Demonstrating an Enhanced Ethernet Switch Supporting Video Sensing with Dynamic QoS

http://www.ieeta.pt/lse/hartes

Rui Santos (rsantos@ua.pt), Paulo Pedreiras (pbrp@ua.pt), Luís Almeida (lda@fe.up.pt)

DCOSS'12 Hangzhou, China

May 16-18, 2012

Motivation

Video applications with **strict QoS requirements** are becoming increasingly common

- E.g. machine vision, object tracking, surveillance, driving aids
- These applications have several idiosyncrasies that must be taken into account:
  - Require a **large bandwidth**
  - Exhibit **high variability**, resulting from video compression
  - Have **dynamic behavior**
    - Sporadic activation (e .g. on-demand activation of video cameras in trucks and cars)
    - Acceptable performance can be attained with variable frame-rate, resolution, compression levels, ...
- Frequently demand **timely and predictable communications**

The **FTT-Enabled switch** features **hierarchical server-based traffic scheduling**, providing:

- **Disciplined use of the network**, thus coping with the large variability of compressed video sources;
- **Adaptability**, since allows changing the servers budget and thus varying dynamically the bandwidth allocated to each video stream
- **Reconfigurability**, since permits adding and removing servers dynamically, and consequently enabling and disabling video-streams

These features are demonstrated via a simplified video surveillance application

Traffic scheduling framework

A flexible multi-level hierarchical server-based scheduling framework for switched Ethernet that:

- Divides, subdivides and so on the network bandwidth in a hierarchical way, creating virtual channels
- Provides real-time guarantees and differentiated QoS (using admission control)
- Reconfigure and adapts the channel for varying flows
- Assures temporal isolation among channels

Model Definition

$$Srv_{y_x} = (C_{y_x}, T_{y_x}, Mmax_{y_x}, Mmin_{y_x}, P_{y_x}, RT_{y_x})$$

$$AS_{y_x} = (C_{y_x}, Tmit_{y_x}, Mmax_{y_x}, Mmin_{y_x}, P_{y_x}, RT_{y_x})$$

C_{y_x} - Maximum message transmission time

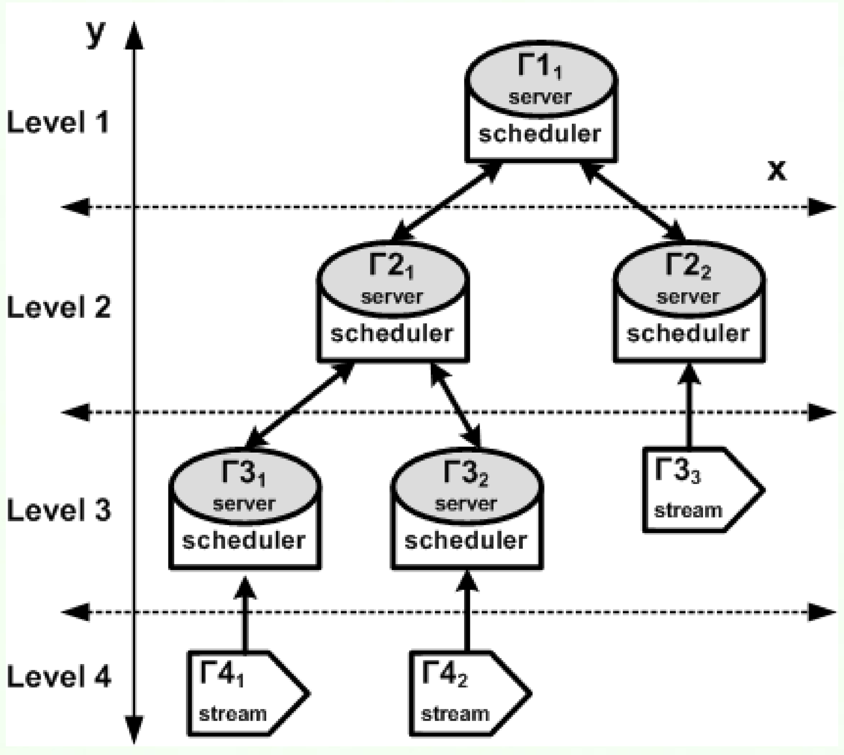
T_{y_x} - Period

$Mmax_{y_x}$ - Transmission time - largest packet

$Mmin_{y_x}$ - Transmission time - smallest packet

P_{y_x} - Parent component (server)

RT_{y_x} - Response time of each component

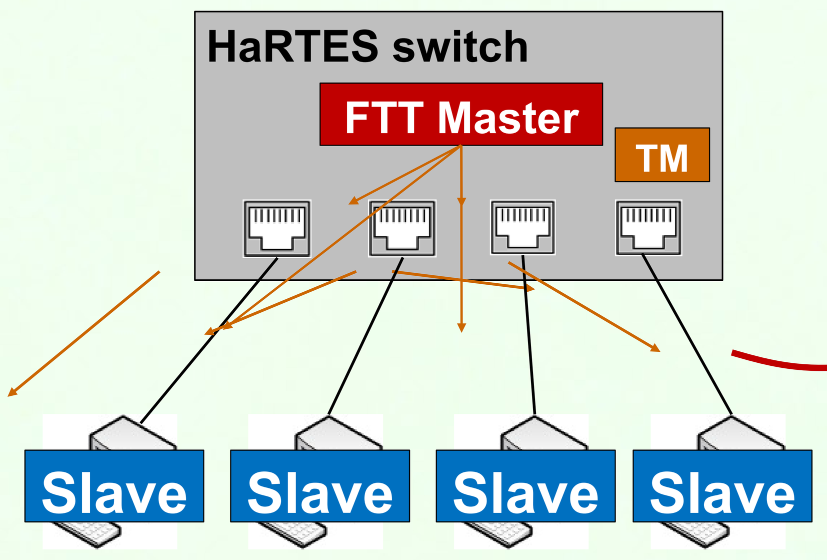


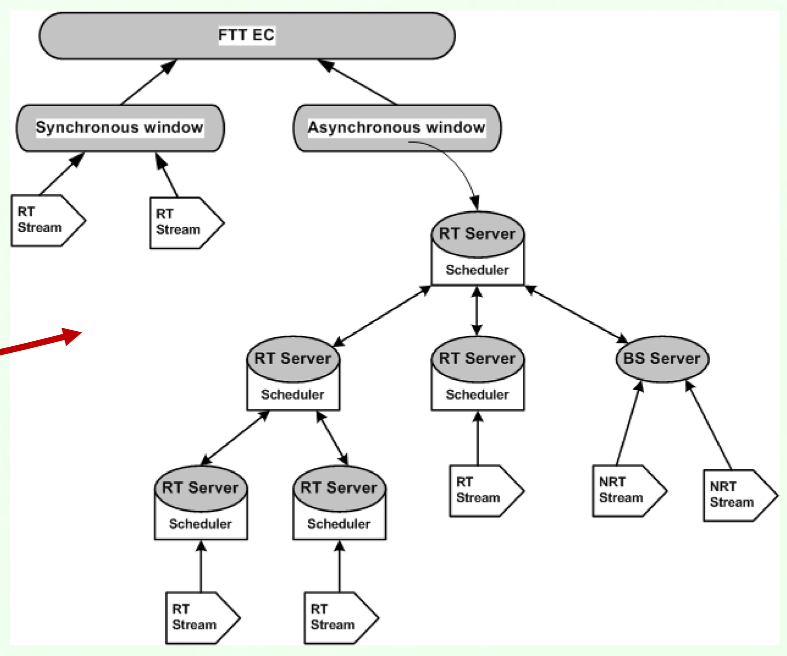
Application Platform

The server-based traffic scheduling has been deployed using HaRTES Switch (FTT-enabled Ethernet Switch)

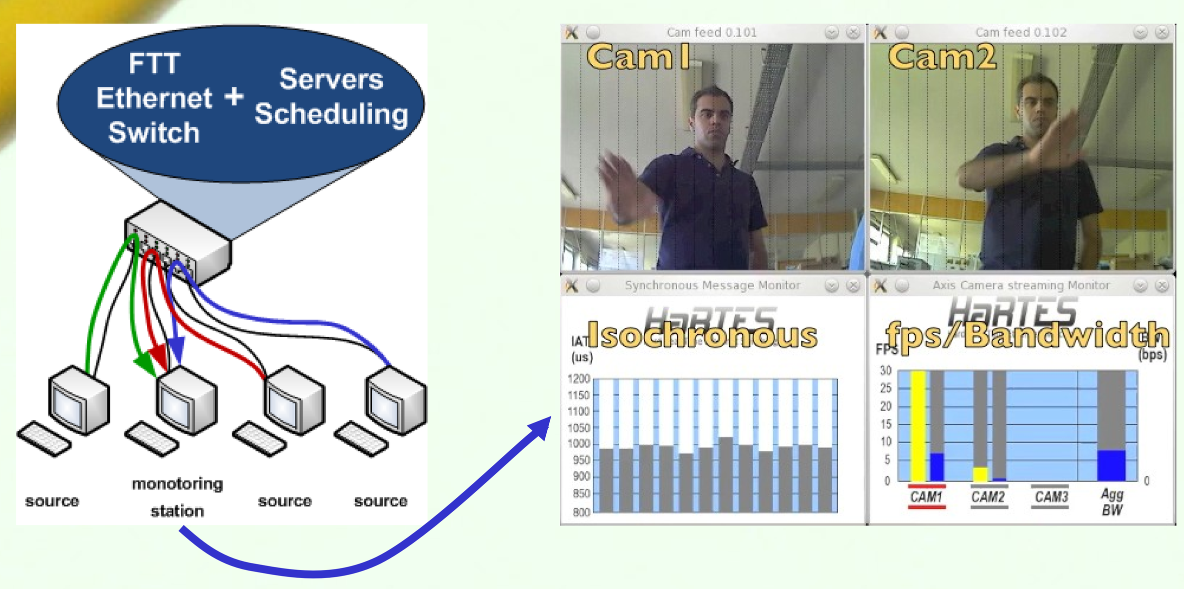
HaRTES Switch:

- Based on FTT paradigm – master / multi-slave transmission control technique
- Communications organized in Elementary Cycles (divided in synchronous and asynchronous windows)
- The asynchronous window is handled by the presented **server framework**





Demonstration



Demo 1 shows:

- The dynamic reconfiguration of the servers
  Changes are made online without any impact on the streams not affected by the reconfiguration
- Traffic isolation among the asynchronous streams
  Sources are not allowed to use more bandwidth than the one granted by the server

For that:

- 3 asynchronous video streams transmitted to the monitoring station

Demo 2 shows:

- Strict periodicity of the isochronous services
- Traffic isolation between isochronous and asynchronous services

The bandwidth, latency and jitter of the isochronous traffic is kept constant, independently of the video streams load

For that:

- 2 asynchronous video streams and one periodic data source concurrently transmitted to the monitoring station

This work was partially supported by the Portuguese Government through FCT - "Fundação para a Ciência e a Tecnologia" in the scope of project Serv-CPS -PTDC/EEA-AUT/122362/2010