JUVIL: A new innovative software framework for data analysis of JET imaging systems intended for the study of plasma physics and machine operational safety

Valentina Huber [a,*], Alexander Huber [b], David Kinna [c], Guy Matthews [c], Itziar Balboa [c], Adrian Capel [c], Paul McCullen [c], Philippe Mertens [b], Gennady Sergienko [b], Scott Silburn [c], Klaus-Dieter Zastrow [c]

[a] Forschungszentrum Jülich GmbH, Supercomputing Centre, 52425 Jülich, Germany
[b] Forschungszentrum Jülich GmbH, Institut für Energie- und Klimaforschung − Plasmaphysik, Partner of the Trilateral Euregio Cluster (TEC), 52425 Jülich, Germany
[c] CCFE, Culham Science Centre, Abingdon, OX14 3DB, UK

HIGHLIGHTS

- Powerful, user-friendly and robust framework for fast data visualization and advanced analysis of all types of imaging data.
- Modular object-oriented components provide interfaces to access video data and automatic post-processing.
- Highly configurable and can be easily extended and adapted for new imaging cameras, data formats or developing applications for users own analysis.
- Tools for first wall protection purposes, analysis of protection system alarms, edge plasma physics studies, investigation and evaluation of the formation and development of hot spots.

ARTICLE INFO

ABSTRACT

A new powerful software framework JUVIL (JET Users Video Imaging Library) has been developed and successfully installed at JET for fast data visualization and advanced analysis of all types of imaging data. The JUVIL framework is based on modular object-oriented components implemented in Python to simplify work with JET scientific data. It provides standard interfaces to access video data and post-processing, which are highly configurable and can be easily extended and adapted for new data formats and imaging cameras. One of the GUI components is the video player, widely used during the last JET campaign. It displays the video data for NIR/IR/VIS cameras and automatically carries out the post-processing (image rotation, data format conversion, scaling of non-interlaced fields to full frames).

1. Introduction

Imaging systems are an indispensable technique for successful plasma operation of fusion devices. At the JET tokamak, several imaging systems in the VIS/NIR/MWIR spectral ranges are used for plasma physics studies as well as for the real time overheating protection of the first wall and for live plasma monitoring during operation [1]:

- 13 near-infrared (NIR) CCD cameras routinely used for monitoring the surface temperature by the protection system installed on JET [2]
- 4 mid-wave infrared (MWIR) cameras for physics studies
- 7 scientific digital CCD cameras in near UV and VIS wavelength range for hydrogen spectroscopy (D_α, D_β, D_γ, D_ε) and for gross erosion measurements of W, Be, C materials in JET, as well as for the monitoring of emission of the seeded impurities (N, Ne, Ar) [3]
- 4 operation colour CCD cameras for the in-vessel video surveillance
- fast visible camera for the detection of destructive plasma events and plasma termination events

* Corresponding author.
E-mail address: V.Huber@fz-juelich.de (V. Huber).

The main challenge described here was to establish the optimal software tool for the quick access and analysis of all types of imaging data. The previously existing software tools (e.g. PINUP [4], IRdisp [5], various Python scripts and MATLAB programs) were
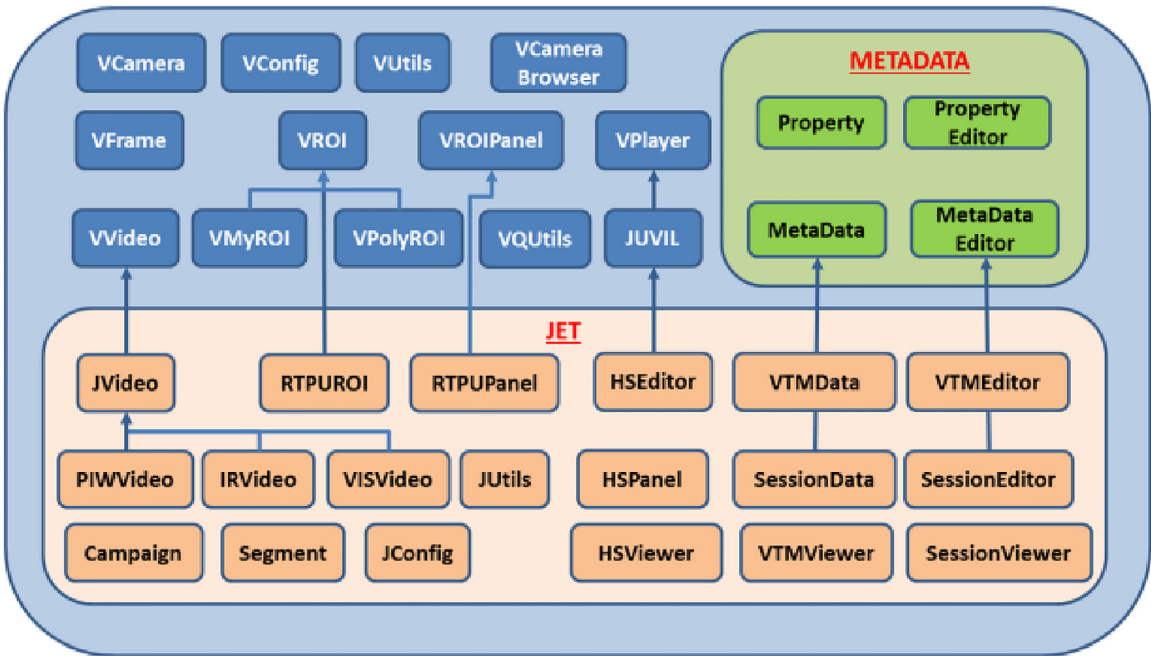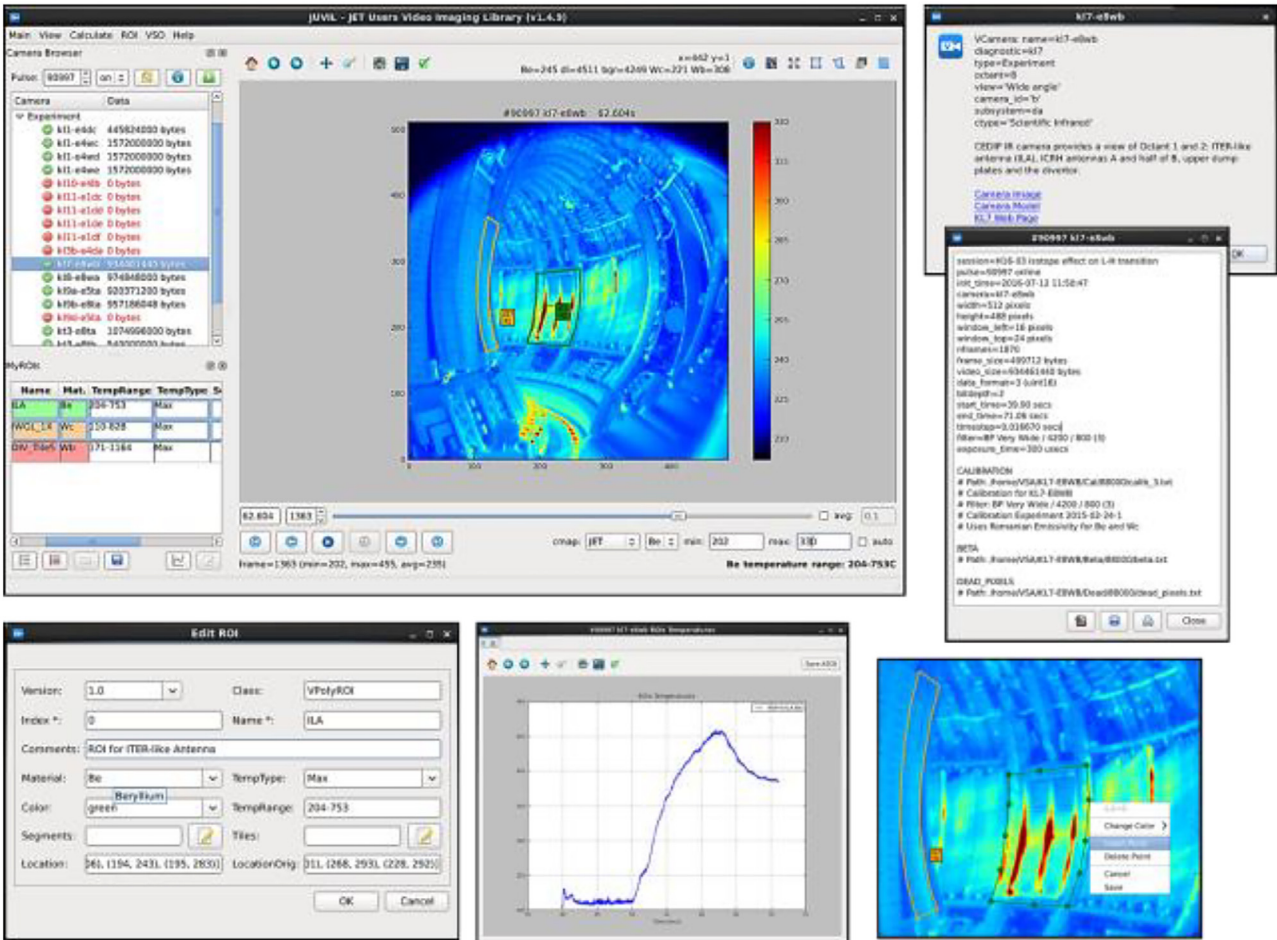

Fig. 1. JUVIL architecture.
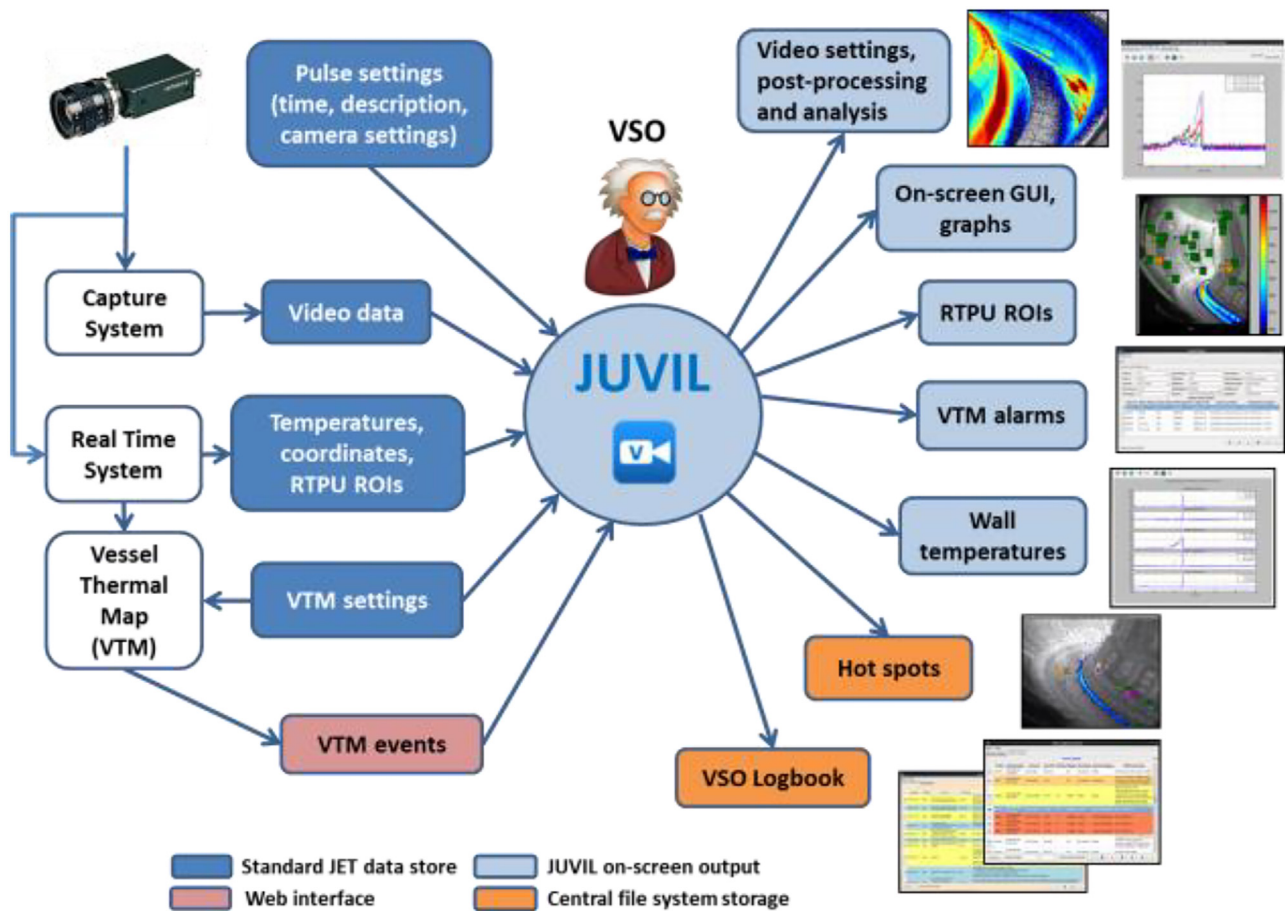

Fig. 2. JUVIL graphical interface.

Fig. 3. Viewing System Operator (VSO) Tools.

implemented using different programming languages and program logics which required from the users the knowledge of which tool should be used for a specific camera type and how to run it. The software maintenance and development were also difficult, especially because some of these programs were not well structured or poorly documented. In addition, the crucial factor is to have an open source product in order to be able to modify the code (e.g. to add some new features on users' requests or to fix the software bugs, especially those which cause a program to crash).

In order to fit all these requirements and to simplify work with JET scientific data, a new powerful, user-friendly and robust software framework *JUVIL (JET Users Video Imaging Library)* has been developed and successfully installed at JET for fast data visualization and advanced analysis of all types of imaging data. It is also successfully implemented for calibration data evaluation of all protection and scientific cameras [6,7].

2. JUVIL architecture

The JUVIL framework consists of modular object-oriented components written in Python. They provide standard interfaces to access video data and post-processing, are highly configurable and can be easily extended and adapted for new imaging cameras, data formats or developing applications for users' own analysis. There is a separation between abstract standard interfaces and site-specific implementations, e.g. JET specifics are implemented in a *jet* package (see Fig. 1). The abstract *VVideo* component provides standard functions to access video data and *JVideo* is a derived subclass of *VVideo*, which inherits all *VVideo* properties and implements these functions in order to access JET data using Python interface *getdat*

[8]. The *PIWVideo* is a subclass derived from *JVideo*, which contains some additional functions and properties for NIR protection cameras. Such object-oriented hierarchical structure provides very efficient common platform with the possibility to reuse the functionality of existing components without the need to copy the code several times and makes the software maintenance much easier. For example, in order to add new camera or data format, it is necessary to create a custom *VVideo* subclass and add it into the configuration file *VConfig*. The new subclass should implement the following *VVideo* functions: *read_metadata* to read the video metadata (geometry, number of frames, data format, etc.), *load_frames* to load a specific frame range, and optionally *load_conf* to load configuration files (calibration, flat field, background, etc.). If automatic image post-processing is required then *VConfig* file should contain the list of post-processing routines.

The package *metadata* contains the useful utilities for saving the metadata (for example, calibration settings or event descriptions) as a text file to the file system and provides the graphical editors dynamically generated on the fly in order to display and modify the metadata.

3. JUVIL functionality for the study of plasma physics

One of the GUI components is the video player, widely used during the last JET campaigns (see Fig. 2). It displays the camera browser with the list of cameras grouped into specific categories: *Protection* for NIR protection cameras, *Operation* for operation colour cameras, *Experiment* for scientific IR and VIS cameras, *Intershot* for IR post-pulse analysis cameras, and *Other* for all other types of cameras.
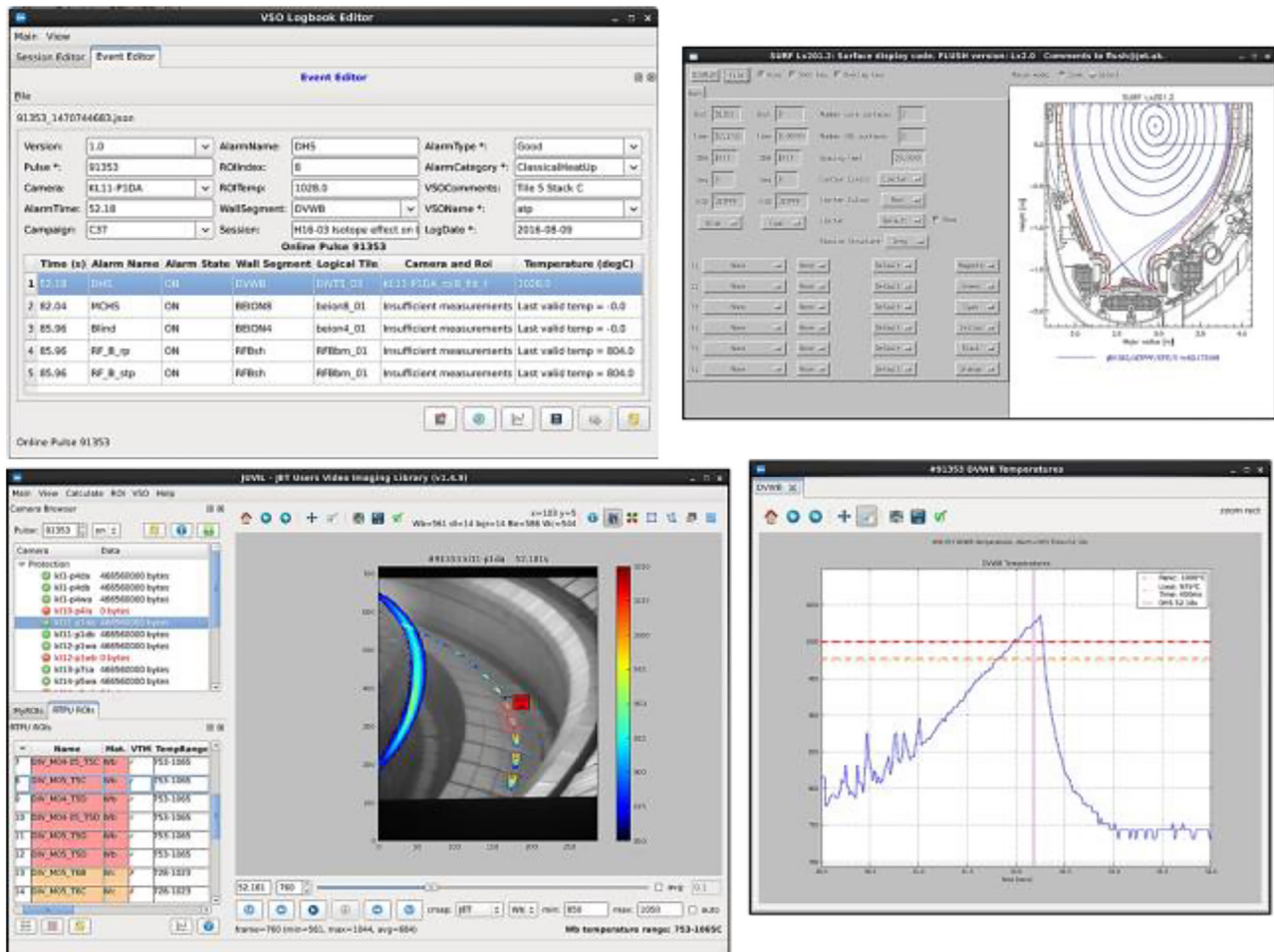
Fig. 4. VSO Logbook Editor.

In order to improve performance, the sizes of the video data for an arbitrary pulse will be refreshed only for those groups, which are expanded. If the video data for a specific camera are available JUVIL loads and displays the video using typical player functions for playing, navigation and zooming. In order to make the loading of video faster, the video data is loaded in chunks of 100 frames without waiting until the whole video will be loaded. The time of first frame is initially set around 50 s (when the probability of the plasma "active state" is relatively high) or it is synchronized to the last time of the previously loaded video in order to compare the data from different cameras at the same time period. In addition, JUVIL automatically carries out the video post-processing: image rotation in order to set a proper camera orientation and – what is also quite important – to improve the user's posture during the video observations; data format conversion (e.g. YUV 422 to RGB format for operation cameras); scaling of non-interlaced fields to full frames for protection cameras. The most relevant parameters like the number of frames to be loaded, the initial timestamp, and the list of post-processing functions for each camera, are stored in the configuration file. JUVIL also shows general information about each camera (camera type, location, description, and useful hyperlinks) and about video settings such as pulse initialisation time, experiment title, video start and end times, number of frames, frame geometry, data format, exposure time, camera filters, etc.

To convert the digital level counts of camera pixels into the real temperatures for a specific material, JUVIL subtracts the stored video background, loads the calibration file if it is available, and performs dead pixels and flat field corrections. If the video background is not found, JUVIL calculates it as the averaged frame of the first 10 video frames. It assumes thereby that the temperatures at the beginning of the pulse are not too hot. In addition, the temperatures for other materials used at JET are displayed in the top status line when a user moves a mouse over the video pixels. Furthermore, a user can select the temperature range to be shown. The temperatures below a specified minimum are hidden, and it enables to overlay the video with the camera image in order to see the distribution of temperatures on the JET vessel structure. The camera image is a frame loaded usually from the video during a plasma disruption assuming the camera was not shaking at this moment. It is preconfigured for different pulse ranges because of the camera potential movement after the next alignment, focusing or replacement.

It is also possible to specify custom Regions of Interest (ROIs) as rectangles or polygons and plot their temperatures. The *MyROIs* panel contains the list of user-defined ROIs displayed in the colours of the corresponding materials (green – for beryllium, red – for tungsten bulk, yellow – for tungsten coatings, and magenta – for digital level counts) or, optionally, in any colour from the predefined colour set. The ROIs that are selected in the list are shown on the video and unselected ROIs are hidden. There is an editor to modify ROI attributes (name, colour, material, temperature type, etc.). It is also possible to modify the shape of ROIs dynamically, e.g. insert or delete new points by pressing a key or by selecting the context-menu options, or drag existing point or the whole ROI using the mouse. The user can save the ROIs to the camera specific ROIs directory in the home file system and load them later.
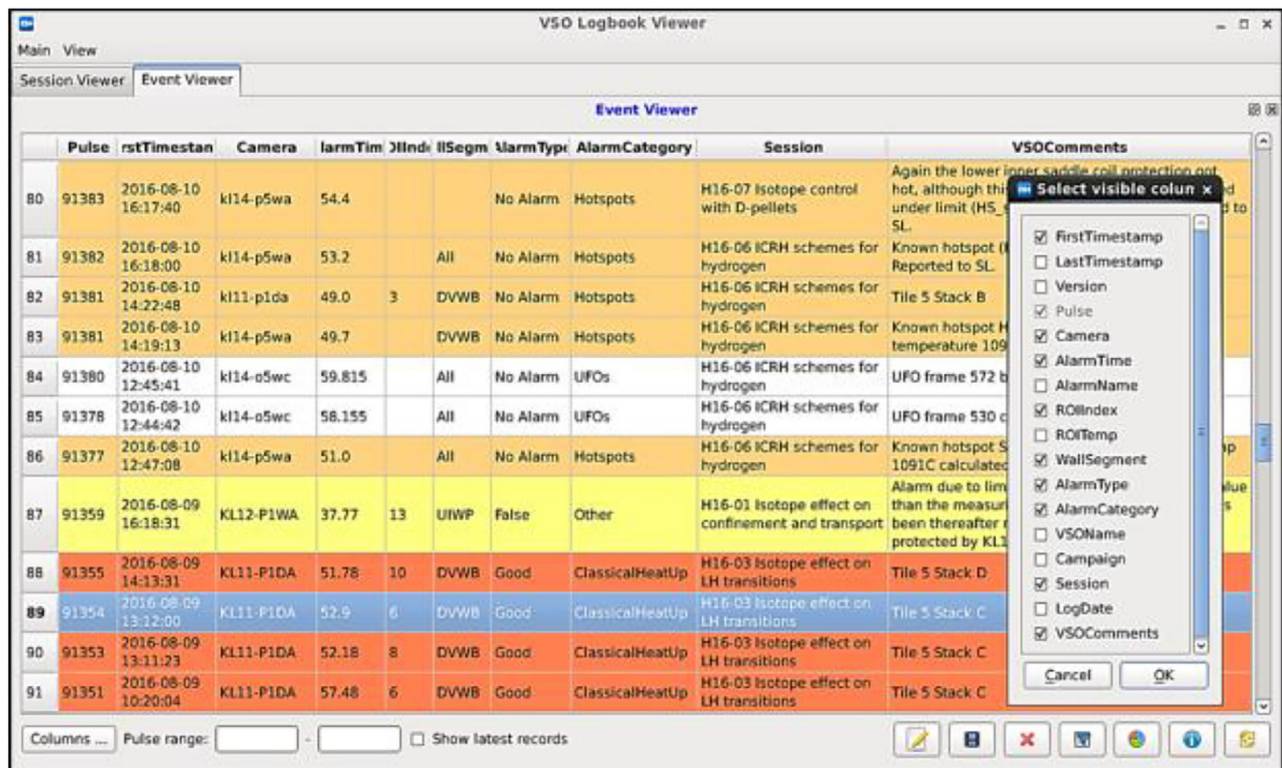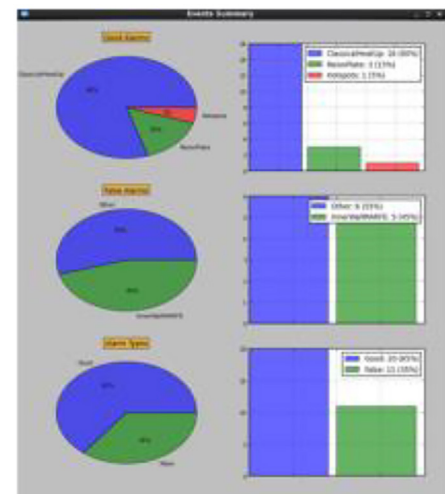
Fig. 5. VSO Logbook Viewer.

JUVIL calculates maximum, average, standard deviation profiles and time traces for selected ROIs or video frames. To make JUVIL even more user-friendly all graphical components provide various menu options, context-menus, hyperlinks and tooltips.

4. JUVIL functionality for machine operational safety

At JET, for each protection camera a Real Time Processing Unit (RTPU) [9] calculates wall temperatures for ROIs [10] and sends them to the Vessel Thermal Map (VTM) which combines data from diverse sources and raises alarms in response to overheating [11]. The viewing system operator (VSO), whose primary role is to assist the session leader, engineer in charge and scientific coordinator in the interpretation of an alarm sent by the VTM due to a protection camera, uses JUVIL for the interpretation of VTM events and also for the quick post-pulse analysis of video data required for the prepa-

ration of the next plasma pulse. In order to simplify the job of the VSO, some new features, the so-called VSO tools (see Fig. 3), were integrated into JUVIL.

4.1. VSO tools

Fig. 4 shows the *VSO Logbook Editor* for automatic loading of raised VTM events and alarms for an arbitrary pulse and storage to the VSO logbook. If a hotspot alarm is detected the Event Editor selects it in the list of VTM events and displays the corresponding camera video at the time when the VTM alarm occurs.

It also automatically loads the RTPU ROIs, selects the ROI which caused the alarm and displays its maximum temperature at the location where it was measured, sets the temperature range and overlays it with the camera image. This saves a significant amount of time, which can be used by the VSO for actual analysis of the
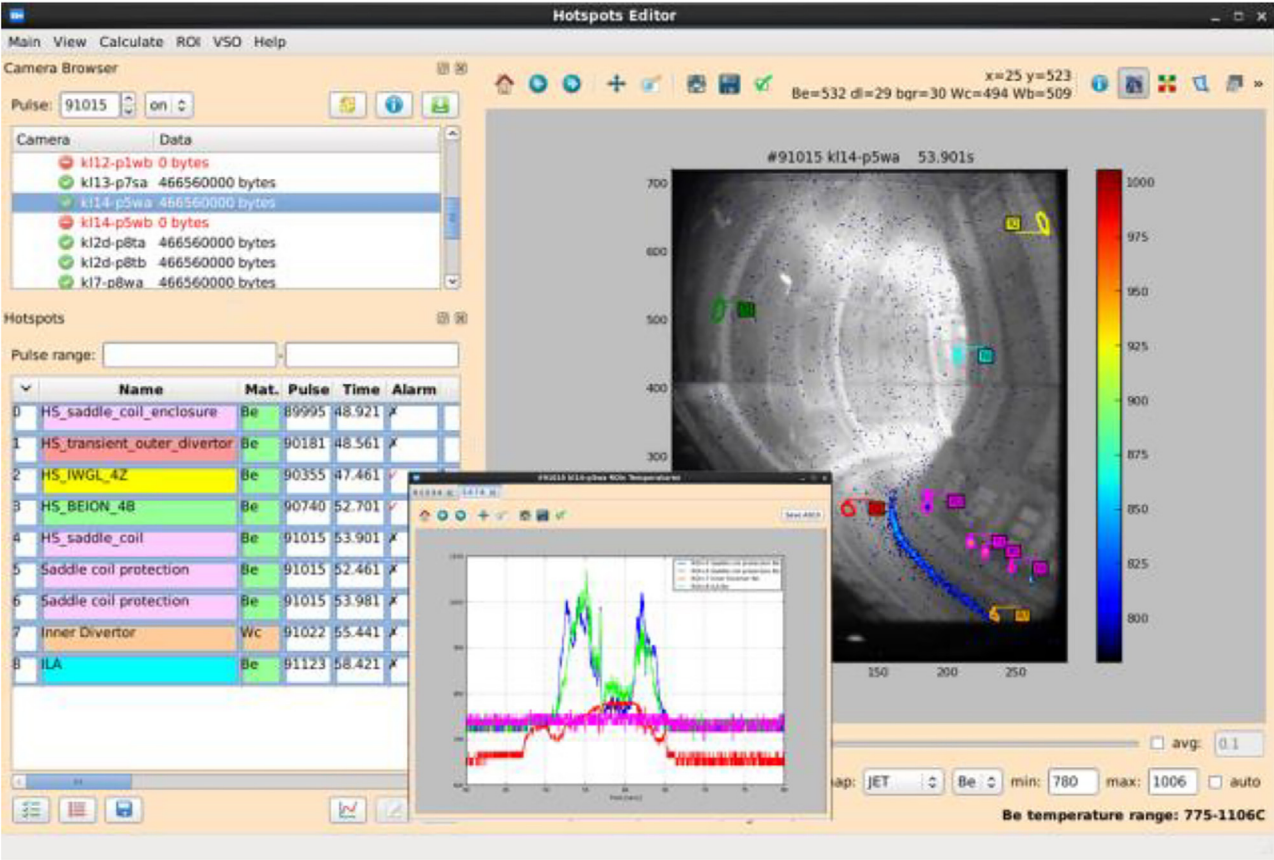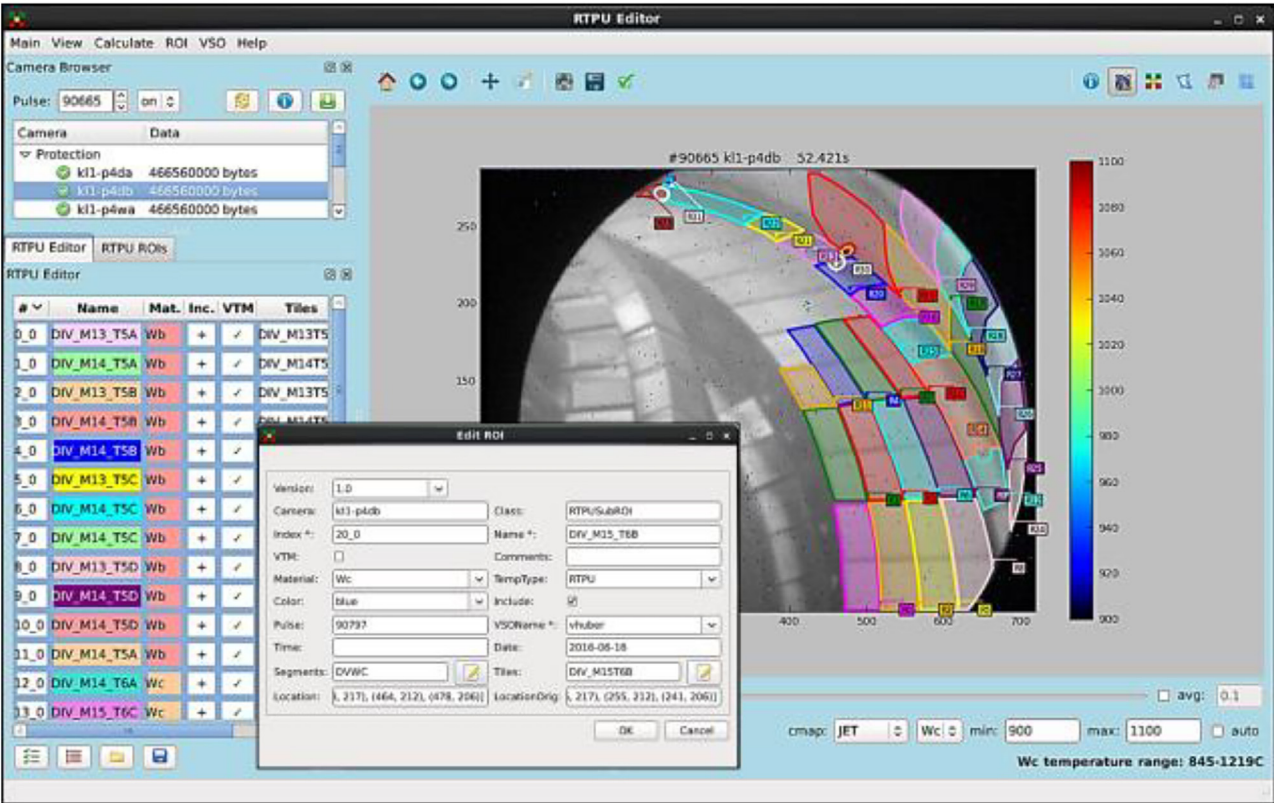
Fig. 6. Hotspot Editor.

Fig. 7. Real Time Processing Unit (RTPU) ROI Editor.

event. In addition, it is also possible to plot maximum RTPU ROI temperatures and temperatures of vessel wall segments with the corresponding VTM alarm thresholds (shown as orange and red dashed lines) as well as their assertion times and the time of the alarm (shown as magenta vertical line), and to display magnetic surfaces using the SURF program [12]. After analysis, the VSO has just to specify the type and category of the VTM event from the predefined lists in the Event Editor and to store the results into the VSO Logbook on the JET centralised file system. In addition, JUVIL enables the VSO to save general comments about a session using the *Session Editor*.

The *VSO Logbook Viewer* (see Fig. 5) displays the list of VTM events or session summaries in different colours for specific alarm types and enables sorting or filtering using different criteria (e.g. extract VTM alarms for a specific experiment or campaign, filter hot spots in a specific pulse range, etc.) as well as it enables to select visible columns. It is also possible to load the corresponding videos for selected VTM events with automatically preconfigured video settings or to display the statistics chart of the filtered VTM events. The VSO database consists of JSON (JavaScript Object Notation) files which are easy for humans to read and write and it is also easy to process them using any JSON parser.

4.2. Hotspot editor

The Hotspot Editor is a tool for the investigation and evaluation of the formation and development of *hot spots*: localized micro-regions on the surface that appear to have much higher temperature than the surrounding regions. Because these hot spots trigger VTM alarms, which can cause the protection system to stop a pulse, it is important to study the appearances of new hot spots and the conditions of disappearance of the existing hot spots, as well as their sizes, locations and temperatures. For this aim, the new component *HSEditor* was derived from JUVIL with some additional features in order to store the hotspots parameters and analyse their evolution. By loading a video the Hotspot Editor automatically loads the hot spots for a specific camera saved in the central filesystem storage into the extension of ROI panel, *HSPanel*, which displays the list of hot spots and shows their mapping by drawing the hot spot shapes on the video (see Fig. 6). The editor for the hot spot attributes enables in addition to usual ROI parameters, like location, shape and size, to set also last detected time, temperature, the flag whether it caused a VTM alarm, and optionally the shot number and time when it clearly disappeared (e.g. after the cleaning up of tiles or their replacement). It is also possible to load the corresponding video for a specific hot spot automatically or to plot hot spot temperatures for an arbitrary pulse. The Hotspot Viewer displays the catalogue of all detected hot spots and shows their evaluation. It provides also the filter options for extracting of new or already existing hot spots during a specific pulse range.

4.3. RTPU ROI editor

The previously existing software at JET for producing RTPU ROIs for the Real Time System, RoiSoft [13] was implemented in MATLAB, which required getting MATLAB licences; their number on JET is very limited. In addition, it is not so easy to use this tool because the loaded video frame is not rotated or scaled, it is also not possible to modify initial ROI attributes like the material or the name of a

ROI or, for example, to load the video frames dynamically without saving and loading them as pictures. In addition, during drawing of a ROI its neighbours are not visible and it makes ROI alignments more difficult. The new component, RTPU ROI Editor (see Fig. 7) was therefore derived from the JUVIL player to provide the full functionality for loading and analysis of camera videos and VTM events dynamically. The advantage is the possibility to map the ROI locations to any video frame and also to the locations of VTM alarms stored in the VSO Logbook in order to avoid detected hot spots or areas of bremsstrahlung and to check ROI alignments and temperatures. The RTPU ROIs require some additional parameters to be specified in comparison to usual ROIs (e.g. not all RTPU ROIs send their temperatures to the VTM, therefore each RTPU ROI should contain a flag if it is under VTM control or not). To this purpose, the RTPU Editor uses a RTPU Panel, the extension of the JUVIL ROI Panel, which provides an advanced editor to modify the RTPU ROI attributes. Additionally, the RTPU Panel saves the ROIs to the predefined directory for each protection camera on the JET centralised file system and generates the corresponding file formats for RTPU and VTM systems.

5. Conclusions and future works

The new powerful, user-friendly and robust software framework JUVIL has been developed and successfully installed for fast data visualization and advance analysis of all types of imaging data. JUVIL is now routinely used at JET for first wall protection purposes, analysis of VTM events and alarms, edge plasma physics studies and quick post-pulse analysis of data required for the preparation of the next plasma pulse. The JUVIL framework is also successfully implemented for the evaluation of the calibration data of all protection and scientific cameras at JET. The next steps for the future are the integration of old video formats, writing of temperature profiles into JET Processed Pulse Files (PPFs), automatic detection of hot spots and image corrections on camera movements, integration of CAD models, performance improvements, and installation of JUVIL at other platforms and fusion sites.

Acknowledgments

References

[1] D. Kinna, JET CDN Report H, 201010 (007).
[2] G. Arnoux, et al., Rev. Sci. Instrum. 83 (2012) 10D727.
[3] A. Huber, et al., Rev. Sci. Instrum. 83 (2012) 10D511.
[4] V. Martin, et al., Fusion Eng. Des. 86 (2011).
[5] G. Arnoux, et al., JET CDN Report E4, 200707 (001).
[6] V. Huber, et al., Rev. Sci. Instrum. 87 (2016) 11D430.
[7] I. Balboa, et al., Rev. Sci. Instrum. 87 (2016) 11D419.
[8] A.J. Capel, JET CDN Report H, 200505 (010).
[9] M. Jouve et al, preprint EFDA-JET-CP(11)06/01 (2011).
[10] D. Kinna, JET CDN Report H, 201111 (005).
[11] D. Alvares, et al., Phys. Rev. ST Accel. Beams 15 (2012).
[12] O. Ford, JET CDN Report T1, 200503 (003).
[13] S. Devaux, JET CDN Report H, 201313 (050).