Supplementary Material For:

ALISSA: an automated live-cell imaging system for signal transduction analyses

Jakub Wenus[1,*], Heiko Düssmann[1,*], Perrine Paul[2], Dimitrios Kalamatianos[2], Markus Rehm[1], Peter E. Wellstead[2], Jochen H.M. Prehn[1], and Heinrich J. Huber[1]

[1]*Department of Physiology & Medical Physics, Royal College of Surgeons in Ireland, Dublin, Ireland and* [2]*Hamilton Institute, National University of Ireland Maynooth, Maynooth, Ireland*

ALISSA interworking protocol in a nutshell

Physical communication PCS-IAS

Communication interfaces of PCS and IAS are periodically started by separate timers of the operating system that are initiated by Visual Basic and MATLAB, respectively. Synchronicity is obtained by storing and retrieving information from a common MATLAB workspace that contains common variables. This workspace is checked by both programs separately in each of the respective timer cycles. While the IAS directly accesses it, the PCS uses two MATLAB service programs for information reading ('GetVariable', 'PutWorkspaceData') that are invoked via component object model (COM) interfaces by Visual Basic. A summary of workspace variables is given in Table 1. Whenever the PCS receives image data from the microscope, each channel is stored as a bitmap in a global workspace variable and the image sequence counter is updated. At each cycle, the IAS checks this counter and processes the image upon update. Control information is exchanged via byte streams. To render the hierarchical nature of the system with PCS being the controller and IAS as service program, these byte streams are further denoted as commands when directed from PCS to IAS or events for the inverse direction. Like images, commands and events are stored in the MATLAB workspace in two separate arrays with sequence numbers identifying the most actual events and commands. When processed, events and commands are deleted by the receiving entity. Typical entries are given in Figure 1.

Logical communication PCS-IAS

Control and reporting between both entities is exercised by commands and events given in Table 2 and Table 3 for an overview.

Detailed information including concepts for further extensions can be found in a message catalog, which we provide on our web site, www.alissaproject.org, or which can be requested from the authors.

Interworking PCS: microscope (microscope drivers)

Microscope control is exercised via specific driver programs that allow updates of microscope configurations, adaptation of temporal sampling rates, image requests, and focus on regions of interest. Configurations are specific for the state of the experiment and contain all settings of the acquisition channels. Their updates are performed by the driver routine 'LoadConf,' which loads predefined configurations stored at the microscope database by their names. Sampling rates are updated by changing the frequency of calling a specialized routine 'ScanImage' which requests images from the microscope employing all channels as specified in the actual configuration. Scanning is executed by accessing the LSM5 routine 'StartScan' when the microscope is ready to perform a scan. Then the images are stored in the internal database of the microscope. Likewise, bitmap data for each acquisition channel is stored in the common PCS-IAS MATLAB workspace via call of the COM
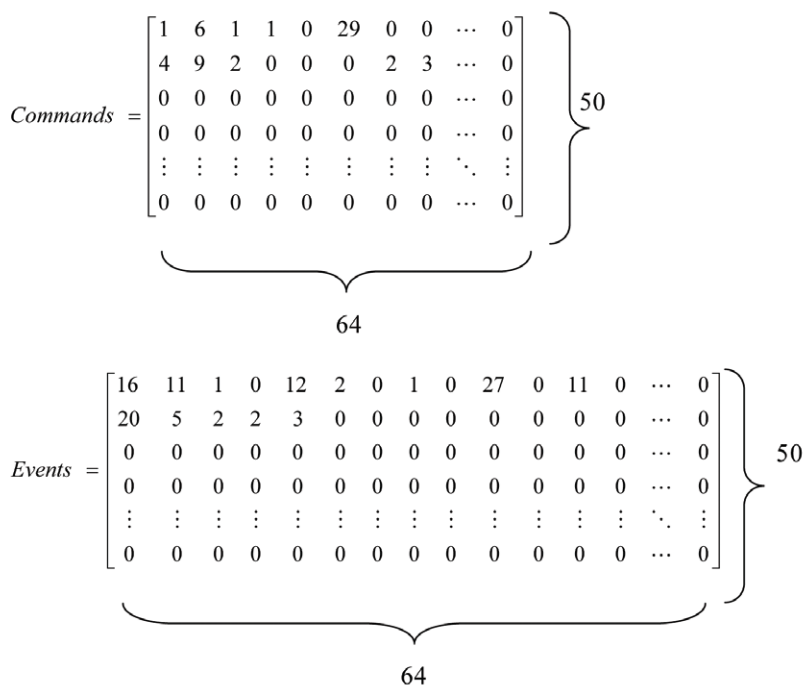
Figure 1. Typical entries in the command and event variable of the common workspace. Each command and each event contains 64 bytes. Both workspaces may contain up to 50 entries. Entries are deleted when processed.
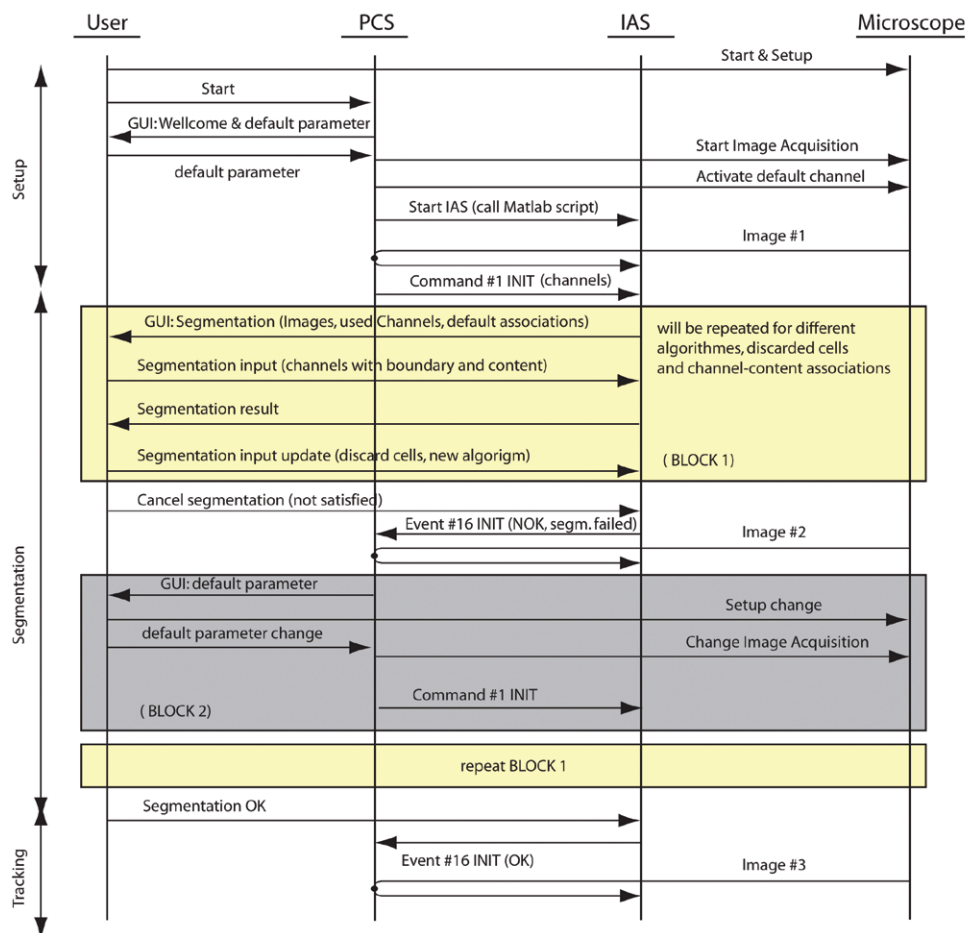
Figure 2. Initialization message flow. The arrows between PCS and IAS indicate events and commands. The arrows between the user and both entities define user interaction via user interfaces. The arrows indicating images illustrate the fact that images are requested. Details about commands and events can be found in the ALISSA message catalog under http://www.alissaproject.org.

service routine 'PutWorkspaceData.' For region scans, the driver routine 'Activate-FocusedScan' is employed. As input parameters for a square-shaped region of interest, the x and y positions of the focus center and the focus size (axis of the square region) have to be given while for an octagon-shaped region of interest, coordinates of all eight vertices are required. In the current implementation of the driver, only region scans without magnifications are possible, which are invoked by the LSM5 driver routine 'UseROIs'.

**Interworking of entities during system start-up**
Initialization needs a synchronised interplay (orchestration) between all three entities—PCS, IAS and microscope. Orchestration is illustrated in Figure 2 where we distinguish between a setup phase, a segmentation phase, and an operation phase with tracking. In the setup phase, the user sets up the sample and the microscope using the default

channel setup as given in the main text. Alternatively, the user may create a new setup. ALISSA is then started and the PCS will pop up with a graphical user interface (GUI) collecting default parameters like the initial sampling rate. The PCS will request the first image from the microscope, store it as bitmaps for all channels in the common workspace and invoke the IAS by the command INIT to initiate the user dialogs for segmentation. Any action by the PCS, including image request, is suspended until a reply from the IAS (Event INIT) is received. In the second phase, the IAS is in control of the communication with the user and will inform the PCS about the result. The IAS displays all channels of the image received from the PCS and provides a menu to select the segmentation algorithm and channels that are used as segmentation information. The user may discard cells that are incorrectly segmented and omit them from further analysis. If the user is not satisfied with the results of the chosen algorithm at all, segmentation fails

and event INIT(parameter: failed) is sent back to the PCS. This gives the user the chance to update the microscope settings. After the user acknowledges continuation, the segmentation process is repeated by resending the command INIT. If the user is satisfied this time, the IAS sends back event INIT(parameter: success). The PCS continues execution and starts regular image requests from the microscope. The IAS starts tracking and time series generation.

Cell tracking
The cell shape and position are updated in real time for each new record by applying a tracking method based on the condensation algorithm for each cell (1). The algorithm takes the previous shape and position of the cell into account to obtain robustness against the lack of information from the images. Unlike most of the available cell tracking software, which require YFP or GFP images (2), tracking is performed on the DIC channel only. The
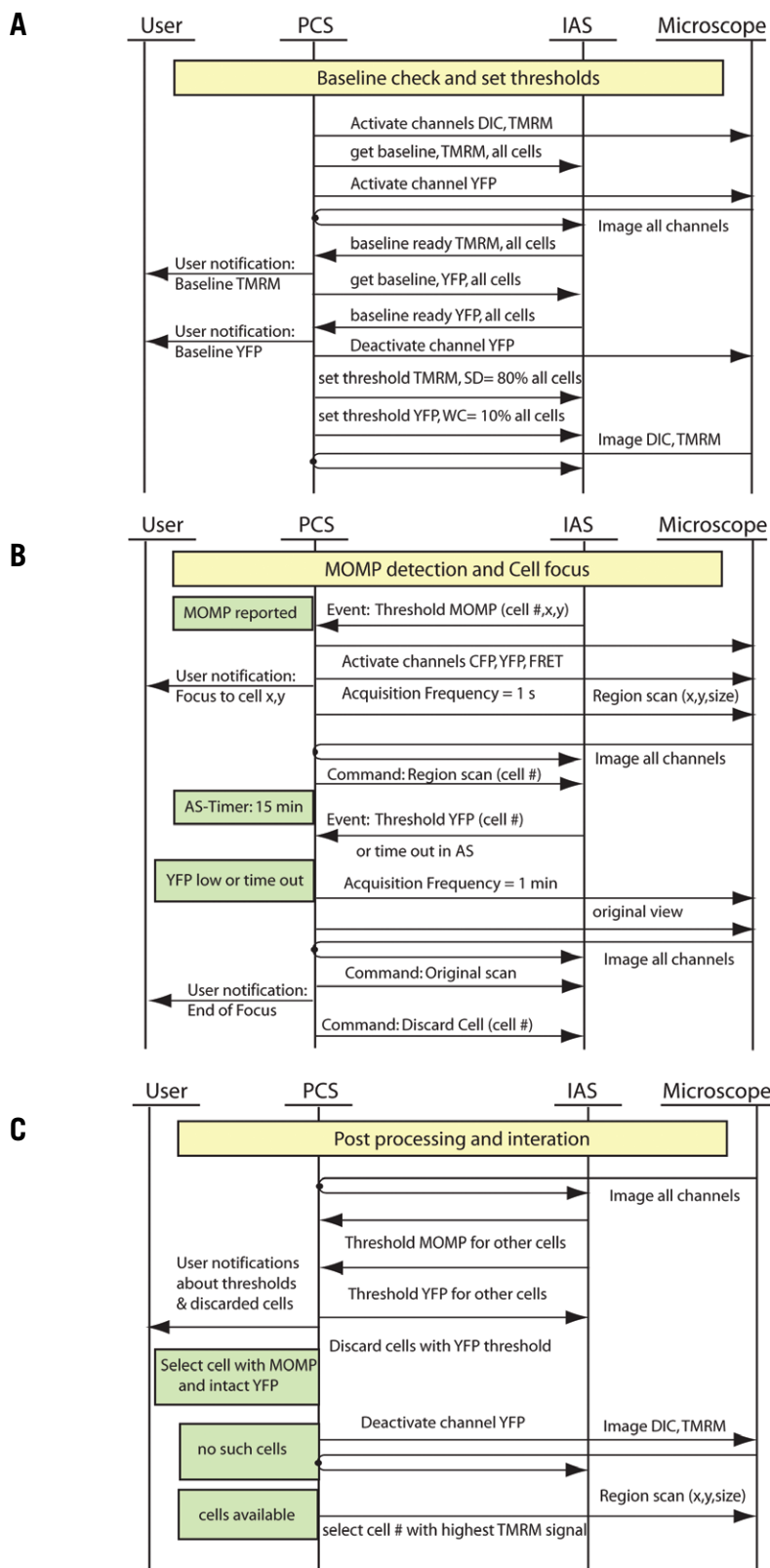
Figure 3. Message flow for the ALISSA use case of the main text (see description therein) following system initialization. (A) PCS requests baseline calculation from the IAS for a certain channel and for all cells. Upon sufficient information, the IAS responds completion. Further, the PCS triggers the IAS to set thresholds for all cells to indicate MOMP occurrence and YFP loss, respectively. (B) After receipt of an event indicating MOMP for a particular cell and until loss of YFP signal or until a 12 min time out, a focus procedure is executed for this cell. This focus procedure employs region scan, rapid sampling and switch on of three further (FRET) channels. The PCS thereby coordinates microscope and IAS to synchronize analysis of the modified images. (C) As scans of other cells are suspended during focus, a post processing procedure collects pending threshold commands. Cells with YFP loss will be discarded. The cell with most recent MOMP (i.e., highest SD of average pixel intensity for TMRM) will be subject of further focus procedure. If during the previous focus no MOM event has occurred, the system will resume MOMP detection.

Supplementary Table 1. Common variables used in the PCS and the IAS for image transfer, commands, events and logging

Name	Type	Description	Length of life
Record_number	Integer	The record (or frame) number, a record being up to 5 greyscale images corresponding to 5 possible channels. This variable acts as an image sequence counter.	updated each cycle by PCS, define a cycle.
Time_stamp	Double	In seconds. Use the PCS timer to provide a time stamp of record acquisition time, not of variable transfer	updated each cycle by PCS
Channels_present	String(5)	channel present: 01001 example: channel 1 and channel 4 contain an image	updated each cycle by PCS
Image_Channel1	Uint8 image matrix	Contains image from channel 1 for record record_number if channels_present set to 1 for this channel	updated each cycle by PCS
Image_Channel2	Uint8 image matrix	Contains image from channel 2 for record record_number if channels_present set to 1 for this channel	updated each cycle by PCS
Image_Channel3	Uint8 image matrix	Contains image from channel 3 for record record_number number if channels_present set to 1 for this channel	updated each cycle by PCS
Image_Channel4	Uint8 image matrix	Contains image from channel 4 for record record_number number if channels_present set to 1 for this channel	updated each cycle by PCS
Image_Channel5	Uint8 image matrix	Contains image from channel 5 for record record_number number if channels_present set to 1 for this channel	updated each cycle by PCS
Commands	50 by 64 uint8 matrix	This is just a communication buffer between IAS and PCS, commands pending, but read, are considered as processed and dumped from this array. Reading is done until the line corresponding to Length_commands	Accessed end of each cycle by IAS to prepare next cycle, processed command are dumped each cycle
Length_commands	Integer (max 50)	Actual size of Commands array	Updated by IAS when command are dumped and by PCS when command are sent.
Events	50 by 64 uint8 matrix	Event is written in this array at Length_events place, and Length_events is updated to Length_events+1.	Accessed by PCS each time length_event is modified by IAS and by IAS when an event occurred at the end of cycle before command reading.
Length_events	Integer (max 50)	Actual size of Events array	Updated by PCS when events are dumped and by IAS when events are sent.

Supplementary Table 2. Command sent from the PCS to the IAS

Command name	Used parameters	Command/ parameter explanation
C: Initialize	Channel bitmap	The PCS informs the IAS about available channels.
	mode	Mode = 0 request start with user interaction.
C: Set Baseline	Cell #	A baseline is calculated for all cells (0) or a specific cell 1…65535
	Channel bitmap	Bitmap of channels for calculating baseline
C: Set Threshold	Cell #	A threshold is defined by the PCS for all cells (0) or a specific cell 1…65535 and stored in the IAS database
	Means / Evaluation	Whole cell average or standard deviations
	Channel bitmap	Imaging channel for that a threshold is set
	Direction	Flag for fallen below or exceeded
	Value	Value relative to baseline (100 = baseline)
C: Discard Cell	Cell #	A list of specific cells 1…65535 is discarded
C: Focus Cell	Cell #	The PCS commands the IAS to focus on a particular cell.
	Focus in/out	Flag for going into focus state of back to previous view
	Magnification * 10	Magnification can range from 0.1 to 25.5

All commands contain command identifier and sequence numbers. Cell number 0 identifies all available cells. Further details are described in the ALISSA command and event catalog that can be downloaded from the web site http://www.alissaproject.org.

Supplementary Table 3. Events reported from the IAS to the PCS

Event name	Used parameters	Event/ parameter explanation
E: Initialization Report	Result (0 = OK, 1 = segmen-tation fault, 2 = tracking fault), channel bitmap that was chosen for the segmentation	The IAS informs the PCS about the success of the segmentation.
E: Baseline Ready	Channel, mean of evaluation	The IAS inform the PCS that baseline for threshold detection is ready, and that a subsequent threshold could then be detected. In the case of the implemented walking average methods, this happens when a sufficient number of record (30) has been reached.
E: Threshold Report	Cell coordinates, Channel, direction, means of evaluation and actual value	A threshold previously set, was detected by the IAS. The event is cell specific and command triggered
E: Bad signal	Cell coordinates	The IAS detects a cell that is below the detection minimum. The event is cell specific and unsolicited. The PCS may react with C: Discard Cell
E: Cell creation	Cell coordinates, parent cell	A cell is created either by division (parent cell ≠ 0) or comes into the field of view (parent cell = 0)
E: Border Cell	Cell coordinates	A cell that has reached the view field border. The PCS may react with C: Discard Cell.

Events that relate to a command contain sequence number of the associated command. "Cell coordinates" denote the x, y center-of-mass position and maximum diameter of the cell in case of a square region of interest and x, y coordinates of 8 vertices in case of a polygon-shaped region of interest.

cell shape is represented by coefficients of B-Splines with 30 knots, and spline coefficients representing shape as well as the cell position are represented variables in a state space that evolves as a stationary process over time to predict hypothetical shapes. The space of transformation from one time step to the other is confined by affine transformations (translation and scaling) considering a combination of the result of the initial segmentation shape and of a polygon approximating the original segmented cell. The ratio of the original shape to the combined original polygon is also a parameter estimated at each new record. A limited number (500 in our case) of predicted cell boundaries are proposed within this transformation space, with more predictions occurring around the estimation of the previous step with the most important weight. To attribute a likelihood to each hypothesis based on measured image data, the DIC channel is first corrected for illumination by pseudo-flat field correction (3). This is done by dividing the image with the illumination trend computed using a large Gaussian kernel. DIC edges are then enhanced by automatic bi-level thresholding. The estimated shape and location for each cell was then chosen to be the weighted mean of the predictions. The B-Splines coordinates representing the cells boundaries and an octagonal bounding box of each cell are stored and available for further actions, such as focusing on one cell area.

References

1. **Isard, M. and A. Blake.** 1998. CONDENSATION—conditional density propagation for visual tracking. Int. J. Comput. Vis. *29*:5-28.
2. **Shen, H., G. Nelson, D.E. Nelson, S. Kennedy, D.G. Spiller, T. Griffiths, N. Paton, S.G. Oliver, et al.** 2006. Automated tracking of gene expression in individual cells and cell compartments. J. R. Soc. Interface *3*:787-794.
3. **Sezgin, M. and B. Sankur.** 2004. Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging *13*:146-165.