TINE Video System

A Modular, Well-Defined, Component-Based and Interoperable TV System

Proceedings On Redesign → VSv3

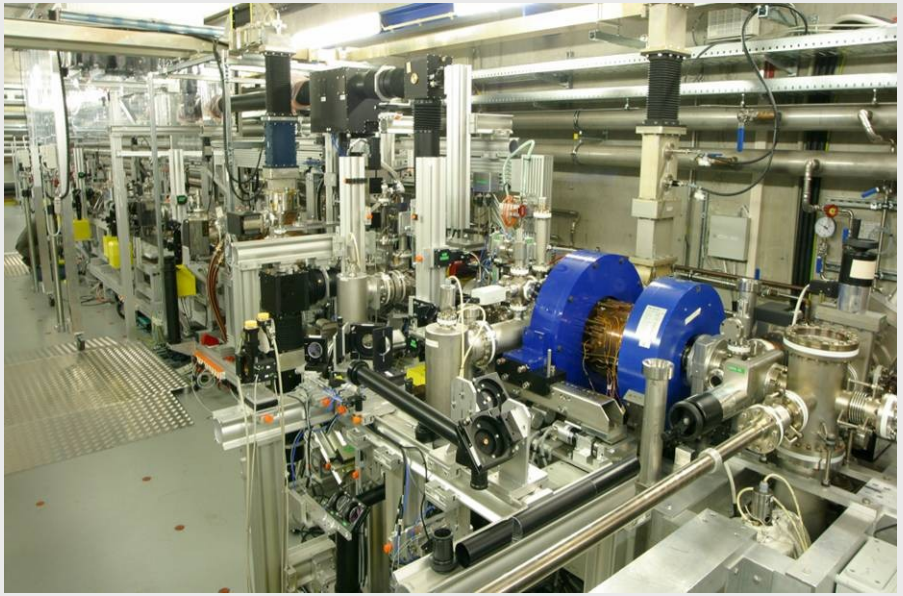
Stefan Weisse, David Melkumyan, Philip Duval

DESY, Germany

Where That Comes From: PITZ

Photo Injector Test Facility Zeuthen (2002 on)

- test, condition and optimize sources of high brightness electron beams for future free electron lasers and linear colliders
- goal: intense electron-beam with very small transverse emittance and reasonably small longitudinal emittance
- goal is requirement for FEL operation



"The challenge of PITZ is the production of such beams with very high quality by applying the most advanced techniques in combination with key parameters of projects based on TESLA technology like the FLASH and the European XFEL."

A Decade of History

2000 **TINE Video System (HH)**

2002 **PITZ Video System (Zeuthen)**

- pre-stage of VSv2

2003-2008 **TINE Video System 2 (VSv2)**

- client/server architecture
- wrapping, encapsulation (for easier reuse)
- still in use (hybrid VSv2 / VSv3)

2003 installed at PITZ
2006 installed at DESY Hamburg (“DESY-2” and around)
2007 installed at EMBL Hamburg

2008 (and on) **TINE Video System 3 (VSv3)**

- modular, component based, interoperable, well-defined, user-friendly
- step by step taking over

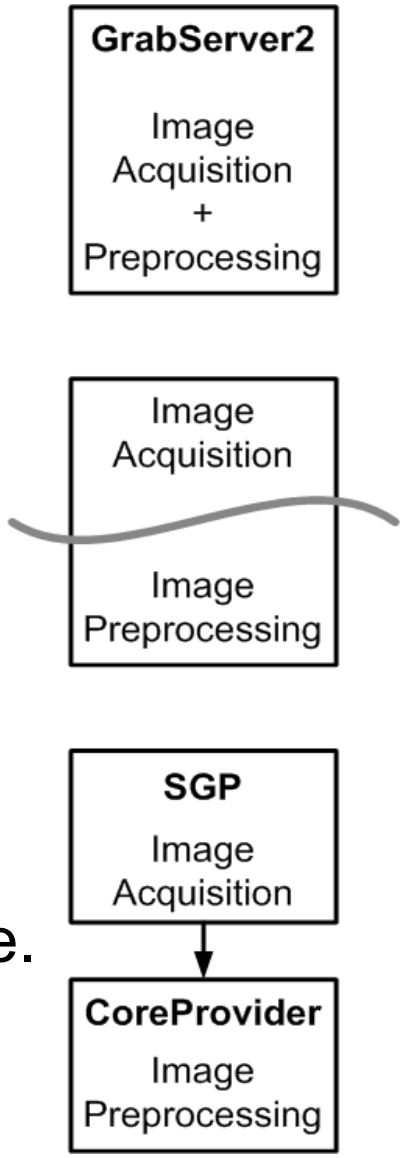
TINE Video System 3 (VSv3)

- modular, component-based
  - each component should only do its primary purpose and nothing else (if not defeated by external influences)
- interoperable
  - interfaces to other control system protocols
  - interfaces to the “outside world”
- well-defined
  - specification separated from implementation (available as plain text files)
- use of standards where possible
  - PNG for video images
  - XML for configuration (files): speaking names, self-explanatory, commented
- user-friendly, multiplatform
  - main components are Windows (server) and Java-based (client, server)
  - by rule: C++ sourcecode does not contain much platform dependency
- while still keeping it
  - easy to use, capable of high-performance, low-latency, lossless if necessary

Overview of Changes

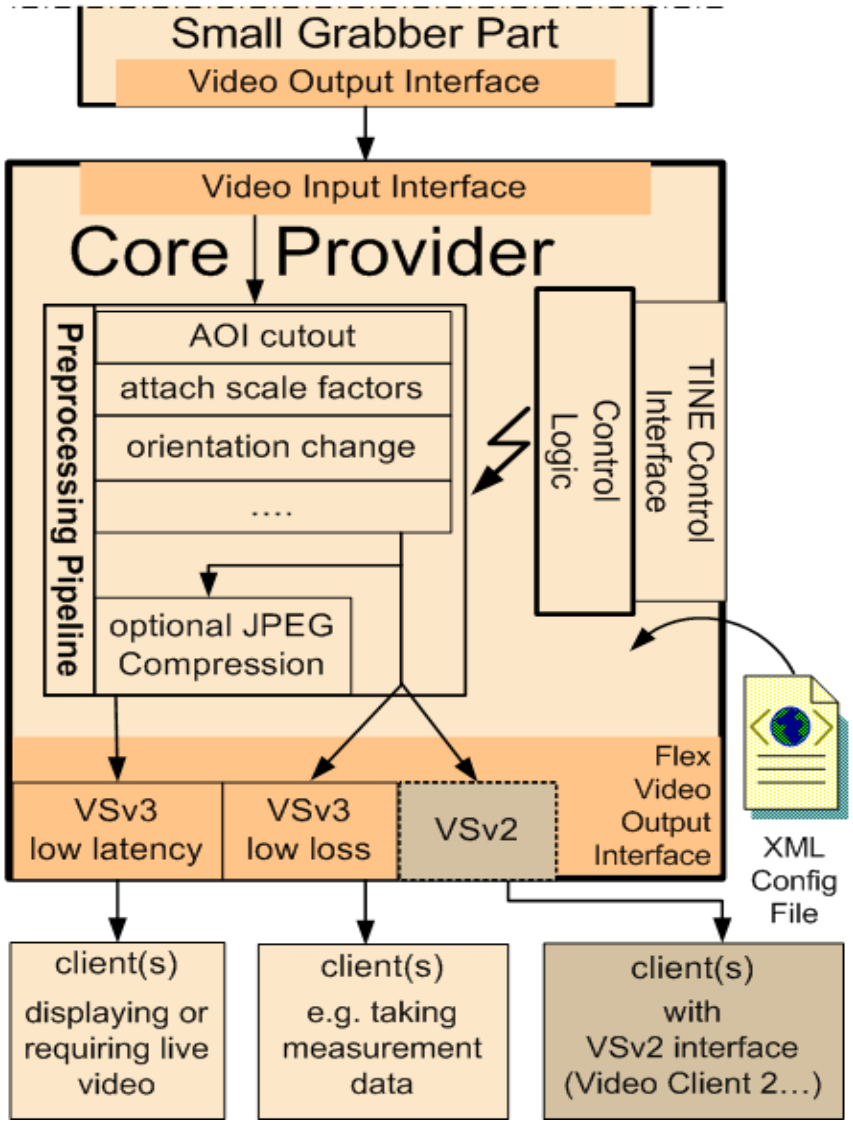
Central points worked on the last year

- completing front-end component (Small Grabber Part), gain experience by near production-level test-installs
- build up Core Provider (hardware-independent video image preprocessing) component
- Standard Image Formats (PNG)
- Universal Slow Control
- polishing Java video receive and display on client-side
- unleash performance (Gigabit Ethernet video transfer)



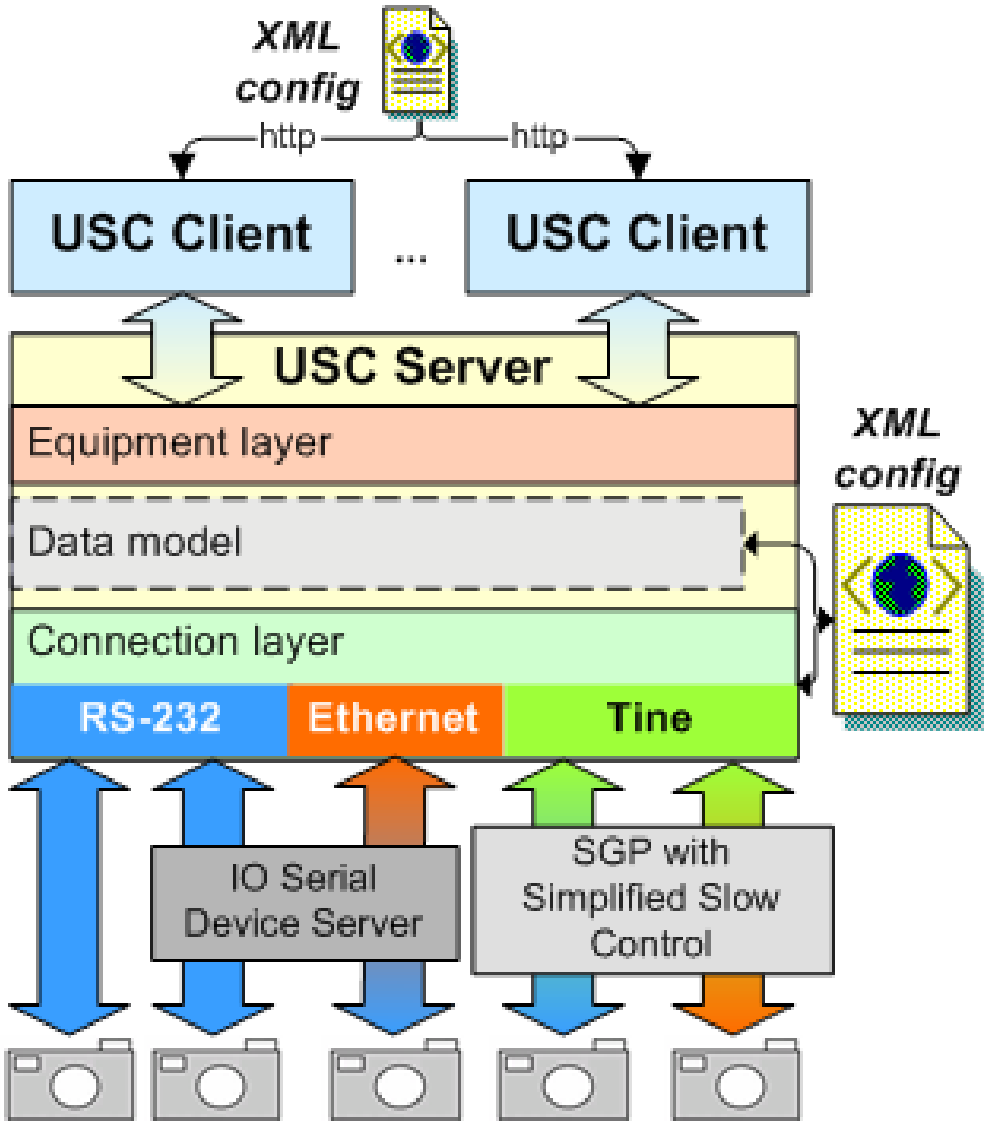
See paper for more detailed list of changes, please.

Core Provider



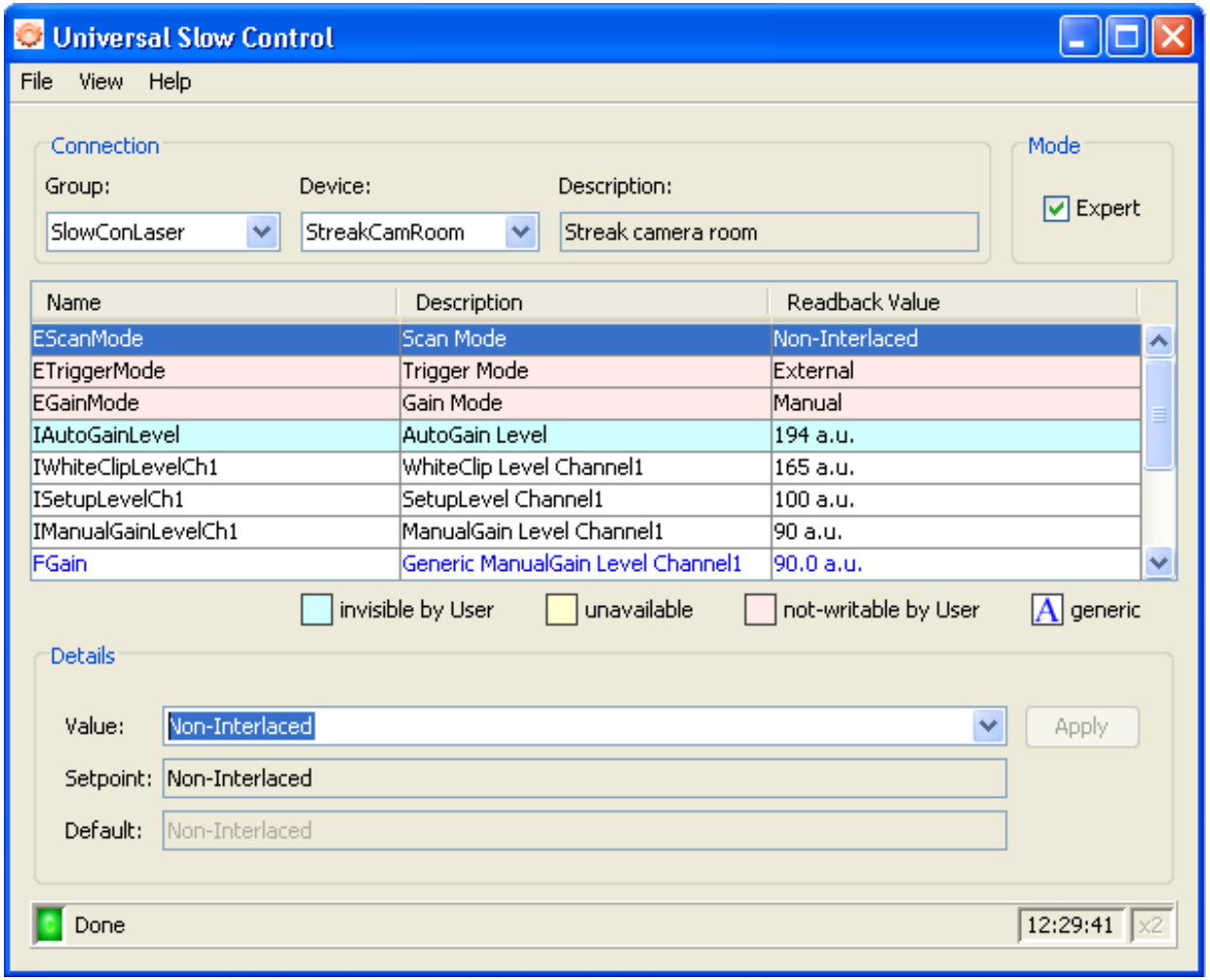
- hardware-independent pre-processing of video data (if necessary)
- can run as companion right beside SGP
- takes live video from component in front
- performs tunable pre-processing
- provides 3 different output interfaces, including support for old-style clients

Universal Slow Control



- more general approach of slow control
- user/expert view on camera properties
- well-defined, separately documented TINE property interface
- subset: Generic Slow Control
- native Java server and rich Java client, TINE communication
- abstract design: control of other devices than cameras can be considered

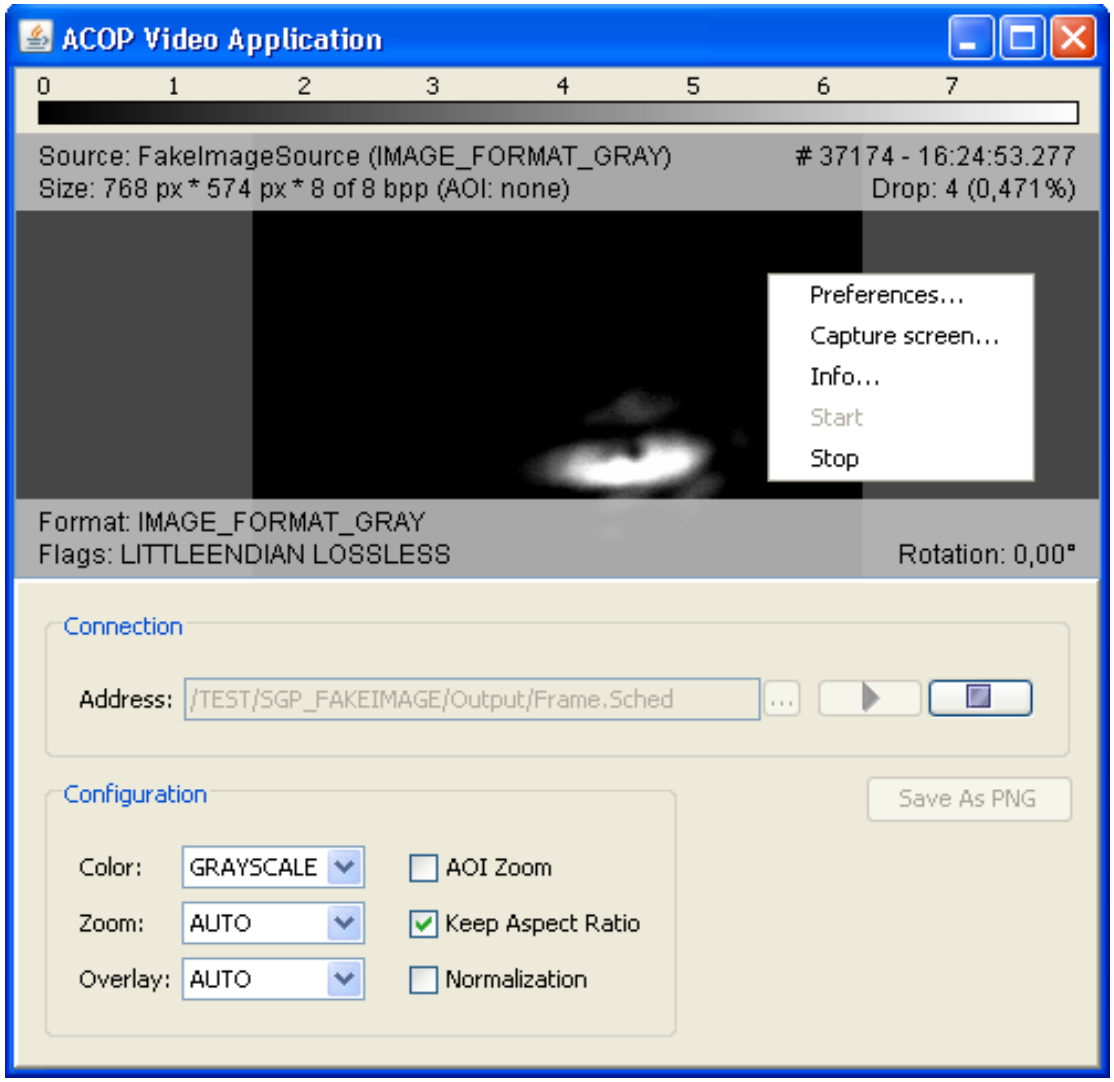
Universal Slow Control Client



- rich Java client application
- TINE for control communication
- Java Web Start
- default GUI for general access to camera slow-control, fitting user and expert needs

runs where Java 6 is available (verified Linux, Windows)

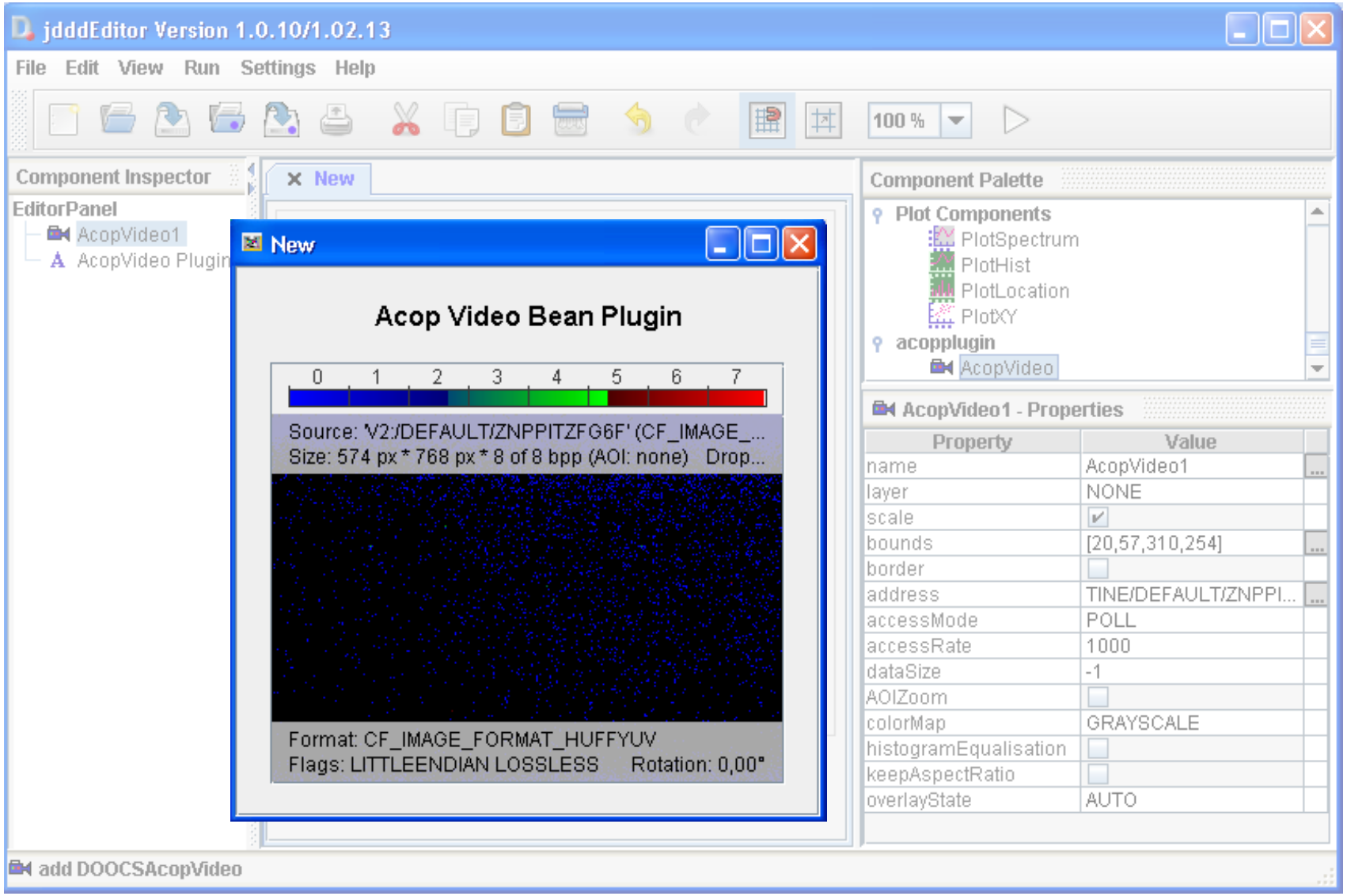
ACOP Video Bean



- rounded corners based on user feedback
- compatible to COMA
- interface was added to intermediate analysis server*
- stable, mature
- development is continued (commissioning phase)

* work done by Cosylab

ACOP Video Bean + jDDD

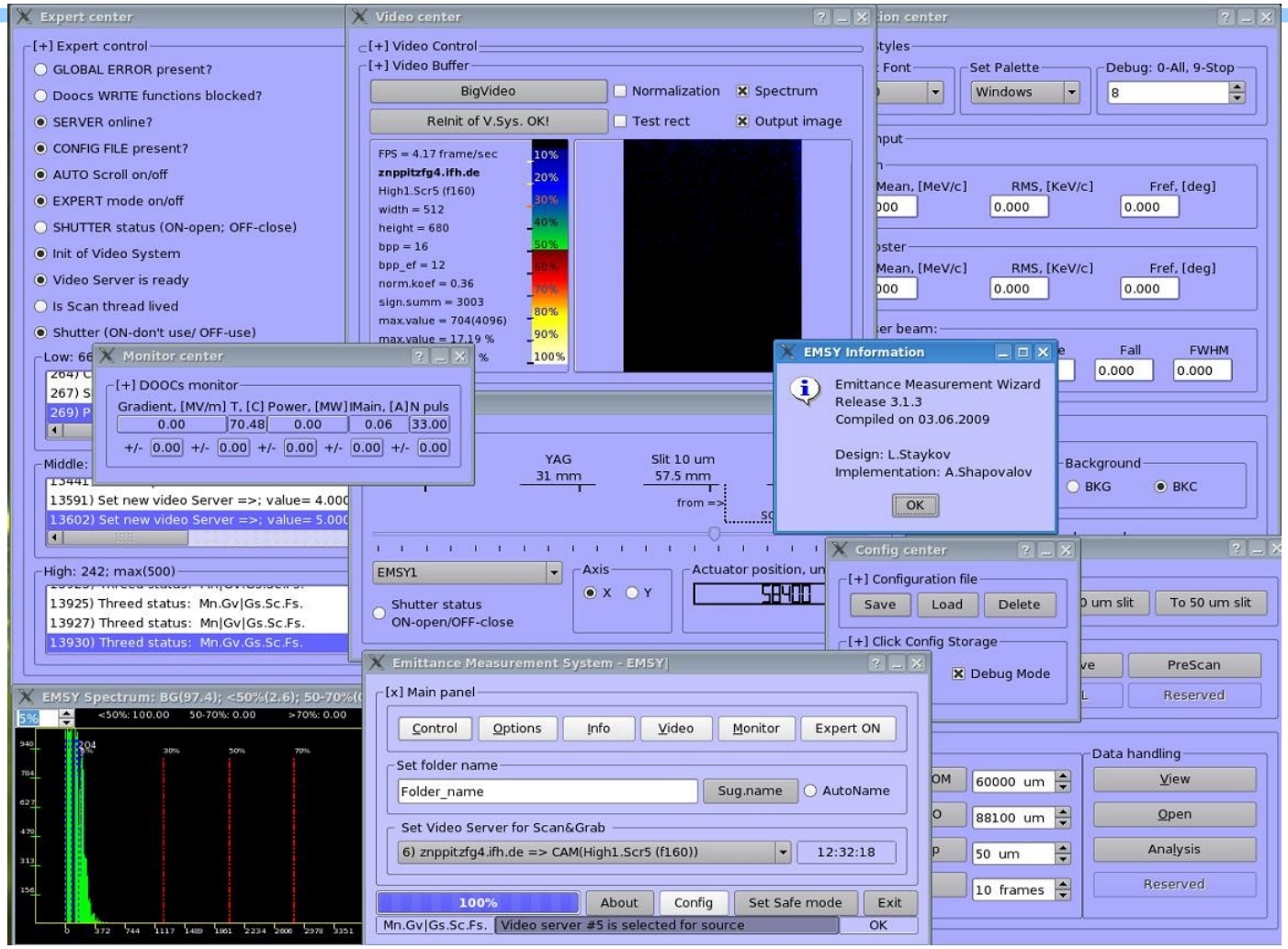


EMWiz

Final emittance value and intermediate data can be measured much quicker, requiring less user-interaction and attention by operators

- user-friendly multiple window interface

- supports semi- and full-automatic operation while control by hand is retained



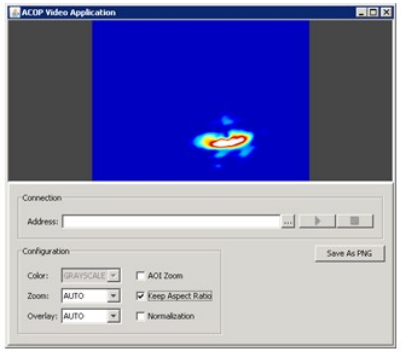
PITZ Emittance Measurement Wizard

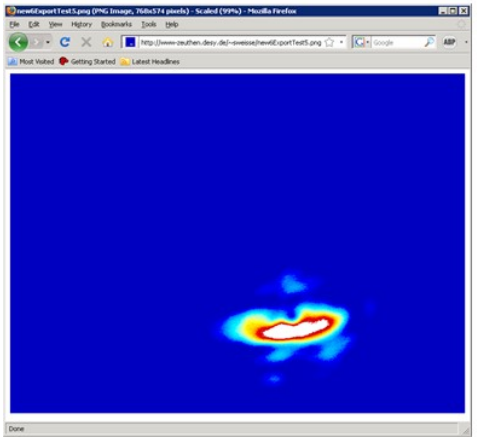
Performance Gain

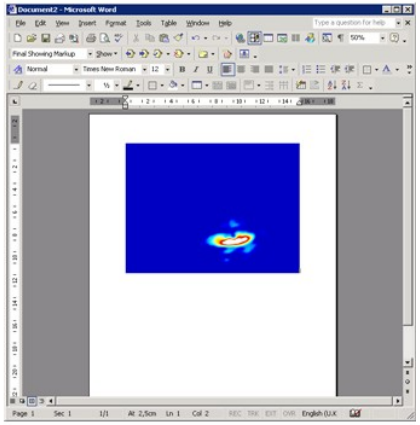
- effort has been made to provide strong figures
- on *clean* Gigabit Ethernet network installation
  - at least **30** MB/s of throughput, also verified to work with multicasting
  - video frame size from few kB to **3** MB verified
  - frame rate from 1 Hz to **30** Hz
  - all that with less than 1 frame drop per minute
- on tuned state-of-the-art setup
  - 50 MB/s, 45 frames per second with less than 1 framedrop per minute
  - these rates cannot be guaranteed, though (yet?)
- our target is nonetheless still higher, work in progress

Standard Image Format PNG

- storing of video images/image sequences to disk and loading them back in
- past: proprietary, loss of metainformation, different styles for grayscale / colour
- PNG was chosen as format (also TIFF and JPEG were considered closer)
- flexible specification created
  - exported to png (postprocessed for presentation)
  - saved as png (no postprocessing, lossless 100% also header and metadata)
- well-defined container for image sequences: PNGZIP (own development) is designated







Sites of Operation

- PITZ
- PETRA III pre-accelerator control
- HASYLAB (Hamburg)
  - PETRA III-based experiments
- EMBL Hamburg
  - transition to VSv3 is planned
  (VSv2 was used on earlier installations)
- *You?*

Perspective

- Already in the pipeline
  - finishing of core components of VSv3 (Core Provider, Video Service)
  - evaluation of XMP as metadata saving standard for PNG file
  - finishing PNGZIP image sequences standard document and reference implementation for C++ / Java
  - move PITZ to full VSv3 setup on server-side

- Planned
  - unleashing the full power of Gigabit Ethernet for video transmission
  - continue to move client-side to Java
  - provide flexible, user-friendly APIs on many platforms
  - GenICam SGP component?

Thank you for your attention!

Questions?

Comments?

stefan.weisse@desy.de
david.melkumyan@desy.de
philip.duval@desy.de

http://tine.desy.de