Many research problems involve model-based analysis and prior knowledge about the specific task.

Therefore the future of imaging calls for task-oriented modules developed by application scientists.

ImarisXT facilitates the communication between biologists and computer scientists by providing interfaces known to both parties.

The XTensions repository offers – free of charge – the modules including source code of numerous extensions to Imaris, allowing researchers to use the modules as is or to develop them further.
IMARIS XT – Requirements and Components

- Imaris * with XT license
- Matlab R2007b (or newer) or MCR Runtime*
- XTensions (installed with Imaris)
- Java Runtime Environment JRE 6* or later.
- Python 2.7* (if using Python XTensions)
- ImageJ / Fiji*
- Plugins (installed with ImageJ/Fiji)
- Programming languages (supporting the XT Java interface)

*available in the customer portal
Configuring the Imaris – Matlab connection on Windows

- Install Java (optional Matlab comes with a old Java installation)
- Install the Matlab Runtime or full version
- Set Path to XTensions
  - Matlab **Runtime**: (compiled XTensions) Imaris rtmatlab folder e.g. C:/Program Files/Bitplane/Imaris x64 X.X.X/XT/rtmatlab
  - Matlab **Application** (m-files XTensions) Imaris matlab folder e.g. C:/Program Files/Bitplane/Imaris x64 X.X.X/XT/matlab

Screenshot shows Runtime setting
Configuring the Imaris – Matlab connection on OSX

- Install the Matlab Runtime or full version
- Set Path to Executable
  - Imaris - Preferences – Custom Tools
  - Matlab Application: /Applications/MATLAB_XXXXX/bin/matlab
  - Matlab runtime environment: /Applications/MATLAB/MATLAB_Compiler_Runtime/v711
- Set Path to XTensions
  - Matlab Runtime: (compiled XTensions) Imaris rtmatlab folder e.g.
  - C:/Program Files/Bitplane/Imaris x64 X.X.X/XT/rtmatlab
  - Matlab Application: (m-files XTensions) Imaris matlab folder e.g.
  - C:/Program Files/Bitplane/Imaris x64 X.X.X/XT/matlab

Screenshot shows Runtime setting
All XTensions can be found in the Image processing menu in Imaris.

Spots, Surface, Cells and Filament functions require at least one of these present to work. Therefore these XTensions have been made available on the Tools tab of the object that they operate on.
Generates a new channel which contains the distance to surface in each voxel. 
Hint: To get precise measurements you need to convert the dataset to 32 bit float first: ´Edit – Change Data Type – To: 32bit float´
XTension Example: Spots to Spots Closest Distance

- Spots XTension, Output Options
  - Additional Spots Statistics
  - Radius = DistMin
  - New Channel with intensities = DistMin

- Distance Options
  - Center to center
  - Border to border
Super Resolution Localization Data To Image

- Import super-resolution point data from QuickPALM format to build a voxel-based image
- Specify voxel size you want
- Intensity corresponds to the number of points contained in that voxel
- Visualize as Volume / Slicer, or analyze using Surfaces, Filaments, etc

Image courtesy of Ricardo Henriques, UCL, MRC-LMCB
Attenuation Correction

Layers on top (close to objective lense) show higher intensities than layers deeper into the tissue.

- bright spot on top: intensity 256.
- bright spot on bottom should have the same intensity (it currently has only 90).
- The software takes these values and corrects accordingly.

THP-1 cell transmigration through a micropore filter. Image courtesy of Dr. Tomoko Shibutani, DAIICHI Pharmaceutical Co.
Introduction to Imaris Interface

Connect to Imaris in Java

To connect to an Imaris instance, it is first necessary to connect to the ImarisServer:IServer at port 4029 using ICE. The package Ice is included in Ice.jar. The package ImarisServer is generated compiling \$imarisServer\$ice.jar.

Example in Java:

```java
ImarisServer.IServerPpx GetServer() {
    Ice.Communicator vCommunicator = Ice.Util.initialize();
    Ice.ObjectPpx vObject = vCommunicator.stringToProxy("ImarisServer.default -p 4029");
    ImarisServer.IServerPpx vServer = ImarisServer.IServerPpxHelper.checkedCast(vObject);
    return vServer;
}
```

Connect to Imaris in Matlab

The code shown to connect to Imaris in Java is provided as a utility in ImarisLib.jar (this file is located in imaris_installation_folder/XT/matlab). This jar file can be used to connect to Imaris from Matlab, that does not directly support ICE.

Example in Matlab:

```matlab
function aImarisApplication = GetImaris
    javaaddpath ImarisLib.jar;
    vImarisLib = imarisLib;
    vObjectid = 0; % this might be replaced by "vObjectid = <a name=getobjectid><b>GetObjectId</b></a>" (see later)
    aImarisApplication = vImarisLib.GetApplication(vObjectid);
```

As the first Imaris instance that registers itself to the Server is assigned to ID zero, an aObjectid equal to zero will work in most of the cases. ImarisLib.jar grants access to the Server: this can be useful in case of multiple instances of Imaris are started.
% 3D to 2D Projection Function for Imaris 7.3.0
% Copyright Bitplane AG 2011
%
% Installation:
% - Copy this file into the XTensions folder in the Imaris installation directory
% - You will find this function in the Image Processing menu
%
<CustomTools>
  <Menu>
    <Item name="Project to 2D" icon="Matlab" tooltip="Resize the dataset to 2 dimensions.">
      <Command>MatlabXT::XT3Dto2DProjection(1)</Command>
    </Item>
  </Menu>
</CustomTools>
%
% Description:
% Resize the dataset to 2 dimensions.
% 2 dialog boxes ask the projection plane XY, XZ or YZ) and the
% projection mode (MIP or Mean) if their values are not specified
% as input parameters.
%
function XT3Dto2DProjection(aImarisApplicationID, aProjectionPlane, aProjectionMode)

if isa(aImarisApplicationID, 'Imaris.IApplicationPrxHelper')
  vImarisApplication = aImarisApplicationID;
else
  % connect to Imaris interface
  javaaddpath ImarisLib.jar
  vImarisLib = ImarisLib;
  if ischar(aImarisApplicationID)
    aImarisApplicationID = round(str2double(aImarisApplicationID));
  end
  vImarisApplication = vImarisLib.GetApplication(aImarisApplicationID);
end
vImarisApplication.DataSetPushUndo('Projection to 2D');
Configuring Imaris – Fiji / ImageJ Bridge

- Install Fiji/ImageJ to a folder were the user has write access
- Configure the path to the Fiji/ImageJ executable
  - Win: C:/Users/Public/Fiji.app/ImageJ-win64.exe
  - Mac:/Applications/Fiji.app/Contents/MacOS/fiji-macosx
- The „bridge“ plugin will be installed automatically when you start Fiji/ ImageJ from Imaris.
Embed possible Fiji/ImageJ Plugins

Menu available, if Fiji/ImageJ configured

Plugin Compatibility

<table>
<thead>
<tr>
<th>file input</th>
<th>image input</th>
<th>other input</th>
</tr>
</thead>
<tbody>
<tr>
<td>no (e.g. run macro on files only)</td>
<td>yes (e.g. save image to file)</td>
<td>no (e.g. store PCA result as file)</td>
</tr>
<tr>
<td>yes (e.g. load image from file)</td>
<td>yes (e.g. gaussian image filter)</td>
<td>no (*)</td>
</tr>
<tr>
<td>no</td>
<td>(yes) output not fed back to Imaris</td>
<td>no (e.g. do PCA based on some statistics)</td>
</tr>
</tbody>
</table>

(*) There could be various formats of data for input or output other than an image or a file. If these formats are compatible with Imaris Spots or Surfaces, we could implement an interface for it (probably a later release).
To properly run the plugin in Imaris it is important to create the **configuration file**:

```xml
// <CustomTools>
//   <Menu name="Fiji">
//     <Submenu name="submenu name (e.g. Process)"><![CDATA[ - optional Line to embed the plugin in the submenu
//       <Item name="plugin name" icon="Fiji">
//         <Command>Fiji::submenu name_plugin name</Command> - plugin name = configuration file name
//       </Item>
//     </Submenu>
//   </Menu>
// </CustomTools>

call("Imaris_Bridge.In", getArgument());
run("plugin name");
call("Imaris_Bridge.Out", getArgument());
call("Imaris_Bridge.Terminate", getArgument());

Save the file, and copy it to the special folder:" C:\Program Files\Bitplane\Imaris version number \imageJPlugins\configurations
What can the IO platform be used for?

- Organic FAQ
- Meeting point between Life and Computer Scientists
- A place for collaborative work
- Word-of-Mouth / Reference Point
- Centralized platform for requesting and sharing XTensions
- Curated repository of XTensions (citable, unique and permanent url)
Foster cross discipline knowledge transfer
Speed up scientific collaboration
Expand a two way link between life and computer scientists
Enable and lead the Imaris user community
Benefits

- One full Imaris license; value of ≈ 30,000 €
- Increase the visibility of your work
- Privileged access to the Imaris Community
- Safe and permanent URL for your work
- Developer profile on open.bitplane.com
- Access to the Imaris Hackathons
- Access to Imaris XT competitions
Criteria

- Published a custom software application or algorithm in peer reviewed journal
- Software developer at a University, Research Institute or similar
- Actively contribute to the Open-Source community
- Developed at least one Imaris XTension

Imaris XT Developer should

- Use the Imaris Developer License primarily for development and testing purposes
- Be an active contributor to IO (Forum and File Exchange)
- Be an active participant of the Hackathons and other events
Imaris Developer License

- Primarily for development and testing purposes
- If abused it will be revoked
- Initial validity is 1 year
- Renewable if member continues to be active
- Not transferrable
Imaris XT Developer Program (Early Adopters)

Aaron Ponti
Angela Stathopoulos
Christopher Wood
Jean-Yves Tinevez
Jonas Dorn
Josh Thackray
Mario Emmenlauer
Mary Cathleen McKinney
Peter Beemiller
Lee Ling (Sharon) Ong
Ricardo Henriques
Richard Alexander
Quantitative imaging of collective cell migration during Drosophila gastrulation: multiphoton microscopy and computational analysis (October 2009)

Authors: Suppato W, McMahon A, Fraser SE, Stathopoulos A
Affiliation: Division of Biology and Beckman Institute, CIT, Pasadena, USA.
Journal: Blood (IF=9.898)
Title: Hematopoiesis in 3 dimensions: human and murine bone marrow architecture visualized by confocal microscopy. (October 2010)

Authors: Takahu T, Malide D, Chen J
Affiliation: National Heart, Lung, and Blood Institute, NIH, USA
Mapping a sensory-motor network onto a structural and functional ground plan in the hindbrain. (January 2011)

Authors: Koyama M, Kinkhabwala A, Satou C, et al.
Affiliation: Department of Neurobiology and Behaviour, Cornell University, USA
Title: The Rab11a GTPase Controls Toll-like Receptor 4-Induced Activation of Interferon Regulatory Factor-3 on Phagosomes. (October 2010)

Affiliation: Norwegian University of Science and Technology, Trondheim, Norway
Journal: Journal of Cell Biology (IF=10.26)
Title: Dynein-dependent processive chromosome motions promote homologous pairing in C. elegans meiosis. (January 2012)

Authors: Wynne D, Rog O, Carlton PM, Dernburg AF
Affiliation: Department of Molecular and Cell Biology, University of California, USA
Journal: Nature Immunology (IF=26.008)
Title: Integration of the movement of signaling microclusters with cellular motility in immunological synapses. (August 2012)
Authors: Beemiller P, Jacobelli J, Krummel MF
Affiliation: Department of Pathology, University of California, USA.