1. Welcome and Introduction (Bozhon & Fritz Dierks, Basler)

2. Agenda Review (Fritz Dierks, Basler)

3. Homework Status/Voting Members (Fritz Dierks, Basler)
   - Allied Vision – GenTL Validation Framework, GenTL Certification, Embedded
   - Basler – GenICam 3.2, Embedded, VS 2017 support
   - Baumer – GenTL Producer Framework
   - BitFlow – CLProtocol
   - MATRIX VISION – GenICam 3.2, PFNC
   - Matrox – SFNC 2.5, GenDC, PFNC
   - MVTec – GenICam 3.2, PFNC, GenTL Producer Framework, Administration
   - NI – PCLint
   - Pleora – PFNC image generator
   - SICK – GenICam SFNC
   - Silicon Software – Processing Results, GenICam/OPC UA
   - STEMMER IMAGING – C-Bindings, Python bindings, GenCP, GenTL
   - Teledyne DALSA – GenICam device validation

4. Chair Election (led by Bob McCurrach, AIA)
   - Election of Chair and Vice-Chairs for next 3-years period 2019-2022
   - Elected (unanimously and each without any objection):
     - Fritz Dierks from Basler as chair of the GenICam Standard Group
     - Stéphane Maurice from Matrox Imaging as vice-chair (SFNC)
     - Rupert Stelz from STEMMER IMAGING as vice-chair (GenTL)
     - Christoph Zierl from MVTec Software as vice-chair (Marketing&Operations)

5. GenApi (Fritz Dierks, Basler)
   - GenApi reference implementation v3.2
     - #2063, #2064, #2065: Fix errors reported by sanitizer tools
       - Done
     - #1956: Merged _dev Branches
       - #1946, #1951, #1739, #1618: done
       - #1881, #1823, #1593, #1991: in progress (HOMEWORK!)
       - Release candidate planned soon within Q2/2019
   - GenApi (new) tickets
     - #2050: Crash loading feature set
       - Windows -> Linux issue
     - #2055: Add virtual destructors for IFirmware* interfaces
     - #2056: Problems with statical linking
     - #2062: Don’t deregister callbacks while they are firing
- Fix by documentation
- PCLint for GenICam (Katie Ensign, NI)
  - Enable Linting in CMakeLists.txt as a prebuild step
  - Does it affect the build system? May need additional Python modules.
- Python Bindings (Kazunari Kudo, STEMMER IMAGING)
  - Requirements:
    - Python 3.4 – 3.7
    - 64-bit versions of Linux, macOS & Windows
  - Included in GenICam build system
  - Easy install and distribution
  - Add new Trac component for Python bindings
- ProcessingResults Handling (Andreas Beyer, Silicon Software)
  - Current state
    - No semantics for arbitrary data structures
  - Benefit of standardization
    - Producer <-> Consumer
    - Embedded devices
    - Interfacing with OPC-UA
  - “Extended chunk description” enabling complex data within chunk
    - Use ChunkParser to decode it on host side
  - Examples: ConfidenceFloat, BlobData
  - Basic principles:
    - Categories describe structure
    - Collections are formed by nested Categories
    - Selectors indicate that the category holds iterable content
  - Naming conventions:
    - Structure buffer as you like
    - Describe structure in category “ProcessingResults” which is a feature of “DeviceControl”
    - Announce presence “ProcessingResults”-Chunk as feature of “ChunkSelector”
  - Next steps
    - Extension of ChunkSelector
    - Protected category name
    - Standardized terminology for frequently used data elements
- GenApi C-Bindings (Sascha Dorenbeck, STEMMER IMAGING)
  - Use cases
    - Stable ABI
    - Enable dynamic loading (DLL so)
    - Enable “stacking” (use different GenApi versions in one process, make “glue” possible)
  - Current state
    - Merging into trunk is in progress
    - Completed NodeMap features (incl. polling, loading/saving, chunks, events)
    - Completed node features
Future steps
- Better integrate with C++ GenApi
- Header-only C++ Wrapper
- Review current state

GenICam Device Validation (Eric Bourbonnais, Teledyne DALSA)
- GigE Vision TC would like to validate GenICam compliancy as USB3 Vision does within the U3V certification tool

What would be needed?
- New requirements for GenICam compliancy
- Certification code to test the requirements
- Then, GEV spec could make it mandatory to be GenICam compliant

GEV requirements for GenAPI-related compliancy
- MUST include a “Root” node with an ICategory interface. It MUST also be marked as standard.
- MUST include a “TLParamsLocked” with an IInteger interface. This node MUST NOT be streamable. It MUST also be marked as standard.
- MUST include a “Device” node with an IDevice interface. This node MUST NOT be streamable. It MUST also be marked as standard.
- Readable features MUST be valid.
- Readable features MUST stay valid when related nodes are changes (in particular selectors).
- When a Feature is readable and writable, it MUST be possible to write back the current node value. This does not apply to nodes with an ICommand interface.
- Swissknifes equations MUST be properly formed.
- Non-features nodes MUST NOT be marked as streamable.
- Streamable features MUST accept and correct invalid values when the validation option is false
- All features that require persistence MUST be marked as streamable.

GEV requirements for SFNC-related compliancy
- SFNC features MUST be marked as standard.
- Standard features MUST use the interface specified in SFNC.
- Standard features names and enumeration entries names MUST be defined in a SFNC specification.
- Non-features nodes MUST NOT be marked as standard with the exception of the “Device” port.
- Non-standard features MUST NOT include standard enumeration entries.
- Standard features “DeviceRegisterStreamingStart” and “DeviceRegisterStreamingEnd” MUST be present when at least one feature is streamable.

GEV requirements for GenDC-related compliancy are part of GenDC spec

GenICam Device Validation
- Source in GenICam repository
- Tested with Camera Link
- Based on Python bindings
- Virtual SFNC device

Conclusion:
- Adopt new requirements (as part of GenICam procedures)
- Release tests for device validation (tag in SVN)

Working Group needed
- At least 4 companies
- Also: Test and feedback on dev branch!
- Transition period for introducing, in particular also in case of new TL version!

6. GenICam for Embedded (Werner Feith, Euresys & Thomas Lück, Allied Vision)
   - Status from EMVA IEVIS (Industrial Embedded Vision Interface Standard)
     - Started efforts to define modules within kernel mode using media control framework
     - GenTL producer as main kernel to user interface
     - Communication with MIPI about membership
   - Three layers:
     - User layer: GenICam
     - Kernel layer: Enable easy adaption of embedded boards
     - Hardware layer: Enhancing SLVS-EC /MIPI CSI-2.0-PHY
   - Next steps:
     - Communication with V4L group
     - Collaboration with MIPI group
     - Alignment on name for the standard
     - White paper about concepts and goals

7. GenICam for Embedded – GenTL to V4L Adapter (Fritz Dierks, Basler)
   - Two use cases
     - Video System – ISP on Camera Module
     - Video System – Using Processor’s ISP
   - Emulating V4L/gstreamer from GenTL
     - Need for a GenTL to V4L converter
     - Also useful for GEV and U3V
   - First experiments with prototype
     - Start GenTL-V4L bridge with V4L loopback together with camera GenTL Producer
     - Use e.g. vlc or gstreamer to see the live images
   - Next steps
     - Bind standard V4L parameters to GenTL
     - Handle more than one device
     - Zero copy using DMAbuf
     - Start/stop capturing

8. GenICam & OPC Vision (Ralf Lay, Silicon Software)
   - Idea: Combine classical GenICam components with IoT / Industry 4.0
     - Factory layer
- Application layer
- Device layer
- First step: Allow direct access to the camera by a generic OPC-UA server
- Proof of concept based on GenTL/Python/Harvester and FreeOPCUA
  - Successful connection between OPC-UA and GenICam
  - Several limitations related to used OPC-UA server
- Proposal: Combine efforts with the VDMA OPC Vision core working group
  - See OPC Vision session on Thursday, March 28th during IVSM 2019 Suzhou

9. GenTL (Rupert Stelz, STEMMER IMAGING)
- GenTL v1.6
  - Stacked info function
    - DSGetBufferInfoStacked
    - DSGetBufferPartInfoStacked
  - Release is planned until the next IVSM meeting in October 2019
- GenDC support
  - Wording: buffer, subbuffer/segments and composites
    - Segments, transported on flows
    - Flow sets, transported on streams
  - Flowset/composite buffer
    - Flow information should be available to the Producer via bootstrap, otherwise Producer must provide nodemap features
  - Re-use existing functions (automatic split into segments if needed):
    - DSAnnounceBuffer
    - DSAllocAndAnnounceBuffer
  - New function:
    - DSAnnounceCompositeBuffer
    - Additional functions to inquire number of flows/segments
  - New payload format for GenDC
  - Support of early processing should be possible, but not planned for v1.6 release
- GenTL Validation Framework (Tom Kirchner, Allied Vision)
  - Cleanup of sources
  - Refinement of release process, see #2057
  - Integration of certification, see #2058
    - Test result as a self-enclosed certificate using given key pair
    - Integrate the core functions into the existing interface
  - Integration of external tests, see #2059
    - Add custom test suites easily
    - More flexible test suite interface
  - Resolve the UI state, see #2060
- GenTL Producer Framework (Roman Moie, MVTec)
  - Prepare v2.0
    - Release candidate is ready, announcement on mailing list soon
    - Some weeks review/testing period
    - Then start ballot
- New tickets for future releases v2.1 or v3.0
  - #1889 (expose buffer nodemap)
  - #1988 (introduce GenTL Core mode allowing direct stream control)
  - Still open tickets: #1891, #1893, #1928, #2071, #2072, #2073

10. GenTL SFNC (Mattias Johannesson, SICK)
   - Already agreed proposals for v1.2
     - Clarify events (#1305)
     - PacketSize renegotiation (#1942)
     - Feature persistence (#1985)
   - Tracking more suggestions for v1.2, see #1750
   - Release v1.2 planned after GenTL v1.6 update (GenDC)
     - Event NewBufferData
     - Support of GenTL flows/segments (GenDC)

11. GenCP (Rupert Stelz, STEMMER IMAGING)
   - GenCP 1.3 just released
     - Stacked read/write commands and acknowledges
     - Clarification about the meaning of existing DeviceVersion register
     - Minor changes

12. PFNC (Uwe Hagmaier, MATRIX VISION)
   - PFNC v2.3 just released
     - DataX formats
     - Several new DataX Pixel formats
   - Polarizer pixel formats
     - Fully/partly- or non-self-describing?
     - Problem: Lots of pixel formats
     - Proposal (for PFNC 2.4 with additional new polarizer values)
       - Keep it self-describing
       - List used angles as part of the PFNC format name once and refer to by index, e.g., POLARIZED_0_45_90_135_xxxx (e.g., xxxx = 2130 as indices to 0, 45, 90, 135)
       - Fully describe the matrix in case of color+polarized, e.g., POLARIZED_CFA_0_45_90_135_xxxxxxxxxxxxxx_yyyyyyyyyyyyy (e.g., xxxxxxxxxxxxxxxx=2121303021213030_RRGGRGGGGBBBGGG)
       - For square filter patterns with width=height keep CFA notation POLARIZED_CFA_xxxx
       - Short-form color filter would also be possible for standard cases, e.g., POLARIZED_CFA_0_45_90_135_2130_RGGB
       - Is this sufficient for the moment? Start HOMEWORK ticket
   - Support of polarizer pixel formats should be the “last complex pixel format” covered by PFNC as it is today
   - Start working group to discuss future complex data types, in particular supporting hyperspectral sensors
13. **GenDC (Stephane Maurice, Matrox Imaging)**
   - GenDC v1.0 released in December 2018
   - First prototypical implementation with GEV server and also GenTL Producer Framework already successful
   - Some new clarifications, in particular regarding the requirements
   - Further Clarifications needed
     - Flow table endianness
     - Handling of GenDC metadata Components including GenICam Chunk in a container
   - Addition of XML Metadata needed
     - Support fully self-described chunk data in a Container

14. **SFNC (Stephane Maurice, Matrox Imaging)**
   - SFNC 2.5 Draft1 already available
   - GenDC related features
     - Add value “GenDC” for feature TestPayloadFormatMode
     - ComponentIdValue (predefined value for all known component types)
     - New feature GroupIdValue
     - New feature GenDCDescriptor (IRegister)
     - New feature GenDCFlowMappingTable (IRegister)
     - New feature GenDCStreamingMode
       - Vote 6:4 for value “Automatic” instead of “Mixed”
     - New feature GenDCStreamingStatus
       - GenDCStreamingStatus (Off, On)
   - Open issue: Define ways to identify conditional mandatory features
   - Clarify SFNC model for components with different resolutions
   - Release Candidate available soon

15. **Marketing & Operations (Christoph Zierl, MVTec)**
   - Update on membership: 15 new member companies since last meeting
   - Currently 13 contributing members, see above
   - Roadmap
     - Planned updated content of next GenICam Package Release end of Q2/2019
       - GenICam reference implementation 3.2 (in progress)
       - GenApi Standard 2.1.1
       - SFNC 2.5 (start ballot right after meeting)
       - PFNC 2.3
       - GenTL 1.5
       - GenTL Producer Framework 2.0 (not included in public download) (in progress)
       - GenTL Validation Framework 1.5.x (not included in public download)
       - GenCP 1.3
       - GenDC 1.0
       - CLProtocol 1.2
       - License 1.7 (in progress)
• www.genicam.org
  ▪ Minor changes
  ▪ Regular updates on “News” page
• New homework packages:
  ▪ Publish official GenICam introduction based on new CVSM training presentations
  ▪ Create first version of a GenICam FAQ
  ▪ Explain the standard(s) and its compliance rules on a separate web page
• Clarification of rules for distributing GenICam runtime/SDK files (Chendra Hadi Suryanto, Omron Sentech)
  ▪ Users are confused with GenICam license information files
    ➢ Different meaning of the terms Development/Runtime
    ➢ Suggestion: Update text in License_ReadMe.txt
    ➢ Rename official Windows download files like the Linux ones
    ➢ Make clear that GenICam_License_yyyymmdd.pdf must be included when distributing the reference implementation
  ▪ New homework ticket to actually integrate the proposed changes

16. Homework session (Fritz Dierks, Basler)
• Go through homework list/items
• Next meeting:
  ▪ 2019, October 7-8, hosted by Lakesight in Stresa, Italy