-18:30, Main Hall

Short ca. 5-minute talks about a number of different topics. There is no time for questions directly after talks, but speakers will usually be happy to answer any questions later in the hallway track, at the social event, or by e-mail.

- gst-mfx, gst-msdk and the Intel Media SDK: an update. *Haihao Xiang, Intel*
- Improved flexibility and stability in GStreamer V4L2 support. *Nicolas Dufresne, Collabora*
- GstQTOverlay. *Carlos Aguero, RidgeRun*
- Documenting GStreamer. *Mathieu Duponchelle, Centricular*
- GstCUDA. *Jose Jimenez-Chavarria, RidgeRun*
- GstWebRTCBin in the real world. *Mathieu Duponchelle, Centricular*
- Servo and GStreamer. *Víctor Jáquez, Igalia*
- Interoperability between GStreamer and DirectShow. *Stéphane Cerveau, Fluendo*
- Interoperability between GStreamer and FFMPEG. *Marek Olejnik, Fluendo*
- Encrypted Media Extensions with GStreamer in WebKit. *Xabier Rodríguez Calvar, Igalia*
- DataChannels in GstWebRTC. *Matthew Waters, Centricular*
- Me TV – a journey from C and Xine to Rust and GStreamer, via D. *Russel Winder*
- GStreamer pipeline on webOS OSE. *Jimmy Ohn (온용진), LG Electronics*
- ...and many more (you can still submit lightning talks until the last minute!)

Speakers, please e-mail your slides to Tim well before the event, so that everyone can use the same computer for the presentations thus avoiding delays between the talks. There will be a countdown timer and you will have to stop after 5 minutes to make sure everyone can have their turn.

NNStreamer: Neural Networks as GStreamer Filters

MyungJoo Ham (함명주), Samsung, Friday 10:30-11:15, Main Hall

In the recent decade, we have witnessed widespread of deep neural networks and their applications. With the evolution of consumer electronics, the range of applicable devices for such deep neural networks is expanding as well to personal, mobile, or even wearable devices. The new challenge of such systems is to efficiently manage data streams between sensors (cameras, mics, radars, lidars, and so on), media filters, neural network models and their post processors, and applications. In order to tackle the challenge with less effort and more effect, we propose to implement general neural network supporting filters for GStreamer, which is actively developed and tested at <https://github.com/nnsuite/nnstreamer>

With NNStreamer, neural network developers may easily configure streams with various sensors and models and execute the streams with high efficiency. Besides, media stream developers can now use deep neural networks as yet another media filters with much less efforts.

MyungJoo Ham, Ph.D., has been working in Samsung Electronics as a software developer after receiving the Ph.D. degree from University of Illinois in 2009. Recently, he has been developing development environment and software platform for on-device AI projects varying from autonomous driving systems to consumer electronics in AI Center of Samsung. Before joining AI Center, he had worked mostly on Tizen as an architect and lead developer with responsibilities on Linux kernel, system frameworks, base libraries, .NET runtime, and so on. He has been a maintainer of a couple of Linux kernel subsystems and contributor of a few other open source projects.

Streams and collections: we're not done yet!

Edward Hervey (bilboed), Centricular, Friday 11:30-12:15, Main Hall

Decodebin3 and playbin3 brought a more efficient handling of playback use-cases by explicitly listing available streams, allowing fast stream-switching (by not decoding all streams), and a leaner codebase. The core feature for allowing this was the addition to GStreamer of collections of GstStream (i.e. explicit listing of streams).

This talk will go over proposed additions to the streams API to go the extra mile and allow use-cases that weren't possible before or weren't efficient:

- stream-selection by any element (as opposed to just decodebin3). This will allow elements such as adaptive demuxers to only download the streams really required (as opposed to all streams).
- reliably notify elements that a given stream won't be used at all downstream (to reduce resource usage even more)
- know as early as possible when elements are ready to receive processing instructions, such as seek events or stream-selection, instead of waiting for pre-rolling.
- handle scalable streams (where the base and enhancement layers are separate) such as SHVC, Dolby Vision, wavpack, and more.

Edward Hervey has been contributing to the GStreamer project for the past 14 years, from core components to high-level projects such as the pitivi video editor. Currently a Senior Engineer at Centricular, he has helped numerous clients in current and past companies to make the most out of GStreamer in various areas. He is currently in charge of Continuous Integration and overseeing QA infrastructure for the GStreamer project.

Profiling GStreamer applications with HawkTracer and tracing subsystem

Marcin Kolny, Amazon, Friday 12:30-13:00, Main Hall

HawkTracer is lightweight and low-overhead profiler that allows to define custom trace events and provides infrastructure to create post-run and live data analyzers. In this talk, I'd like to demonstrate how can GStreamer applications be profiled and tuned in real time using HawkTracer, GStreamer Tracing subsystem and gst-debugger. I'll explain base concepts of HawkTracer, how to extend the profiler and how to integrate it to existing applications and GStreamer plugins(e.g. gst-shark tracing plugins) to get live profiling data.

Marcin is a software development engineer at Amazon. He spends his free time on contributing to several open source projects. For a few years he's been also a member of GNOME Foundation, where he maintains gstreamermm library and gst-debugger application, and contributes to a few other (mostly C++ related) projects.

Marcin's started using GStreamer framework couple of years ago in his previous job, where he was responsible for delivering video library for UAV system.

In which the protagonist explores why ASIO is still alive

Nirbheek Chauhan (nirbheek), Centricular, Friday 12:30-12:50, Anatomy Lecture Theatre

ASIO is an audio API often used on Windows. It stands for Audio Stream Input/Output, which is technical jargon meant to lull you into a false sense of security. To make you believe in the High Quality of your Low-Latency Professional Audio Equipment and Software.

These are lies. One wishes the house was made of cards, because at least then one could see through it. Nay, this is a House of Leaves that leaves one imagining what Lovecraftian nightmare saw fit to create the circumstances at Steinberg GmbH that led to such an elegant heap of bitrot and backwards compatibility that Windows itself would turn away at the sight.

Nirbheek Chauhan writes software and hacks on GStreamer for a living and for fun. In recent times and despite his best efforts, he accidentally became a build system expert and continues to contribute to the Meson build system as a core contributor. When not fixing broken builds, he works on interesting WebRTC applications using GStreamer and complains about how slow Rust adoption is.



Trust but verify. Our road to robust multimedia and graphics stack verification (aka Multimedia testing on the budget for everyone)

Michał Budzyński & Marcin Kidzinski, Samsung, Friday 14:00-14:40, Main Hall

As digital appliances become more sophisticated, quality of multimedia content presented to the user can be affected by problems nearly anywhere along the stack from transmission to presentation. What is worse, the problems might surface only in exotic multi-device setups, specific network conditions or user interaction patterns.

In this talk we will go into details of our black box verification approach based on GStreamer. How do we achieve robust and mostly setup independent regression testing. What types of defects can be addressed and how are user interaction patterns modelled. Finally we will talk about our plans for future improvements, Q&A session and hopefully some discussion on other available solutions or ways to improve presented one.

Michał Budzyński and Marcin Kidzinski are multimedia Engineers at Samsung R&D Institute Poland.

Android camera source 2 - a continuation story

Justin Kim, SK Telecom, Friday 14:00-14:20, Anatomy Lecture Theatre

Although Android announced NDK API for Camera HAL3 a few years ago, GStreamer doesn't have a correspondent element to use the API yet. In the meantime, I have tried to make it, but it is still an ongoing project. Thus, in this talk, I'd like to share what I have done and what I should do more to land this element correctly.

Justin Kim has been contributing to GStreamer since 2012. He is an open source project enthusiast and recently joined ICT R&D Center, SK Telecom in Korea to spread the open source habits.

Microsoft Teams Connector

Håvard Graff, Pexip, Friday 15:10-15:55, Main Hall

In the fall of 2017 Pexip got bestowed the honor of creating a way for traditional video-conferencing endpoints to join a Microsoft Teams meeting. This meant porting much of our existing codebase from Linux to Windows, and being able to interact with our GStreamer-based mediastack using C#. This is a story about Windows, Meson, .NET, GstApp, Azure, AVX512, Consultants, PDD (Panic Driven Development), GIR-files and perfect video-streams.

Håvard Graff has been working professionally with GStreamer since 2007, for Tandberg, Cisco and now Pexip, creating video-conferencing products. The desire for quality has made him an obsessional crusader for more and better testing, and he will try to spring GstHarness on you at any given opportunity.

GstInference: A GStreamer Deep Learning Framework

Jose Manuel Jimenez, RidgeRun, Friday 15:10-15:40, Anatomy Lecture Theatre

Deep Learning has revolutionized classic computer vision techniques to enable even more intelligent and autonomous systems. Multimedia frameworks, such as GStreamer, are a basic complement of automatic recognition and classification systems.

On this talk you will hear about a suggested design for GstInference, a GStreamer framework that allows easy integration of deep learning networks into your existing pipeline. Leverage GStreamer's flexibility and scalability with your existing models and achieve high performance inference. Use your pre-trained model from the most popular machine learning frameworks (Keras, Tensorflow, Caffe) to infer and execute them in a variety of platforms (x86, iMX8, TX1/TX2).

Link in your tracking, recognition and classification networks into your existing pipeline and achieve real-time deep learning inference.

Jose Jimenez-Chavarria is a senior embedded software engineer at Ridgerun working on GNU/Linux and GStreamer based solutions since 2013. Jose has a master degree on computer science specialized on machine learning, his graduation work consisted on deep learning techniques applied in nematode segmentation for microscopy images. He's currently interested on computer vision, AI, image processing, multimedia streaming technologies and machine learning applications on embedded systems.



Discovering Video4Linux CODECs

Nicolas Dufresne (ndufresne) & Ezequiel Garcia, Collabora, Friday 16:10-16:40, Main Hall

As Video4Linux gain support for stateful and stateless CODECs, it is important to have an elegant mechanism to enumerate available CODECs and register GStreamer elements.

This talk will be presented by Ezequiel Garcia and Nicolas Dufresne. Ezequiel will present the Linux Kernel Video4Linux media controller and CODEC APIs and how we can improve the dynamic enumeration of CODEC capabilities. Nicolas will explain how GStreamer Video4linux plugin can leverage these APIs to dynamically expose these CODECs as GStreamer elements.

Nicolas Dufresne is a Principal Multimedia Engineer at Collabora. Based in Montréal, he was initially a generalist developer with background in STB development. Nicolas started in 2011 contributing to GStreamer Multimedia Framework adding infrastructure and primitives to support accelerated upload of buffers to GL textures. Today, Nicolas is implicated in both GStreamer and Linux Media communities to help create a solid support for CODEC on Linux.

Ezequiel is a software engineer. With 15 years of experience, Ezequiel has been an active Linux Kernel contributor since 2012, and maintains two Video4linux drivers. From 2015 to 2018, Ezequiel worked for a cloud video surveillance company, working on GStreamer-based applications. In 2018, he joined Collabora as a Senior Core Engineer, where he currently works on Video4Linux CODECs.

What's new with GStreamer & Rust

Sebastian Dröge (slomo), Centricular, Friday 16:10-16:40, Anatomy Lecture Theatre

Since last year's presentation about the existence of Rust bindings for GStreamer a lot has happened. Bigger parts of the GStreamer API is covered by the bindings now, many parts of the bindings API were refactored to be easier to use and there were of course also changes to the infrastructure for writing GStreamer plugins in Rust.

An overview of the most important changes since last year will be given in this presentation, as well as a (very short!) overview of how GStreamer can be used from Rust and why you should consider that for your next project.

Sebastian Dröge (slomo) is a Free Software developer and one of the GStreamer maintainers and core developers. He has been involved with the project since more than 10 years now. He also contributes to various other Free Software projects, like Debian, Rust, GNOME and WebKit. While finishing his master's degree in computer sciences at the University of Paderborn in Germany, he started working as a contractor for GStreamer and related technologies. Sebastian is one of the founders of Centricular, a company providing consultancy services, where he's working from his new home in Greece on improving GStreamer and the Free Software ecosystem in general.

Apart from multimedia related topics, Sebastian has an interest in digital signal processing, programming languages, machine learning, network protocols and distributed systems.



Using GStreamer for Servo's WebAudio implementation in Rust

Manish Goregaokar, Mozilla, Friday 16:50-17:20, Main Hall

Servo, the experimental browser engine written in Rust, is adding WebAudio support. We decided to use gstreamer-rs for handling decoding and playback, and plan to use gst-player for <audio>, <video>, and WebRTC. We found it to be very easy to use from Rust.

This talk is about our experiences with gstreamer-rs, as well as the design of servo-media and how this all comes together to create a clean WebAudio interface.

Manish Goregaokar is a Research Engineer at Mozilla working on the experimental Servo browser engine. He's also active in the Rust community, and cares a lot about making programming more accessible to others.

Platinum Sponsors



COLLABORA



Gold Sponsors



igalia



Fluendo



Centricular

facebook

Media Partner



UBICAST

Silver Sponsor

