TINE Video System

Short Status Report

Jai/Pulnix GigE cameras for Laser Profile Readout

- cameras with removable microlenses and coverplate above CCD sensor (better profile, hope: no “burning” by PITZ laser light)

- Jai/Pulnix RM1405GE
  - 1392x1040 1/2” CCD sensor, 10bits per pixel, GigE interface
- Jai/Pulnix RM2040GE
  - 1600x1200 1” CCD sensor, 12 bits per pixel, GigE interface
- JAI tools (configure and test cameras) in early state, not very stable but was getting better in the last months
- documentation for these cameras inferior

TINE Video System

Short Status Report

Jai/Pulnix GigE cameras for Laser Profile Readout

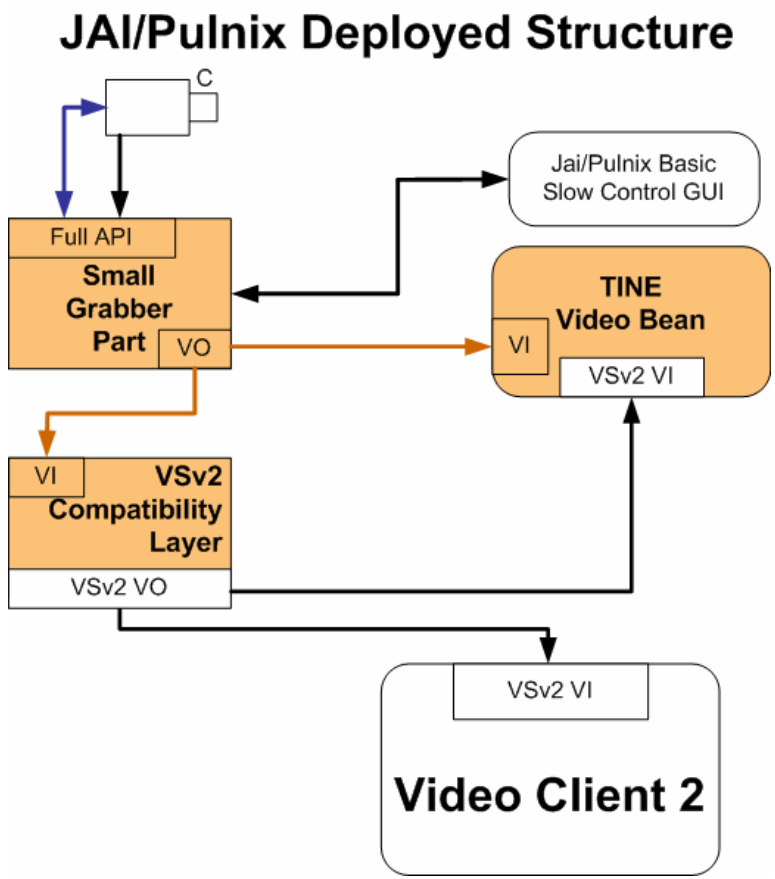
- Software development
  - JAI SDK
    - ~ 100% gen<i>cam
    - free of charge
    - is in early state (but already usable via MS Visual C++)
  - SGP was written to control and readoutJAI/Pulnix cameras using JAI SDK
  - nice side effect
    - Prosilica GigE and Basler GigE are also gen<i>cam compilant
    - Tests performed. Basically (some minor issues left) they’re working with this server!
    - Maybe in 1 or 2 years one could really have one SDK that works with all gen<i>cam compatible cameras (huge step for easy seamless software integration)

TINE Video System

Short Status Report

Jai/Pulnix GigE cameras for Laser Profile Readout

- Software deployment:
  - RM1405GE is now installed in PITZ tunnel
  - dedicated GigE network is used to deliver Ethernet packets from camera to framegrabber PC
  - SGP for Jai/Pulnix is used for slow control and image readout
  - GS2CompatLayer (early version of CoreProvider) takes raw uncompressed frames, preprocesses and sends to VSV2 clients (like Video Client 2) via PITZ GigE network



TINE Video System

Short Status Report

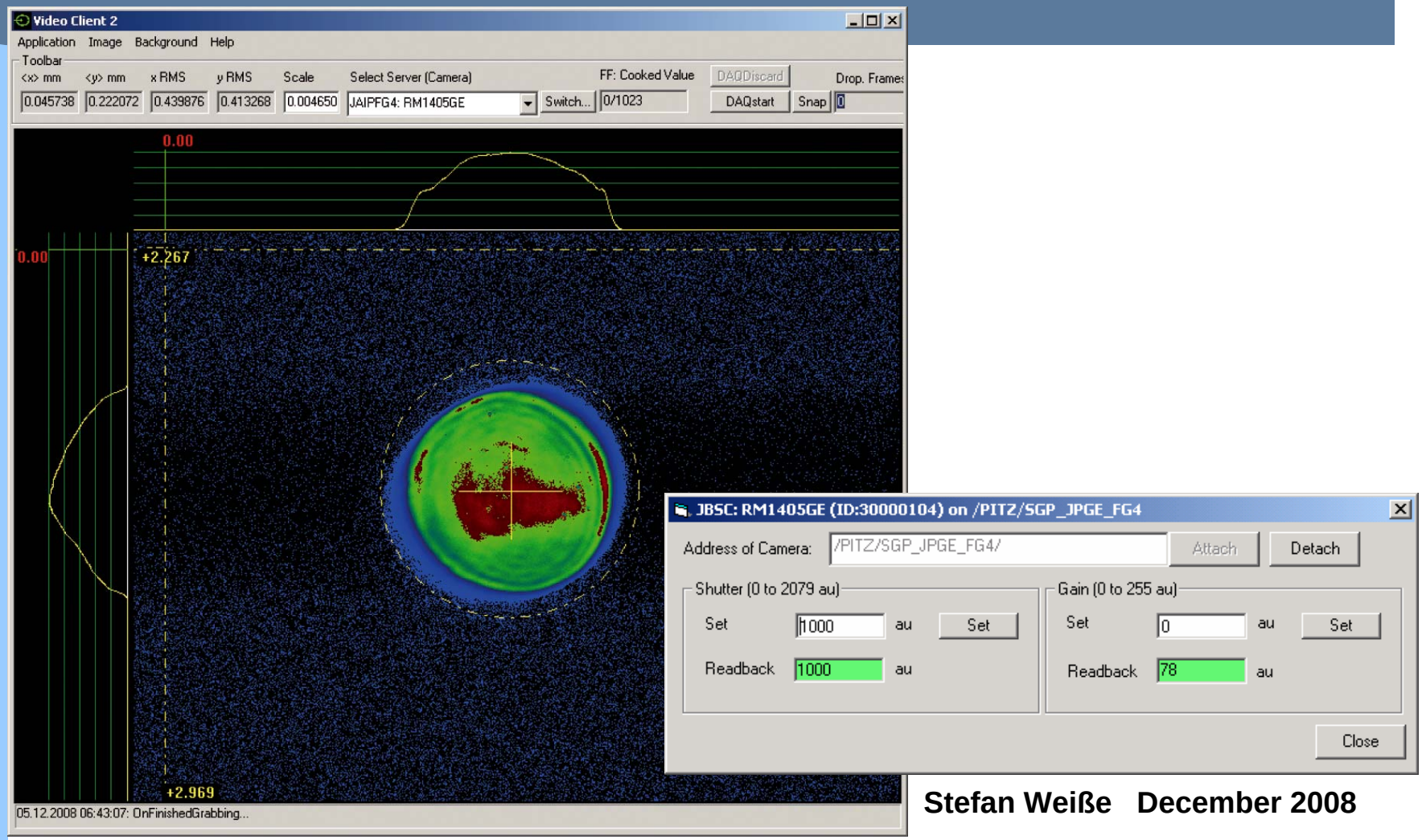
Jai/Pulnix GigE cameras for Laser Profile Readout

- Software deployment: facts and results
  - TINE shm transfer of huge bandwidth works fine
  - high bandwidth, high resolution: 1392x1040x2 bytes x 10 Hz
    ~ 29 MB/sec
  - TCP socket transfer of 1392x1040x2 bytes x 10 Hz to VS v2 clients works, if no CPU bottleneck is hit then at full speed in near realtime (only very little delay) with no losses
  - COTS (Common Off-The-Shelf) hardware:
    - Windows XP Pro Server: Core2Duo 1,83 GHz (first and slowest Core2Duo from 2007), 2x Gigabit NIC (2x PCIe)
    - Windows XP Pro Client(s): Pentium IV 2,4 GHz or more (non-HT) as client running Video Client 2 is possible (if constraints are maintained very easily possible)

TINE Video System

Short Status Report

Jai/Pulnix GigE cameras for Laser Profile Readout



TINE Video System

Short Status Report

VSV3 <-> Labview building blocks

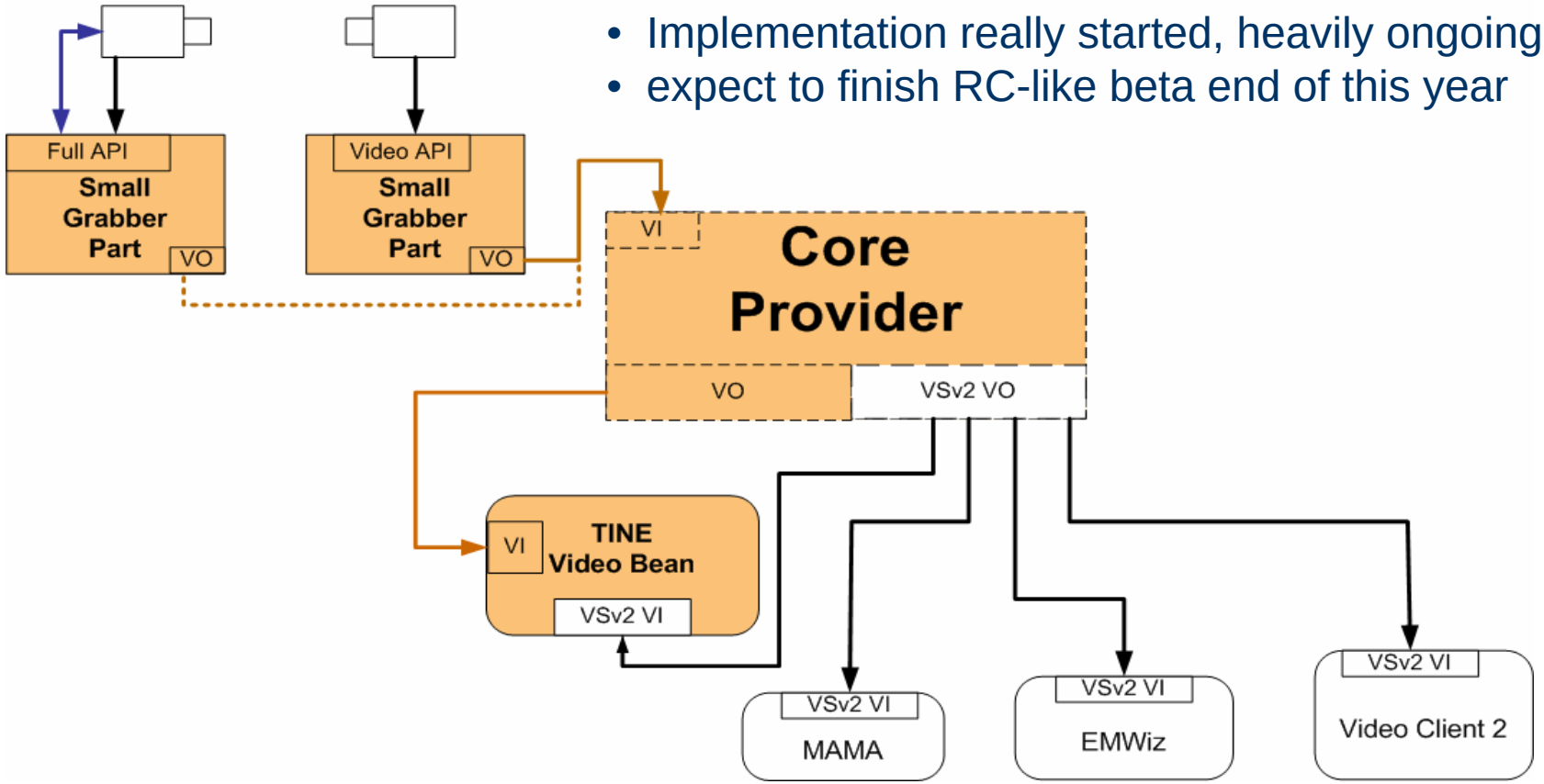
- Labview<->TINE VS v3 was open point
  (Mario DiCastro/EMBL Hamburg and Stefan Weisse)

- Three building blocks
  - **bring GigE Video (like Prosilica camera) control and readout directly to Labview**
  - **Create component in order to use Labview as an Image Provider for VSv3 (send out CF_IMAGE stream(s)**
  - Create component in order to receive and process images from VSV3 sources (like SGP, CoreProvider) to Labview

TINE Video System

Short Status Report

CoreProvider



- Implementation really started, heavily ongoing
- expect to finish RC-like beta end of this year

TINE Video System

Short Status Report

CoreProvider Functionality

- Input: raw CF_IMAGE streams from front-end TINE servers (e.g. **S**mall **G**rabber **P**arts (**SGP**))
- Optional: preprocessing of image data
  - software AOI cutout, shrinking of image dimensions (saving bandwidth)
  - Attaching scale factors to image (apply pixel to mm ratio)
  - Orientation change (adjust image orientation)
- Output:
  - VSv3 (CF_IMAGE) TINE feed (might be lossy JPEG compressed)
  - VSv3 (CF_IMAGE) TCP socket feed (lossless, guaranteed delivery for sequences)
  - VSv2 feed (if possible, e.g. colour can't be delivered this way)
    - TINE byte array feed and TCP socket feed

TINE Video System

Short Status Report

Generic Camera Slow Control (TINE) Properties

- "Problem": each distinct camera model has a different way to access almost identical set of slow control properties
  - e.g. gain, shutter speed, trigger mode, white balance, blacklevel, white clip level
- For users, operators, software developers and clients the best would be to have a somehow "fixed" (TINE) property interface that provides all information and gives opportunity to control "any" camera type
- Design guideline document finished

TINE Video System

Short Status Report

Generic Camera Slow Control (TINE) Properties

- Soon to do
  - do first test implementation using Prosilica API (end of this/beginning of next year)
  - adjust more SGPs for VSv3 to provide generic slow control properties
  - change clients or create new clients that use generic slow control properties instead of proprietary slow control properties

TINE Video System

Short Status Report

SVN repository discussion and change

- Experience: svn trunk is not current development head but will be used for stable releases, no unstable development code possible to store there (for collaborative development)
- but Videobean code should be collaboratively developed and tested in a wider area before it is released and semiautomatically deployed to DESY site for production
- Development-tree with testing-release of acopbeans.jar (VideoBean inside) is created and will be used soon

TINE Video System

Short Status Report

Future Outlook

- work hand-in-hand with Cosylab to create JAVA-based video analysis application
- Finish vsv3 core parts (CoreProvider, VideoService)
- rollout (even more) core parts at PITZ
- Test installations / rollout at DESY2, EMBL, Petra 3