WP6 - Integration to mobile

Markus Turtinen

Technical meeting, Brno, Czech Republic, 10-11.12.2009

WP 6 status

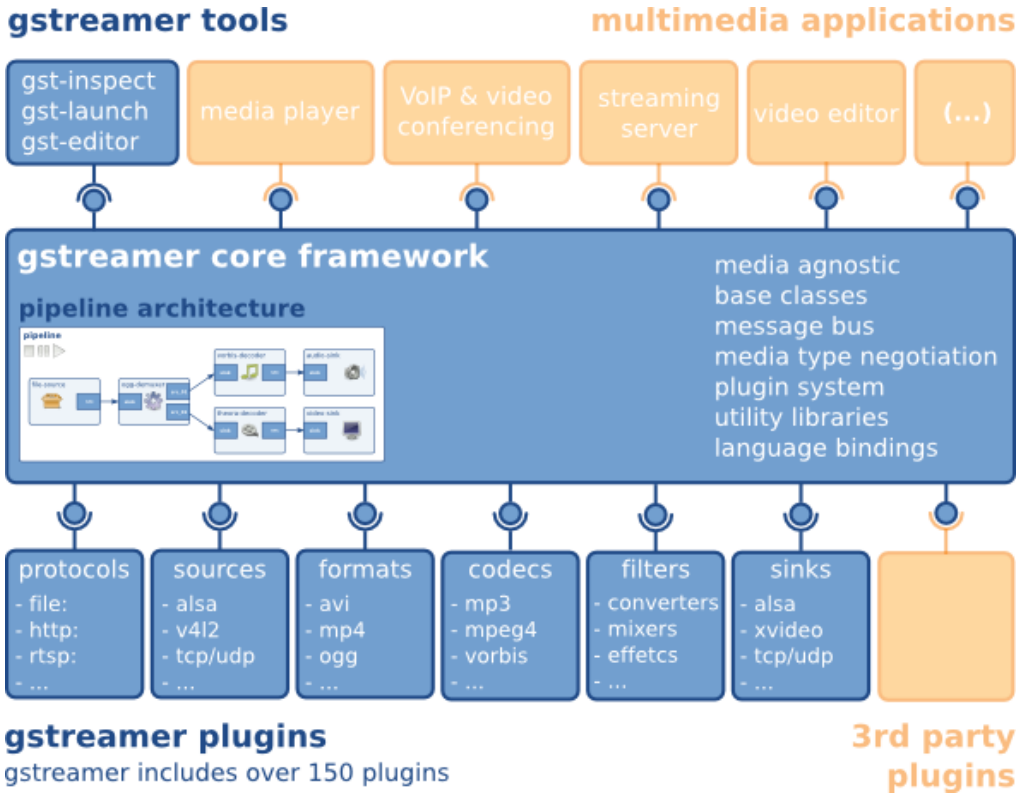
- The device for Mobio mobile experiments has been selected
  - Nokia N900 offers the most powerful and flexible platform for performing the experiments
  - N900 still suffers from some annoying bugs but they are most likely to be solved in the near future releases

N900 overview

- State-of-the-art mobile HW
  - CPU: 600Mhz ARM Cortex-A8
  - Floating-point units (pipelined, and non-pipelined, currently we can compile only non-pipelined code)
  - RAM: 256MB available for user applications
- Flexible SW development
  - Based on Maemo Linux OS
  - Freely available development tools
  - Easy development on Linux PC

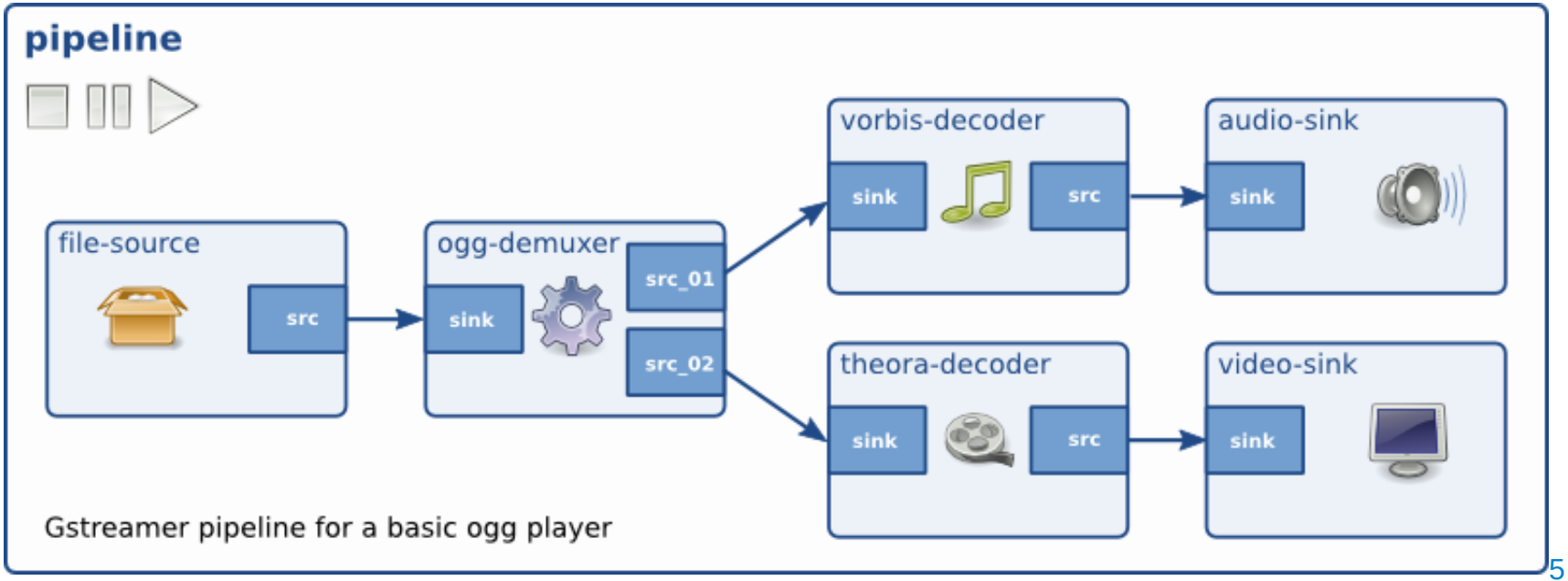
N900 multimedia framework

- The main A/V framework in Maemo Linux is Gstreamer
  - General “plug-in” based platform for handling various multimedia content
  - The main camera and audio applications in Maemo are built over Gstreamer
  - Very general framework offering the basic handles for camera and microphone



Gstreamer framework

- Basically the Gstreamer is a pipelining A/V framework
  - Data source → filters → data sink



Gstreamer pipeline for a basic ogg player

N900 – Gstreamer – and MOBIO experiments

- In Mobio experiments the Gstreamer is used for easy accessing the camera and microphone
  - For obtaining raw camera frames and audio samples
- The actual Gstreamer is hidden from biometric modules
  - The Gstreamer is only used in higher level application for obtaining data
  - We want to keep biometry modules as independent of the platform as possible for enabling easy porting to new devices
- The interfaces of biometric modules are provided without any reference to Gstreamer

Implementation issues

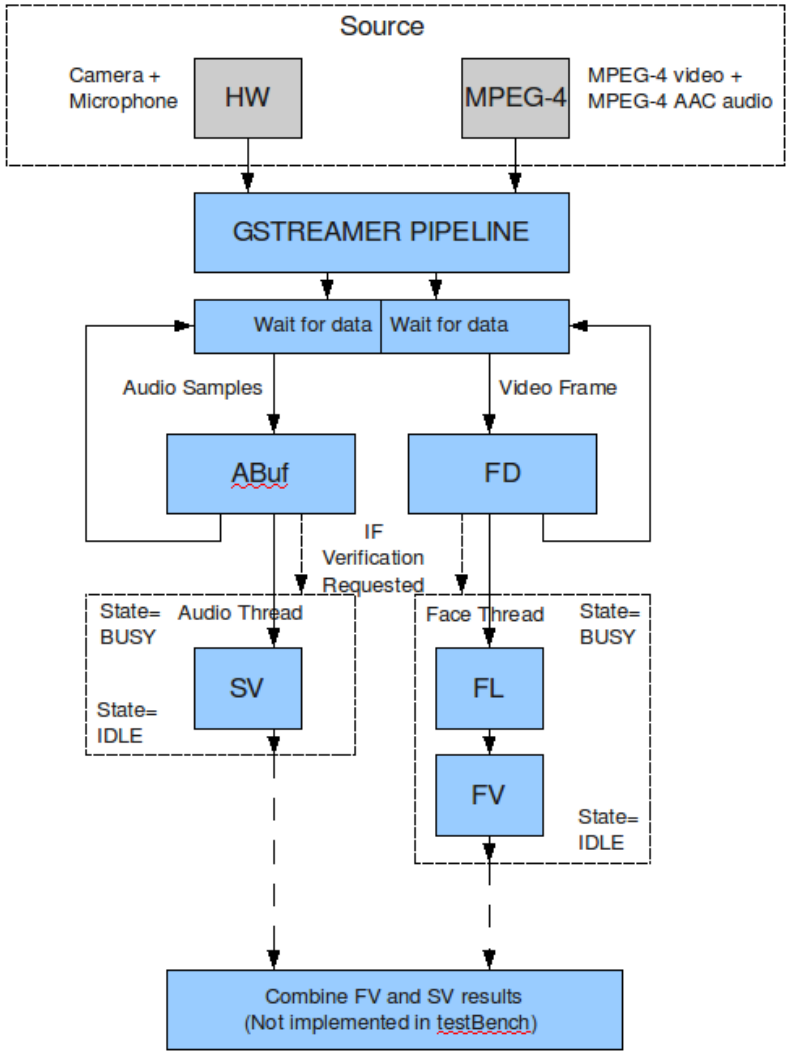
- Maemo development tools can be downloaded from the internet
  - Developing is made with a PC
  - Initial testing can be made with a PC
  - The same code (some modifications for device handles etc.) should compile for mobile and PC
  - Lot of support available from Maemo community (even though most them still consider the previous Maemo devices)
  - Maemo offers similar UI than the desktop Linux, also terminal access to make the development and experimenting easier

Implementation issues

- Even though N900 has FPU it is not as efficient as integer based operations due the very efficient pipeline of integer ALU
- Memory is always limited in embedded application and allocation is expensive. All the memory allocation should be done during the initialization of the biometric module
- File loading / saving is expensive in mobile – avoid it, model loading can be done during initialization
- Remember, the whole application should fit into 256MB (GUI, controlling SW, biometry modules, communication with the DB and server, etc. → one module can't use it all!)
- The compiler can't optimize very well – with simple coding tips you can save a lot of cycles:
  - do {} while;
  - Loop unrollment
  - No conditional branches in the intensive loops
  - Beforehand calculated look-up tables
  - Used integers when possible
  - Reuse working memory buffers
  - Etc.

Discussion on mobile integration

- There are separate interfaces for each biometry module in the MOBIO SVN
- The testbech application provides means for developing and testing the modules – on PC and N900
- It creates a basic architecture for testing
  - Gstreames is used for obtaining new image and audio samples
  - Face detection is performed for each frame
  - Audio samples are collected to the buffer
  - Speaker and face verification modules are run in their own threads
    - Started after data is available and they are ready to process new data



Discussion on mobile integration

- We need to
  - Review biometry module interfaces
    - Is there something that must be changed (i.e. you need some additional data or information from the higher level or need to produce some additional data for controlling SW)
    - MOBIO_FaceDetectionlib
    - MOBIO_FaceLocalizationlib
    - MOBIO_FaceVerificationlib
    - MOBIO_SpeakerVerificationlib
  - What about fusion
    - Do we need the actual feature data from FV or SV?
    - Or only scores?
    - Or some other data?

Thank you for your attention