EDITOR’S NOTE

The Kitware Source contains articles related to the development of Kitware products in addition to software updates on recent releases, Kitware news, and other content relevant to the open-source community.

In this issue, Xiaoxiao Liu discusses the popularity of ITK and VTK in China; Jothybasu Selvaraj explains how VTK was used to perform dose-volume histogram calculations for radiotherapy treatment; Aashish Chaudhary details his work on bringing climate data to the masses; Rusty Blue, Aashish Chaudhary, Ben Payne, and Matthew Woehlke introduce the VisGUI visualization framework for video analysis; and Bill Sherman, Eric Whiting and Aashish Chaudhary discuss immersive ParaView experiences at Idaho National Laboratory.

The Source is part of a suite of products and services offered to assist developers in getting the most out of Kitware’s open-source tools. Project websites include links to free resources such as mailing lists, documentation, tutorials, FAQs, and Wikis. Kitware supports its open-source projects with textbooks, consulting services, support contracts, and training courses. For more information, please visit our website at www.kitware.com.

RECENT RELEASES

MIDAS 3.2.8
In September, the Midas team released version 3.2.8. This release includes improved LDAP support, multi-factor authentication, and improved logging for auditing. In addition, there is now a plugin for algorithm dashboards, Apache Solr can be used as a search index, metadata can be extracted from DICOM data, and there is support for thumbnailing ITK files.

There are also a number of other changes and improvements in the Midas 3.2.8 release, and the full list is available in the change log. Users are encouraged to download and try the latest release from the Midas download page, and join the Midas community and mailing list.

CMAKE 2.8.9
CMake 2.8.9 was released in August. This release featured several significant developments and enhancements, including the new, default-enabled Ninja generator; a position-independent code target property, and MUMPS coverage support in CTest.

The Ninja generator is a replacement for Make, and uses all cores in a machine to enable faster builds by maximizing the ability to parallelize the build code. This is now enabled by default on Linux, Mac, and Windows. To see how this is leveraged for a dashboard machine, you can look at our CMake dashboard scripts.

The position-independent code target property automatically adds -fPIC and -fPIE compiler flags for compilers that require it when building static libraries, which makes it easy to avoid adding platform specifics to your CMakeLists.

Lastly, the MUMPS programming language now has coverage tools, with added support in CTest to include GTM and Cache. This is already in use by the OSEHRA community.

ITK 4.2.1
In early October, ITK 4.2.1 was issued as a bug fix/patch release. This release addressed some critical build and performance issues. Important performance gains were the result of efforts by community members Bradley Lowekamp and Kent Williams through the removal of filter GetInput/GetOutput calls in inner loops. Compilation errors were fixed for Visual Studio 2012. A number of CMake/WrapITK issues were addressed to improve Python wrapping on Windows. There were also other assorted fixes. The next minor feature release is scheduled for December.

New and existing ITK users are encouraged to download and try out the latest release by visiting the download page, or joining the community mailing list.
The Insight Segmentation and Registration Toolkit (ITK) and the Visualization Toolkit (VTK) both have a wide range of applications in medical imaging and geological explorations all over the world. Recently, Professor Yang and his students from Beijing Institute of Technology conducted an interesting survey of the usage of ITK and VTK in China, in terms of publications and web forums among universities and research institutions.

**ITK/VTK PUBLICATIONS**

Here are some statistics about the number of publications that used ITK/VTK libraries in their research work.

![Figure 1. ITK/VTK publications in China (2000 – 2011)](image)

<table>
<thead>
<tr>
<th>Year</th>
<th>Journals</th>
<th>Dissertations</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>'01</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>'02</td>
<td>12</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>'03</td>
<td>4</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>'04</td>
<td>21</td>
<td>10</td>
<td>31</td>
</tr>
<tr>
<td>'05</td>
<td>31</td>
<td>26</td>
<td>57</td>
</tr>
<tr>
<td>'06</td>
<td>47</td>
<td>38</td>
<td>85</td>
</tr>
<tr>
<td>'07</td>
<td>68</td>
<td>40</td>
<td>108</td>
</tr>
<tr>
<td>'08</td>
<td>73</td>
<td>47</td>
<td>120</td>
</tr>
<tr>
<td>'09</td>
<td>62</td>
<td>49</td>
<td>111</td>
</tr>
<tr>
<td>'10</td>
<td>82</td>
<td>49</td>
<td>131</td>
</tr>
<tr>
<td>'11</td>
<td>68</td>
<td>39</td>
<td>107</td>
</tr>
<tr>
<td>'12</td>
<td>13</td>
<td>0</td>
<td>13</td>
</tr>
</tbody>
</table>

Chinese researchers and engineers began to use ITK and VTK libraries in 2001, and the number gradually increased to more than 100 papers published in various journals each year since 2007. The top three research fields in China that are heavily adopting ITK/VTK libraries are: medical Imaging, geological exploration, industrial manufacturing, and aerospace science.

Huazhong University of Science and Technology, Zhejiang University, Beijing Institute of Technology, Southern Medical University, Fourth Military Medical University, Chongqing University, Hefei Industrial University, and Shanghai Jiao tong University have conducted extensive research using ITK and VTK in the field of medical imaging.

Xi’an University of Architecture and Technology, Geological Surveying and Mapping Institute, Chinese University of Mining and Technology, Wuhan Institute of Rock and Soil Mechanics, Henan Polytechnic University, Chengdu University of Technology, and Nanjing Normal University lead China’s ITK and VTK usage in the field of geological exploration.

Huazhong University of Science and Zhejiang University have launched a number of research projects in the field of industrial manufacturing based on ITK/VTK libraries.

Nanjing University of Aeronautics Air University, Beijing University of Aeronautics, and Northwestern Polytechnical University have conducted research projects with ITK and VTK in aerospace science.

**ITK/VTK ONLINE FORUMS**

According to the number of registered members of the Chinese Visualization Laboratory Forum (the most active forum for ITK/VTK in China), the overall national number of ITK and VTK users is about 12,000 and 20,000 respectively. There are four forums for discussing and exchanging information. Several major download websites such as CSDN, Sina love to ask, Pudn etc, and a couple of personal blogs and QQ groups are also providing ITK/VTK resources.

The forums and web resources for ITK/VTK in China are rather sparse and limited. Most ITK/VTK users and developers communicate through QQ groups (similar to AOL and MSN) from the Visualization Laboratory forum (there are about four main QQ groups, with 500 members in total), which are not easy to archive information from for future reference.

Here are some statistics collected from the two major Chinese ITK/VTK forums:

1) The most popular ITK/VTK forum:
   VisLab (http://www.vislab.cn/bbs/index.php)
   The major panels are the VTK forum, ITK forum, and resource forum. Among these forums, there are 11,082 registered users who have contributed more than 1,500 posts and 8,000 replies. The discussion topics mainly involve installation errors in ITK and VTK, basic visualization concept clarification, and how to use specific VTK classes and simple VTK algorithms. The resources panel collects ITK and VTK software guides and a few tutorials, and shares some useful links in related research fields.

2) The second most popular ITK/VTK forum:
This website provides forums for ITK and VTK discussion. The VTK Forum has 1,853 posted messages and more than 10,000 replies. There are 189 ITK discussion topics with 911 replies. The discussions are mostly technical details with specific programming questions.

Outreach to Chinese ITK/VTK Communities
We have seen an increasing number of source code downloads from China over the past few years for ITK, VTK, and Slicer. During my ITK tutorial tour of three Universities/Institutes of three different cities in China this January, I felt that there is a strong demand for ITK and VTK community support. Graduate students and researchers need better communication channels for getting involved in the global open-source community. Although many of them are signed up to the mailing lists, very few of them actually contribute code to ITK and VTK due to the lack of tutoring on the committing process.

As with the releases of ITK 4.0 and VTK 6.0, the need and opportunities for recruiting new developers into our open-source communities presented themselves. Chinese developers need more community support to migrate to newer versions of ITK and VTK, and the community needs more contributions and involvement from China. The needs are mutual and the time is now!

ACKNOWLEDGEMENT
Many thanks to Professor Jian Yang of the Beijing Institute of Technology in China. All of the survey statistics in the article were provided by him and his students.

Xiaoxiao Liu is an R&D Engineer at Kitware. Her research interests are in medical image analysis and applications, including statistical shape analysis for anatomical structures, deformable shape modeling and segmentation, diffeomorphic image registration techniques and image-guided radiotherapy.

DOSE-VOLUME HISTOGRAM CALCULATION IN RADIOTHERAPY TREATMENT PLANS USING VTK
Radiotherapy is the use of ionizing radiation to treat cancer. Sophisticated software programs known as treatment planning systems (TPS) are commercially available to tailor-make treatment plans for individual patients. These systems mainly allow clinicians to predict the dose delivered to each voxel in the patient for a given orientation of the radiation beams and other parameters used before actually treating the patients. The best possible plan, which provides the highest therapeutic ratio (highest tumour control with least normal tissue complications), is selected for treatment. There are many tools available for evaluating rival plans in radiotherapy. The most widely used are the dose-volume histograms (DVH), which are ubiquitous in radiotherapy. Assimilating the vast amount of information in a 3D radiation dose array is very difficult. DVH condenses this vast information into easily interpretable 2D graphs. There are two types of DVHs commonly used, namely differential and cumulative DVHs. Differential DVH represents the percentage or absolute volume (depending on the mode of display) receiving dose in the corresponding dose bin, whereas the cumulative DVH represents the percentage or absolute volume receiving greater than or equal to the value in the corresponding dose bin. These DVHs are also used to calculate other radiobiological plan metrics such as tumour control probability (TCP) and normal tissue complication probability (NTCP), among others. Though DVH is widely-used in radiotherapy, there is neither a single method adopted nor the methods to derive them documented by the TPS vendors. This article briefly explains the method for calculating DVH using VTK.

In radiotherapy planning, the 3D shape of the structures is generally derived from a set of 2D contours drawn on the CT image along the z-axis. A class “DCMImporter” has been developed to import RT structures and plans exported from TPS in DICOM format. A surface mesh of the structure is generated from the imported set of 2D contours using vtkMarchingCubes, and the resulting mesh is stored as vtkPolyData. The dose calculated within the patient volume by the TPS is a 3D array with a commonly used resolution of 2-to-3 mm. This 3D dose array is exported in DICOM format for the corresponding patient, and read in using vtkGDCMImageReader (which is available in gdcm built with VTK support); the output is stored as vtkPolyData. The key step in calculating DVH is to identify the voxels inside the structure of interest. This can be achieved with a combination of some of the VTK classes, namely vtkPolyDataToImageStencil and vtkImageStencil. The dose outside the structure is set to zero using SetBackgroundValue in vtkImageStencil. The vtkImageAccumulate class can then be used to calculate maximum, minimum, mean dose, and standard deviation directly from the output of vtkImageStencil. The entire dose array and dose extracted for bladder using vtkPolyDataToImageStencil is shown on an axial slice in...
Figures 1 and 2 respectively. The code snippet for the DVH calculation is given below.

```cpp
//Extract dose inside the structure
vtkSmartPointer<vtkPolyDataToImageStencil> meshToStencil = vtkSmartPointer<vtkPolyDataToImageStencil>::New();
meshToStencil->SetTolerance(0);
meshToStencil->SetInformationInput(doseGrid);
meshToStencil->SetOutputSpacing(doseGrid->GetSpacing());
meshToStencil->SetOutputOrigin(doseGrid->GetOrigin());
meshToStencil->SetInput(structure);
meshToStencil->Update();

//Convert image stencil to vtkImageData
vtkSmartPointer<vtkImageStencil> stencilToImage = vtkSmartPointer<vtkImageStencil>::New();
stencilToImage->SetStencil(meshToStencil->GetOutput());
stencilToImage->SetInput(doseGrid);
stencilToImage->ReverseStencilOff();
stencilToImage->SetBackgroundValue(0);
stencilToImage->Update();
vtkSmartPointer<vtkImageAccumulate> ia = vtkSmartPointer<vtkImageAccumulate>::New();
ia->SetInputConnection(stencilToImage->GetOutputPort());
ia->IgnoreZeroOn();
ia->Update();
long vc = ia->GetVoxelCount(); //No. of voxels inside

double maxDose = ia->GetMax()[0]; //1st comp. val
double minDose = ia->GetMin()[0];
double meanDose = ia->GetMean()[0];
double std = ia->GetStandardDeviation()[0];
```

The non-zero scalar values (voxels inside the structure) from the output of `vtkImageStencil` are used to make a frequency (number of voxels) histogram using a bin width of 0.02 Gy to reduce the size of the data values. Finally, one gets the volume receiving the dose in each dose bin. These differential DVHs are converted to cumulative DVHs, which are more commonly used in the clinic. Figure 3 shows a differential DVH for a tumour in a breast case. Cumulative DVHs are plotted using vtkCharts along with the DVHs calculated by commercially available TPS (Eclipse™ treatment planning system, Varian Medical Systems, Palo Alto, CA) for a breast and a prostate case in Figures 4 and 5 respectively. The `vtkImageResample` class can be used to resample the dose grid to a finer resolution to obtain more accurate results which will reduce the partial-volume effect.

The volume for the given structure is calculated as the number of voxels times the voxelVolume. The volume is calculated using VTK and analytical methods for a sphere (25 mm radius), cube (25 mm sides), and cylinder (25 mm radius and 40 mm length) are shown for 1 mm and 2.5 mm resolution of dose array in Figure 6. The results are closer to the analytical method for 1 mm resolution because of the reduced partial-volume effect.

The non-zero scalar values (voxels inside the structure) from the output of `vtkImageStencil` are used to make a frequency (number of voxels) histogram using a bin width of 0.02 Gy to reduce the size of the data values. Finally, one gets the volume receiving the dose in each dose bin. These differential DVHs are converted to cumulative DVHs, which are more commonly used in the clinic. Figure 3 shows a differential DVH for a tumour in a breast case. Cumulative DVHs are plotted using vtkCharts along with the DVHs calculated by commercially available TPS (Eclipse™ treatment planning system, Varian Medical Systems, Palo Alto, CA) for a breast and a prostate case in Figures 4 and 5 respectively. The `vtkImageResample` class can be used to resample the dose grid to a finer resolution to obtain more accurate results which will reduce the partial-volume effect.

The volume for the given structure is calculated as the number of voxels times the voxelVolume. The volume is calculated using VTK and analytical methods for a sphere (25 mm radius), cube (25 mm sides), and cylinder (25 mm radius and 40 mm length) are shown for 1 mm and 2.5 mm resolution of dose array in Figure 6. The results are closer to the analytical method for 1 mm resolution because of the reduced partial-volume effect.
CONCLUSION

In summary, it is possible to calculate DVHs using VTK with reasonable accuracy and speed. The dose array can be resampled to finer resolutions if higher accuracy is needed.

Jothybasu Selvaraj is a Ph.D. student in the Department of Medical Physics, Clatterbridge Cancer Centre, UK. His research interests include the effect of geometric uncertainties in radiotherapy, radiobiological modeling and image-guided radiotherapy. He is also passionate about visualization of radiotherapy datasets with VTK.

CLIMATE FOR THE MASSES

Understanding climate change is important, and therefore has been discussed in a large variety of media over the last several years. Changes in climate resonate though a broad range of fields, including public health, infrastructure, water resources, and many others. Over the last few years, climate simulations have produced a wealth of climate simulation data, including output from high-resolution, long-term, climate change projections performed as part of the United States Global Change Research Program [1] and the Coupled Model Intercomparison Project [2]. Widespread use of climate data is impeded by the lack of easy-to-use, effective analysis and visualization tools for policy makers, researchers (from non-climate communities), and other possible users of climate data. Thus, there is a need for communicating climate data to all age, social, cultural, and professional groups.

PHASE I ACHIEVEMENTS

Visualization is visual communication for the purpose of the presentation and exploration of data, concepts, relationships and processes; it presents powerful capabilities that can yield useful insights into climate data. Kitware, in collaboration with NYU-Poly, has leveraged ParaViewWeb [3], VisTrails [4], Ultrascale Visualization – Climate Data Analysis Tools [5] (see Figure 1), Earth System Grid Federation [6], and the latest web technologies to provide a prototype of our state-of-the-art web tool, ClimatePipes, to climate data users.

Figure 1: Various 2D and 3D plots in UV-CDAT

ClimatePipes enables users to access, query and visualize climate datasets from multiple sources. For the Phase I of the project, ClimatePipes provided a simple, easy-to-use form-based interface, and a more advanced dataflow interface to construct more complex workflows. Figure 2 shows an example dataflow created using the pipeline interface to fetch a list of datasets available from an ESGF source.

The simpler interface (see Figure 3) enabled users to select a region of interest, select a data source; set spatial and temporal filters, and desired output; and view a list of matched data items or visualization of closest-matched climate data.

PHASE II VISION

Recently, ClimatePipes received a notice of Phase II award from the DOE to transform the ClimatePipes prototype into a more advanced and user-friendly tool.
an open-source production quality tool, which could be used by researchers or non-researchers to understand long-term climate change projections. In Phase II of the project, ClimatePipes will enable users to run complex computations, and generate effective visualizations in the cloud or on a user-provided cluster. Users will be able to upload data or visualizations to a server for analysis and integration. Figure 4 provides an overview of the Phase II design.

The key objectives of the Phase II of ClimatePipes are as follows:

Enable users to explore real-world questions related to climate change.
ClimatePipes will enable users to find answers to real-world questions regarding climate change. Using semantic search on climate datasets will enable ClimatePipes to provide relevant results to user queries.

Provide tools for data access, analysis, and visualizations.
ClimatePipes will enable users to create state-of-the-art info-views, which will provide the ability to combine data analysis and visualization into a single view. Drawing from a variety of visual results including 2D charts, geographic plots, and 3D visualizations, ClimatePipes will enables users to not only access but also explore climate data efficiently and effectively.

Facilitate collaboration by enabling users to share datasets, workflows, and visualizations.
ClimatePipes will provide the means to share datasets, workflows, and visualizations between users of communities. Similar to other social networks, users will be empowered to create groups of communities. ClimatePipes will also enable users to annotate data with specific tags and notes.

CONCLUSION
ClimatePipes is a community-supported tool, and as such, will foster collaboration between various commercial, non-profit, and governmental groups. Different agencies and companies publish datasets and tools that are often difficult to reconcile with other data or software, but ClimatePipes provides an extensible infrastructure to integrate them. The tool will also allow users to not only browse and visualize datasets in a Web browser, but also to download their data and analyses to work offline with their own tools of choice. Additionally, it will track the provenance of such analyses so the entire process can be reviewed and shared by other researchers, agencies, and the public.

This work has been supported by the Department of Energy (DOE) under award number DE-SC0006493.

REFERENCES:

Aashish Chaudhary is an R&D Engineer on the Scientific Computing team at Kitware. Prior to joining Kitware, he developed a graphics engine and open-source tools for information and geo-visualization. Some of his interests are software engineering, rendering, and visualization.

VISGUI: A VISUALIZATION FRAMEWORK FOR VIDEO ANALYSIS

The visualization capabilities offered by current tools for vision datasets are limited in many ways, hindering the ability of researchers to quickly inspect image and video data and verify algorithmic results. Therefore, we developed VisGUI; a novel, high-performance, extensible, and cross-platform framework, which offers various ways to visualize and analyze vision datasets in an effective and efficient manner. VisGUI is built on top of Qt [1] and VTK [2], sophisticated and mature cross-platform frameworks, and is described in detail in subsequent sections.

We recently received approval for the public (open-source) release of the VisGUI framework. We’re currently in the process of preparing its release, which is expected before the end of 2013.

DESIGN AND IMPLEMENTATION
Figure 1 shows the overall architecture of VisGUI. The core dependencies of VisGUI are Qt and VTK; lesser dependencies include other toolkits such as VXL, VIDTK, PROJ4, GDAL, GeographicLib, and KML. The VisGUI architecture primarily consists of a model-view framework, a scene graph, GUI elements, and a testing framework.
Model-View Framework

VisGUI uses a VTK-based model-view framework, separating the domain logic from the visual representation. This separation enables the creation of different representations for the same underlying data and rendering optimizations where only the data required for rendering is passed to the representation. Figure 2 shows a list of the models and representations available within the VisGUI framework.

Scene Graph

The map view of Viqui (discussed later) contains video results and other related visual entities that are spatially and temporally bonded to each other. Since this relationship can be described by a scene graph data structure, we created a VTK-based scene graph in which leaf nodes represent visual entities such as a track or video in the scene. These leaf nodes are comprised of at least one data model and one-or-more data representations. The scene is rendered by performing a depth-first traversal.

GUI Extensions

The application code within VisGUI primarily consists of VTK rendering elements within a Qt infrastructure. QtExtensions is a set of classes that extend Qt by providing useful features such as a command-line parser, real-number slider widget with arbitrary precision, helper classes for application configuration management, and a unit testing framework. QtExtensions is a standalone library and could be used independently of VisGUI or VTK. Similarly, we extended and developed several VTK-based widgets to provide enhanced interaction capabilities.

Testing Framework

No framework or application is complete if it is not tested. The VisGUI framework provides system integration and unit tests to ensure software quality and reliability. The core of the VisGUI testing framework is built on top of the CMake/CTest/CDash framework. GUI testing is performed by recording a sequence of events and playing these events back on a test machine. The final viewport render is then compared against a baseline image. VisGUI uses the QtTesting library, which was developed as part of a GUI testing mechanism for ParaView, as the underlying engine for GUI testing.

APPLICATIONS

We have leveraged the VisGUI framework in developing three GUI applications, which provide graphical interfaces to control aspects of the visualization such that the video data, metadata, and derived data can be analyzed in a way that is easier to comprehend. We describe some of the features of these applications in the remainder of this article.

VSPLAY

The VsPlay application is targeted at working with a single stream source. It provides tools for interactively visualizing video and metadata from an input stream, and the high-level detections from a video processing and analysis system associated with that stream. The stream can be either “live” (data that is currently being processed by the system) or “archived” (data that has been previously processed). Working with live video also provides limited ability, in the form of alerts, to interact with the running system. Additionally, live feedback can be submitted to the running system to provide a form of streaming Interactive Query Refinement (IQR). High-level features of VsPlay include video viewing with DVR-style playback controls, visualization of tracks and events, geospatial filtering of events and tracks, and the ability to generate new events based on user input. Some of these features can be seen in Figure 3.

Viewing Video

Accessing and displaying the video is one of the core features of VsPlay. Full DVR-style playback controls are provided, allowing the user to pause and rewind the video, as well as alter the playback speed (both faster and slower than real time). Video playback is time-based, so playback follows real time in the face of uneven or dropped frames, and will automatically skip frames to maintain playback speed if the user’s machine is excessively loaded. Additionally, there is a time scrubber to allow quick seeking within the video. Various metadata is displayed in addition to the video image, including the time that the video was shot, the
frame number within the current stream, and the estimated ground sample distance of the current frame. The interface also provides the estimated geographical coordinates of a given pixel, and the ability to measure the approximate ground distance between two points on the image.

Visualization of Tracking and Events
VsPlay enables the visualization of tracks formed from the detection of moving objects. Tracking boxes, showing the video region that has been identified as an object, are drawn on the current frame for objects that are detected at that frame. Tracking history is drawn as a line from the estimated ground center of the tracked object, allowing the location and motion of the object over time to be visualized. Display of track boxes, track history, and track labels (including track classification) is all optional and can be turned on or off as required. Events such as walking, running, carrying, starting, and U-turning are displayed in a similar manner (See Figure 4).

Annotations
VsPlay provides the ability for users to make annotations on the video. Annotations are persistent, user-created shapes, which can be drawn while the video is playing and are persistent. By using the system’s video stabilization, their ground location remains constant as the camera moves. While an annotation is simply a marking on the video, the underlying capability is leveraged to provide additional functionality, including trip wires and geospatial filtering.

User-Defined Events
In addition to "canned" event detection and classification, VsPlay includes features that allow analysts to input additional parameters from which events should be generated. These can come in multiple variations, and the design of VsPlay enables new variations to be added as needed. One type of interactive event detection is the "trip wire" event (see Figure 5), which is a type of user region, like an annotation, that generates an event whenever an object (track) crosses it.

Another type of interactive event is a streaming query. The user selects a previously defined query (discussed more in a later section) and asks the system to match against it in real time. This can be used to detect event types not currently "built into" the system, or as a means of bringing attention to events matching more complicated criteria than the simpler built-in event detections. For example, the system might identify the event "vehicle turning," but a streaming query could potentially be used to identify a red vehicle that stops and then turns. The user may then provide feedback to the system to refine the query in real time.

Information Filtering
In order to reduce cognitive load and to help focus attention on relevant events, VsPlay provides several features to filter and organize information. Filtering is provided based on both event classification and geospatial location. Classification filtering operates in two stages. First, all information pertaining to a tracked object may be filtered based on its type. This is most typically used to tune detection to reduce false positives, but can also be used to quickly eliminate object classes that are not currently of interest. Second, events can be filtered by specific classification. Examples of specific classifications include "vehicle turning," "person running," "trip wire crossed," or specific streaming queries. The UI also provides controls to easily toggle groups of classifications. Both styles of classification filtering also allow the user to specify a confidence threshold, so that only classifications above the requested threshold are shown (see the panel in the right portion of Figure 3).

Geospatial filtering makes use of user regions that have been designated as either filters or selectors, to allow the user to eliminate specific geospatial areas (such as a rooftop prone to false detections, as shown in Figure 6), or restrict displayed tracks and events to a specific region of interest.

Figure 4: Events detected by the system are shown as a thicker line overlaid on the track, with labels indicating event type and probabilities. Source data is from the VIRAT public dataset [3].

Figure 5: Example of a trip wire (shown in yellow) drawn around a structure, such that events will be created for tracks entering and exiting the structure. Source data is from the VIRAT public dataset [3].

Figure 6: Example of a “filter region,” a user drawn and ground stabilized region that “filters out” undesired returns (the false detections on the roof of the building). Source data is from the VIRAT public dataset [3].
Lastly, VsPlay provides event and track panes to allow for fine tuning of track and event display and controls to jump to the start or end of a track or event.

VIQUI
Viqui is the front-end for the archival capabilities of a vision system, allowing a user to search a database of ingested video for video clips that match flexible user criteria. In summary, Viqui enables an analyst to:

- Define a query
- View and analyze results of a query
- Interactively refine a query

Results in Viqui are visualized using either a georeferenced display over a context map (which could be a street map or satellite imagery), or using a timeline chart. Result lists are also available, giving more detailed textual information such as the calculated relevancy score, rank, and the start/end time of each result. Individual results may also be viewed in a simple video player, which helps the user to focus on that result, while the other views provide an overview and can allow users to identify patterns in a result set (see Figure 7).

Query Formulation
Query formulation can take several forms, including exemplars and predefined queries that can be further constrained by geospatial or temporal limits. Users can also save queries and query results for later use. Viqui provides several methods to define a query. The primary query types are:

- Video Exemplar
- Image Exemplar
- System Predefined

Video Exemplar
Perhaps the most flexible query method is the video exemplar, where users provide an example video containing an instance of the event they wish to find. This video is processed by the system, after which users select one-or-more appropriate descriptors that numerically represent various aspects of the processed video. The system also provides a video player that presents tracks and regions detected by the system in the processed video, as shown in Figure 8.

In some cases, it may be desirable to formulate a new query based on the contents of an existing result video clip. For example, the user might notice an interesting event in a video clip returned by a query. Users can then use this video clip for query formulation purposes by loading the clip in the query formulation interface and selecting the descriptors corresponding to the interesting event as the basis for a new query.

Image Exemplar
Image exemplar queries are formed by providing the system with an example image, rather than a video clip. The image is processed using a special pipeline optimized for this mode of operation. The user then selects the desired descriptors (appearance properties for different sub-extents of the image) using the same interface as the one used in video exemplar query formulation.

System Predefined
The simplest means of performing a query is by selecting one from a predefined list; that is, using queries that have previously been created using an alternate method of formulation. The list is generated by traversing a directory structure that contains previously saved query files. Additional constraints may be placed on a query by providing temporal or spatial limits for the results.

Query Result Overview
The map view, also the main view of Viqui, displays an overview of the query results in a geographical context. A VTK-based scene graph was developed for this view to capture the spatio-temporal relationship between the video results and related visual entities. In this view, the first frame of each result is displayed in its geospatially-registered location on the context as seen in Figure 1. Since the first frame of a result gives only a limited sense of the quality of the result, the border of each of the results is colored according to one of a variety of schemes to provide additional information. Results are also displayed in a table view and a timeline view. Playback of a particular result can be started from either the context view (by double-clicking the desired video window) or by selecting the desired result in any of the other views. A standalone video player is provided to view the video in a traditional mode. Playback of the video in the video player is synchronized with playback on the context, which is geospatially registered for each video such that the video pixels
correspond roughly to the ground plane; the video window is warped and rotated according to the camera parameters for each video frame.

The map view provides basic interaction features such as pan, zoom, and rubber band zoom. Since a query can potentially have thousands of results, it is possible that some of these results will be co-located or nearby, such that overlap or occlusion occurs. To help alleviate this problem and to reduce cognitive overload, Viqui provides Z-Sort, Blast, and Stack layout strategies as shown in Figure 9.

![Figure 9: Different layout modes: Z-Sort, Blast, Stack (Left to Right) for results in Viqui (Source data is from the VIRAT public dataset [3])](image)

**Timeline View**

It is often useful for analysts to get a sense of the temporal spacing, ordering, and duration of various results. The timeline pane (shown at the bottom of Viqui in Figure 7) provides this functionality by displaying the time span of results along the X-axis, and an offset for overlapping results along the Y-axis. Inspired by similar tools found in non-linear video editing software, the timeline chart is fully interactive and enables users to pan, zoom, and select results. The timeline is also integrated with the rest of the application; selecting an interval in the timeline will cause the corresponding result to be selected in the result list, and its video will begin playback in the context view and video player.

**Clip Viewer**

The clip viewer (as shown in Figure 10) provides the ability to view video clips from multiple results simultaneously. Multiple result clips are cropped to the relevant detection and played back in real time, allowing the analyst to quickly compare results of interest.

![Figure 10: Clip viewer showing multiple result clips simultaneously playing back. Source data is from the VIRAT public dataset [3].](image)

**Analyzing and Refining the Result Set**

Interactive Query Refinement (IQR) enables users to focus the search algorithm on (or away from) particular aspects or features within the results. During the review process, the user has the ability to mark each result as either relevant or not relevant. This feedback is then used by the system as a means to rescore and thus reorder the results to be returned.

**Report Generation**

VsPlay and Viqui both provide the ability to compile and output information about a selected subset of events to aid in creating summary reports for later analysis. The content included in the summary report includes an encoded full-frame video for each event, a representative sample image from a single frame of video with track and region overlay information, geographic coordinates of the event for that frame, any per-event notes added by the user, and any other metadata relevant to the detection. The data contained in the generated report are human-readable, but are also formatted in a way to make it easy to process with other tools.

**VPVIEW**

VpView was initially developed to visualize and analyze threat detection in wide-area motion imagery and is similar to VsPlay with respect to track and event visualization. However, the large wide-area images, inherent to the collected data, led to the development of a new multi-resolution JPEG (MRJ) format, which enables VpView to have fine control of image level-of-detail in order to maintain interactive frame rates for the visualization. Additional features unique to VpView within the VisGUI tools are activity visualization, a 3D graph view for data analysis, and an interactive graph model editor.

**Activity Visualization**

Activities are a collection of coherent events. To visualize the activity, a representation was developed that uses VTK’s implicit modeler to build a reasonably tight 2D shell around the events that comprise the activity. The 2D shell is then rendered translucently such that the imagery is visible beneath the representation (see Figure 11).

![Figure 11: Tracks, events and activities shown in a 2D View. Source data is from CLIF dataset [4].](image)

**3D Graph View**

As the number of tracks, events, and activities increases or the relationships become more complex, the basic 2D view can become too cluttered. This led to the development of a 3D view (shown in Figure 12) where the entities are drawn as graph nodes and the relationship between the entities are shown as edges of the graph. In this view, activities are drawn at the highest level, followed by events and then tracks. This hierarchy made sense since activities are
comprised of events, which in turn are comprised of tracks. The 3D view mode also provides a full volume mode, which when activated, simultaneously renders all of the detected entities and their relationships. The 3D view utilizes VTK’s graph framework and a custom GraphMapper.

Figure 12: Tracks, events and activities shown in the 3D Graph View. Source data is from CLIF dataset [4].

Graph Model Tool
The VpView graph editor provides researchers with a convenient method of creating a model for a new activity and defining the relationship of the events that comprise the activity. Internally it uses a custom GraphModel and GraphMapper from the VisGUI framework. The graph editor view provides various interactive features such as pan and zoom, node and edge selection, interactive addition and deletion of nodes and edges, and interactive labelling of nodes. Importing and exporting of a model is accomplished via a JSON file format. Figure 13 shows a graph created using the graph editor.

Figure 13: Graph Created by the User using the Interactive Graph Editor of VpView

CONCLUSION
The VisGUI toolkit has proven to be quite capable in building fully-featured, interactive, cross-platform, high-performance applications. The tools have served as the delivery mechanism for vision results of many high-profile demonstrations, and have received praise from active-duty analysts. Originally developed in support of the DARPA PerSEAS [5] and VIRAT [6] projects, the tools are being used with only minor modifications on other research projects within the vision program.

For further information or to discuss potential collaboration, please contact kitware@kitware.com.

REFERENCES

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract Nos. HR0011-08-C-0135 and HR0011-10-C-0112. It is approved for release under DISTAR #19562. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA.

Aashish Chaudhary is an R&D Engineer on the Scientific Computing team at Kitware. Prior to joining Kitware, he developed a graphics engine and open-source tools for information and geo-visualization. Some of his interests are software engineering, rendering, and visualization.

Rusty Blue is a Technical Leader on the Computer Vision team at Kitware. With nearly 15 years of VTK experience, he leads a team developing visualization tools and applications utilizing VTK in support of many of the vision projects.

Ben Payne is an R&D Engineer on the Computer Vision team at Kitware. Before joining Kitware, Ben worked in the video game industry. He holds a B.S. in Applied Science with a focus on Computer Engineering, and a Master’s degree in Computer Science.

Matthew Woehlke is an R&D Engineer on the Computer Vision team at Kitware. He holds a B.A. in Computer Science and Mathematics from Concordia University, which he received in 2004.
The Center for Advanced Modeling and Simulation (CAMS) at the Idaho National Laboratory offers an advanced visualization facility that includes a four-sided CAVE-style immersive display system, a large 3x3 tiled display, and low-cost immersive systems dubbed the IQ-station [5] (also see Figure 1). Given these immersive environments, the CAMS team has a vested interest in making it very easy for scientists to bring their data into their facility and enable them to quickly explore and interact with it. Highly customized tools requiring specific data formats work very well in the CAVE, but these tools do not meet the needs of the new or occasional user without the means to develop custom code. The common user use-model drives the need for a general purpose visualization tool compatible with immersive environments.

ParaView [1] was recognized as a great fit for introducing scientists to using immersive systems by allowing them to make use of a tool they can learn to wield on their desktop, and transfer these skills to the advanced visualization systems. In 2010, the CAMS team began a collaboration with Kitware, Indiana University, and the University of Wyoming to add features necessary for immersion (i.e. virtual reality interfaces) to ParaView, including some new features to the underlying VTK library [2].

The result of this collaboration is a collection of code changes labeled as ‘Immersive ParaView.’ These extensions to the code base enable ParaView to display and interact with objects in an immersive environment. This work followed a development cycle that included: analysis and exploration of similar efforts from earlier releases of ParaView and other tools; a first-pass implementation modifying VTK for basic operation and gaining understanding of the various connections required for immersive applications; and a second-pass implementation that re-engineered and overhauled the code to improve correctness and maintainability.

The team recently gathered at the Center for Advanced Energy Studies (CAES) facility in Idaho Falls, Idaho to test and evaluate the prototype code, which is now part of the ParaView source code release. The goals of this meeting were to test the functionality of the prototype and determine how to move forward in a way that will allow the process to be easier to configure and run for the general user.

TESTING THE PROTOTYPE IMPLEMENTATION

The new immersive functionality makes use of ParaView’s client/server operational mode. One-or-more server processes are configured to render all the views needed to fill the display(s) of the immersive system. The client then resides on a separate screen, and remains the primary means by which the visualization pipeline is built and configured.

Once the client/server connection is established, the plugin manager is used to load the “VR Plugin” into both the client and server processes. Doing so reveals a new widget panel (a new addition based on our evaluation, see Figure 2) that allows the user to initiate a VR session.

Our tests at the INL CAMS facility (see Figure 3) included the use of the crushed can sample dataset, as well as some computational data provided by INL researchers. With the recent off-axis stereo feature in VTK, we were able to get correct immersive rendering both in the four-sided CAVE and on the smaller-scale IQ-station. The tracking feature is part of the “VR Plugin,” and can use either the VRPN [4] or
the Vrui Device Daemon protocol [3]. With tracking enabled, immersive rendering responds properly to head movement; and additionally the world can be grabbed and moved with a hand-held wand.

![Figure 3: Testing the prototype implementation in the INL four-sided immersive environment](image)

Testing the prototype demonstrated the practicality of adding immersion through the VR plugin system. Given the functional baseline code, it was decided to address the identified shortcomings that were encountered. These were primarily in the process of configuring the immersive features. As part of preparations for a wider user base, we focused on creating a new VR widget panel and adding Python scripting access to the new VR plugin features.

![Figure 4: Latest version of VR plugin running on IQ station at INL](image)

**FUTURE WORK AND CONCLUSIONS**

We are actively making improvements to the VR plugin to make it easy to configure VR input device servers and interactor styles. We are also making changes to the code base so that interactor styles can be wrapped using the VTK-Python wrapping framework. Another very useful added feature is the support for side-by-side stereo (or split viewport stereo) for the current generation of 3D TVs. This feature will enable more users to run ParaView in "VR" mode on a low-cost immersive system built from consumer model televisions.

Presently, there are two primary interactions styles available through the VR plugin: head-tracking and grab-the-world. We recognize and are working to address the need for additional methods of navigation, and a means of interacting with control widgets such as slice planes. Overall, the ParaView VR plugin is proving to be extremely useful in the various immersive systems we have tested and we believe that it is ready to be used in production environments.

This work has been supported by the Idaho National Laboratory and is publicly released as open-source software in the ParaView repository.

**ACKNOWLEDGEMENT**

The authors would like to thank the following Kitwareans for their contributions to this effort: Nikhil Shetty, David Lonie, Patrick O’Leary, Utkash Ayachit, and Berk Geveci.

**REFERENCES**


---

**Bill Sherman** is a Senior Technical Advisor in the Advanced Visualization Lab at Indiana University. Sherman’s primary area of interest is in applying immersive technologies to scientific visualizations. He has been involved in visualization and virtual reality technologies for over 20 years, and has been involved in establishing several immersive research facilities.

**Eric Whiting** is the Director of the Center for Advanced Modeling and Simulation at the Idaho National Laboratory. His area of expertise focuses on the area of high performance computing. In addition to HPC, he is experienced in electronics, signal processing, computer architectures, computer programming, and networking.

**Aashish Chaudhary** is an R&D Engineer on the Scientific Computing team at Kitware. Prior to joining Kitware, he developed a graphics engine and open-source tools for information and geo-visualization. Some of his interests are software engineering, rendering, and visualization.
KITWARE PRESENTS PYTHON TUTORIAL AT CVPR

Over the summer at the CVPR conference, Kitware collaborated with Google Research to present the “Python for MATLAB Users” tutorial. The tutorial was taught by Matt Leotta, Eran Swears, and Patrick Reynolds from Kitware, and Yong Zhao and Varun Ganapathi from Google Research.

The course provided a hands-on introduction to using Python for rapid prototyping and computer vision research. Python was presented from a MATLAB users’ point-of-view, showing how to directly map the familiar MATLAB syntax and workflow to a very similar syntax and workflow in Python. The team demonstrated a free and open-source alternative to MATLAB for promoting open science, and to better prepare students for careers outside of academia.

Attendees received a virtual appliance (VirtualBox) with the full scientific Python environment pre-configured, and executed programming exercises ranging from introductory to intermediate levels. The material, including the virtual appliance, was distributed on pre-configured USB memory sticks and DVDs. Participants spent the day working on exercises, asking questions, and collaborating.

Based on the positive feedback from participants and CVPR attendees who wanted to but could not join the tutorial, the material has been made available on Kitware’s website at the tutorial page: http://www.kitware.com/cvpr2012.html.

AVOGADRO IN JOURNAL OF CHEMINFORMATICS

“Avogadro: An advanced semantic chemical editor, visualization and analysis platform” was recently published in Volume 4 of the Journal of Cheminformatics. The paper was authored by Marcus D Hanwell, Donald E Curtis, David C Lonie, Tim Vandermeersch, Eva Zurek, and Geoffrey R Hutchinson, and discusses Avogadro 1.0.

Avogadro, the open-source, cross-platform molecule editor and visualization platform is designed for use in computational chemistry, molecular modeling, bioinformatics, materials science, and related research areas. The paper describes the platform’s direct applications in research and education as they relate to chemistry, physics, materials science, and biology, and includes examples on how to extend the tool via plugins. The Avogadro platform is being developed further as part of the Open Chemistry initiative at Kitware.

ITK CHAPTER IN AOSA BOOK VOLUME II

Last year, four Kitware authors contributed two chapters, one on VTK and another on CMake, to “The Architecture of Open Source Applications.” In the next edition, “The Architecture of Open Source, Volume II: Structure, Scale, and a Few More Fearless Hacks,” Luis Ibáñez and Brad King contributed a chapter on the Insight Segmentation and Registration Toolkit (ITK). This chapter covers what ITK is, its architectural features, and lessons learned during development.

The book is edited by Greg Wilson and Amy Brown, and features chapters on many other open-source projects. The book is available online or as a printed copy through Lulu and is distributed under the Creative Commons BY license.

KITWARE WINS NIH PHASE II SBIR AWARD

In August, Kitware announced a new Phase II SBIR grant from the National Institutes of Health to develop a computational infrastructure for mapping the mammalian connectome. The nearly $1M project is a collaborative effort between Kitware and Harvard University.

A human brain is estimated to have roughly 100 billion neurons connected through more than 100 thousand miles of axons and a quadrillion synaptic connections. The neural circuits within each brain are its connectome; understanding how the connectome works to enable cognition, consciousness, and intelligence is one of the most fundamental questions in science.

KITWARE #1245 ON THE 2012 INC. 5000 LIST

In August, Inc. Magazine announced its 2012 Inc. 5000 list, an exclusive ranking of the nation’s fastest-growing private companies. Kitware ranked #1245 overall, with three year revenue growth of 248%.

This is the fifth year that Kitware has been featured on the Inc 5000 list, and is Kitware’s highest rank to date! With an industry ranking of #100, Kitware attributes its success to its open-source business model and support of talented employees who are energized about the company’s research initiatives and mission.

THE PARAVIEW GUIDE PUBLISHED AS AN EBOOK

The ParaView Guide was released as an eBook this summer, and is now available from Kitware’s website. This edition of the ParaView Guide has been updated for ParaView 3.14 and is Kitware’s first foray into the world of electronic publishing. The content of the revised ParaView Guide comes from the ParaView wiki, with additional chapters covering CFD post-processing, an introduction to visualization with ParaView, and visualization and analysis of astrophysical AMR datasets. As an eBook, Kitware will be able to update and disseminate new versions of the Guide more quickly and efficiently to keep pace with real-world, digital demand. Additionally, users who purchase the eBook can upload and access it from any number of their electronic devices.

If you have any feedback about the eBook or topics you’d like to see in future editions, please let us know by emailing editor@kitware.com.
Advances in modern technology have made it easier to collect and prepare tissue samples for high-resolution scanning. This leads to digital representations of the connectome that are currently 60-70 terabytes in size, with this expected to grow rapidly as imaging technology improves.

Charles Law, Principal Investigator, will lead the development of a computational infrastructure that will include a set of segmentation algorithms that can extract neurons and their synapses from electron micrographs of serial tissue sections, and support arbitrarily large image volumes that will allow researchers to trace neural processes. Additionally, the suite will include fusion methods that take segmentation results as input and generate tracings of neural processes by linking segmentation results from one section to the next.

KITWARE WINS PHASE II FUNDING TO FURTHER DEVELOP CLIMATEPIPES

Kitware was awarded nearly $1M in Phase II funding from the U.S. Department of Energy to further develop ClimatePipes, a platform for providing non-researchers with access to and analysis tools for understanding long-term climate change projections. This is a collaborative project with researchers at the Polytechnic Institute of New York University.

The pace of global climate change is expected to accelerate and impact a variety of fields, including public health, infrastructure, and water resources. In recent years, multitudes of climate data have been collected as part of the U.S. Global Change Research Program and the Coupled Model Intercomparison Project (CMIP); however, hurdles including large data size, a lack of adequate metadata, poor documentation, and insufficient computational and visualization resources have limited research progress in this area.

Upon completion, ClimatePipes will make such valuable climate data available to non-researchers including policy makers, health officials, agriculturists, and industry leaders. Users will be able to run complex computations and visualizations in the cloud or on a user-provided cluster, and upload data to a server for analysis and integration.

In Phase I, ClimatePipes was developed as a web-based workflow with interfaces for accessing, querying, and visualizing datasets from one-or-more sources. In Phase II, Kitware will implement support mechanisms for more elaborate and relevant queries, and improve the system’s usability, robustness, and scalability. Advanced querying will be supported by a semantic search tool that relies on natural language processing techniques.

This work has been supported by the Department of Energy (DOE) under award number DE-SC0006493.

KITWARE PRESENTS AT OSCON 2012

Kitware team members attended OSCON, the Open Source Convention, this summer, where they gave a presentation on OSEHRA, the Open Source Electronic Health Record Agent, and mobile 3D visualization.

Luis Ibáñez and Rick Avila, with Seong Ki Mun of OSHEHRA, presented “OSEHRA - Building an Open Source EHR for All” on July 19th as part of OSCON’s Healthcare track. The presentation covered the creation of OSEHRA in 2011, the background of the VistA EHR system, and how OSEHRA is working to create an open source ecosystem in which VistA can continue to thrive.

Pat Marion, one of the principal developers of VES, and Utkarsh Ayachit, one of the principal developers of ParaViewWeb, presented “Mobile 3D Visualization.” They described the architecture, API, and usage of these systems, and provided real-time demonstrations, including client interface to a computing cluster. They also demonstrated the power of these tools through examples such as exploring a brain atlas on a phone, and interactive streamline generation and visualization on a tablet.

OPEN CHEMISTRY AT ACS MEETING

In August, Marcus Hanwell and Kyle Lutz attended the American Chemical Society meeting and presented some of the latest developments in Avogadro and Open Chemistry. Marcus presented some of the history and recent developments in “Avogadro, Open Chemistry and Semantics” at the Skolnik Symposium in the morning session. Kyle presented “Exploring Large Chemical Data Sets” in the afternoon session, where he showcased our recent efforts in chemical informatics on the desktop. They also presented a poster summarizing the current work being done at Kitware on Open Chemistry.

UPCOMING CONFERENCES AND EVENTS

Introduction to Midas Online Course
October 25 from 2:00-3:30PM. Kitware is offering a free online tutorial that will cover the use of the Midas Platform.

Pathology Visions 2012
October 28-21 in Baltimore, MD. Charles Law is a co-author on the paper “Development of a Web-Based Digital Pathology System Used for Dermatopathology Teaching,” which will be presented by Dr. Beverly Faulkner-Jones.

ACM Multimedia
October 29-November 2 in Nara, Japan. Sangmin Oh will attend.

27th International Conference on Screening for Lung Cancer
November 4-5 in New York, NY. Rick Avila will attend.

CMake and Friends: Open Source Tools to Build, Test & Package Software
November 7th in Clifton Park, NY. The day-long course will cover a variety of topics and be taught by Bill Hoffman.

Full-Motion Video Conference
November 8-9 in San Diego, CA. Matt Turek will attend.

Supercomputing 2012
November 10-16 in Salt Lake City, UT. Kitware will be exhibiting at SC12 at Booth 3951. We will also be presenting two collaborative tutorials on Sunday: Large Scale Visualization with ParaView, and In-Situ Visualization with Catalyst. Lisa Avila, Utkarsh Ayachit, Andy Bauer, Berk Geveci, Marcus Hanwell, Chris Harris, Rob Maynard, Bob O’Bara, Katie Osterdahl and Patrick O’Leary will attend.

RSNA 2012
November 25-30 in Chicago, IL. Rick Avila and Wes Turner will attend.

Scientific Software Day
December 17 in Austin, TX. Hosted by the Texas Advanced Computing Center, the Scientific Software Day is a day of discussions on the software that scientists use and that they develop as part of their research. Will Schroeder will attend.
NEW HIRES

Roni Choudhury
Roni Choudhury joined the Clifton Park, NY office as an R&D Engineer. He recently received his Ph.D. in computer science from the University of Utah, where his dissertation focused on “Visualization of Program Behavior via Memory Reference Traces.” He also holds a B.S. in computer science and mathematics from the University of Chicago. Prior to joining Kitware, Roni worked as a graduate research assistant in various roles at the University of Utah.

David Thompson
David Thompson joined Kitware as an R&D Engineer at the Carrboro, NC office. He holds a Ph.D. from the University of Texas at Austin, where he also received his M.S. in engineering. Prior to joining Kitware, David was with Sandia National Labs, where his work included developing visualization techniques for higher-order finite elements and monitoring HPC platforms to detect and statistically characterize failures.

Johan Andruejol
Johan Andruejol joined Kitware as an R&D Engineer on the medical imaging team at the Carrboro office, where he previously interned. Johan is in his fifth year completing his degree at the École Supérieure de Chimie Physique Électronique de Lyon (CPE Lyon), with a focus on image synthesis and processing.

KITWARE INTERNSHIPS

Kitware Internships provide current college students with the opportunity to gain hands-on experience working with leaders in their fields on cutting edge problems. Our business model is based on open source software—an exciting, rewarding work environment.

Our interns assist in developing foundational research and leading-edge technology across six business areas: supercomputing visualization, computer vision, medical computing, data management, informatics and quality software process. We offer our interns a challenging work environment and the opportunity to attend advanced software training. To apply for an internship, please visit our employment site at jobs.kitware.com and submit a resume and cover letter through our online portal.

EMPLOYMENT OPPORTUNITIES

Kitware is seeking talented, motivated and creative individuals to fill open positions in all of our offices. As one of the fastest growing companies in the country, we have an immediate need for software developers and researchers, especially those with experience in computer vision, scientific computing and medical imaging.

At Kitware, you will work on cutting-edge research alongside experts in the field, and our open source business model means that your impact goes far beyond Kitware as you become part of the worldwide communities surrounding our projects.

Kitware employees are passionate and dedicated to innovative open-source solutions. They enjoy a collaborative work environment that empowers them to pursue new opportunities and challenge the status quo with new ideas. In addition to providing an excellent workplace, we offer comprehensive benefits including: flexible hours; six weeks paid time off; a computer hardware budget; 401(k); health, vision, dental and life insurance; short- and long-term disability, visa processing; a generous compensation plan; yearly bonus; and free drinks and snacks. For more details, visit our employment site at jobs.kitware.com

Interested applicants are encouraged to visit our employment site at jobs.kitware.com and submit a resume and cover letter through our online portal.


In addition to providing readers with updates on Kitware product development and news pertinent to the open source community, the Kitware Source delivers basic information on recent releases, upcoming changes and detailed technical articles related to Kitware’s open-source projects.

For an up-to-date list of Kitware’s projects and to learn about areas the company is expanding into, please visit the open source pages on the website at http://www.kitware.com/opensource/provensolutions.html.

A digital version of the Source is available in a blog format at http://www.kitware.com/source.

Kitware would like to encourage our active developer community to contribute to the Source. Contributions may include a technical article describing an enhancement you’ve made to a Kitware open-source project or successes/lessons learned via developing a product built upon one or more of Kitware’s open-source projects. Kitware’s Software Developer’s Quarterly is published by Kitware, Inc., Clifton Park, New York.

To contribute to Kitware’s open-source dialogue in future editions, or for more information on contributing to specific projects, please contact us at editor@kitware.com.


Graphic Design: Steve Jordan
Editors: Katie Sharkey, Katie Osterdahl

To contribute to Kitware’s open-source dialogue in future editions, or for more information on contributing to specific projects, please contact us at editor@kitware.com.


In addition to providing readers with updates on Kitware product development and news pertinent to the open source community, the Kitware Source delivers basic information on recent releases, upcoming changes and detailed technical articles related to Kitware’s open-source projects.

For an up-to-date list of Kitware’s projects and to learn about areas the company is expanding into, please visit the open source pages on the website at http://www.kitware.com/opensource/provensolutions.html.

A digital version of the Source is available in a blog format at http://www.kitware.com/source.

Kitware would like to encourage our active developer community to contribute to the Source. Contributions may include a technical article describing an enhancement you’ve made to a Kitware open-source project or successes/lessons learned via developing a product built upon one or more of Kitware’s open-source projects. Kitware’s Software Developer’s Quarterly is published by Kitware, Inc., Clifton Park, New York.


In addition to providing readers with updates on Kitware product development and news pertinent to the open source community, the Kitware Source delivers basic information on recent releases, upcoming changes and detailed technical articles related to Kitware’s open-source projects.

For an up-to-date list of Kitware’s projects and to learn about areas the company is expanding into, please visit the open source pages on the website at http://www.kitware.com/opensource/provensolutions.html.

A digital version of the Source is available in a blog format at http://www.kitware.com/source.

Kitware would like to encourage our active developer community to contribute to the Source. Contributions may include a technical article describing an enhancement you’ve made to a Kitware open-source project or successes/lessons learned via developing a product built upon one or more of Kitware’s open-source projects. Kitware’s Software Developer’s Quarterly is published by Kitware, Inc., Clifton Park, New York.


In addition to providing readers with updates on Kitware product development and news pertinent to the open source community, the Kitware Source delivers basic information on recent releases, upcoming changes and detailed technical articles related to Kitware’s open-source projects.

For an up-to-date list of Kitware’s projects and to learn about areas the company is expanding into, please visit the open source pages on the website at http://www.kitware.com/opensource/provensolutions.html.

A digital version of the Source is available in a blog format at http://www.kitware.com/source.

Kitware would like to encourage our active developer community to contribute to the Source. Contributions may include a technical article describing an enhancement you’ve made to a Kitware open-source project or successes/lessons learned via developing a product built upon one or more of Kitware’s open-source projects. Kitware’s Software Developer’s Quarterly is published by Kitware, Inc., Clifton Park, New York.