

## General Stimulus Features



### Multi-modal Stimulus Delivery

Presentation can deliver 2-d images, 3-d graphics, compressed video, and audio - all at the same time. Play multiple videos and multiple sounds, mixed with any number of many graphical elements. You are only limited by the power and speed of your computer system.

code	time	time_uncertainty
pic 1	-983.3	0.1
pic 2	-966.6	0.1
pic 3	-950.3	0.9
pic 4	-933.3	0.1
pic 5	-916.7	0.1
pic 6	-900	0.1
pic 7	-883.3	0.1

### Power and Precision

Presentation takes advantage of modern PC gaming technology to provide powerful and flexible stimulus displays, while at the same time accurately logging all stimulus and response events. The foundation of Presentation and all features are designed and implemented with precision in mind.

## General Visual Stimulus Features



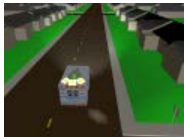
### Multiple Graphical Elements

Arbitrarily many graphical elements of any type may be combined or displayed simultaneously - video, 3D, images, generated 2D graphics.

event_type	code	duration
Picture	pic 1	16.7
Picture	pic 2	16.7
Picture	pic 3	16.6
Picture	pic 4	16.7
Picture	pic 5	16.7
Picture	pic 6	16.6
Picture	pic 7	16.7

### Vertical Refresh Synchronization

All display changes are synchronized with the vertical refresh of the video signal. Event times for visual events are the start of the scan signal containing the change.



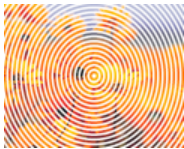
### Rapid Presentation and Frame Control

For most types of stimuli, the display can be changed on every vertical refresh. Stimulus scheduling can be done to achieve desired refresh count durations.



### Any Resolution

You may run at any display resolution that your video card supports.



### Transparency

Transparency can be used and applied to all types of visual stimuli - images, generated 2d graphics, 3d graphics, even videos.



### Color Profile Control

Presentation can apply a color profile (icm) to control the color balance of your display. You can even programmatically alter the video card color calibration table at run time.



### Multiple Independent Stimulus Displays

In addition to an experimenter control screen, you may have arbitrarily many stimulus displays displaying independent stimuli, depending on your hardware.



### Custom Distance Units

All visual stimulus related parameters for an experiment may be specified in pixels, or in your own custom units, including visual angles.

## 2D Visual Stimuli



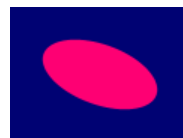
### Images

Load images from a variety of formats.



### Text

Generate text in any installed font. Unicode is supported.



### Ellipses

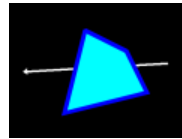
Generate ellipses with variable size, orientation, and color, including transparency.



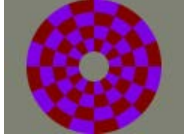
**Annuli**  
Generate annuli with variable size, orientation, and color, including transparency.



**Polygons**  
Generate polygons with variable size, side count, line color, line width, line corner types, and fill color, including transparency.



**Arbitrary Lines/Shapes/Arrows**  
Generate arbitrary shapes and arrows with variable lines, line width, line color, arrow head types, line corner types, line color, and fill color, including transparency.



**Checker Circles**  
Generate checker circle graphics with variable inner radius, outer radius, ring count, slice count, and colors. The highly optimized generation code is extremely fast.



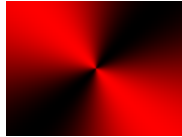
**Circular Gradients**  
Generate circular gradients or rings with variable period, phase, and colors. The highly optimized generation code is extremely fast.



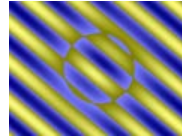
**Gaussians**  
Generate gaussians with variable width, and colors. The highly optimized generation code is extremely fast.



**Linear Gradients**  
Generate linear gradients or lines with variable period, phase, orientation, and colors. The highly optimized generation code is extremely fast.



**Radial Gradients**  
Generate radial gradients or wedges with variable count, phase, and colors. The highly optimized generation code is extremely fast.



**Combinations**  
Combine multiple graphics by masking, multiplying, averaging, adding, or min/max.

## 3D Visual Stimuli



**Multiple 3D Objects**  
Display arbitrarily many 3D elements placed and oriented in space.



**Camera Control**  
Control the camera position and direction. Create animations through the 3D space.



**Lighting Control**  
Control lighting properties of each object, and create light sources or different types placed and oriented in 3D space.



**Texture Control**  
Apply textures to surfaces of 3D objects.



**Import 3D Models**  
Load and display stored 3D models.



**3D Shapes**  
Generate planes, spheres, cylinders, cuboids, and cones.

**Stereoscopic Display**  
Video card support can automatically produce alternating frame stereo



displays from your 3D stimuli. For stereo devices with 2 inputs, you can use multiple stimulus displays to completely control the image to each eye.

## Compressed Video Stimuli



### Multiple Formats

Display compressed video using any DirectShow compatible codecs installed on your system.



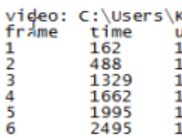
### Playback Control

Seek, start, stop or pause videos at specific times or video frames.



### Multiple Simultaneous Videos

Play multiple videos simultaneously, depending on hardware performance.



### Detailed Logging of Frames

Optional detailed logging of video playback, including duration, frame count, or even exact times for each frame.



### Speed Control

Specify video speed (without audio) by frequency, or refreshes per frame.



### Size/Location Control

Control video size and screen location.



### Mix with Other Stimuli

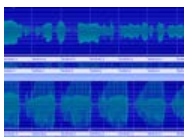
Display other 2D or 3D stimuli, and get input while videos are playing.



### Custom Display

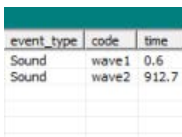
Control video display precisely including mapping to 3D objects, and using transparency.

## Auditory Stimuli



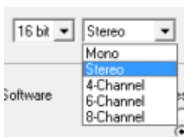
### Multiple Sounds

Play multiple auditory stimuli at the same time. Loop auditory stimuli indefinitely.



### Accurate Start Time Reporting

Sub-millisecond playback accuracy can be achieved for reported sound start times depending on hardware.



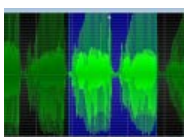
### Up to 8 Speakers with Attenuation Control

Independently attenuated and route multiple channels to up to 8 speakers, depending on hardware.



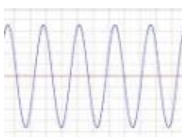
### Windows Volume Control

Automatically reset, or control at runtime the Windows volume setting.



### Cue Events

Generate events and port output from cue events stored in the waveform.



### Simple Sound Generation

Generate Linear combinations and products of sine waves, gaussians, and lines.

## Force Feedback Stimuli



### Direct X Compatible Device Support

Control force motors and effects on Direct X compatible force feedback devices.



### Multiple Effect Types

Use constant, ramp, periodic, or condition forces, depending on device capabilities.

## Response Monitoring

### Button Press Responses

Monitor button press or release responses from multiple devices at all

### Axis Positions

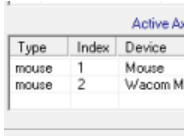
Programmatically access position information from joystick and mouse



times, even during high-rate stimuli.



devices.



### Multiple Mice

Simultaneously monitor and distinguish multiple mouse devices.



### Multiple Keyboards

Simultaneously monitor and distinguish multiple keyboard devices.



### HID Devices

Monitor and distinguish multiple mouse or keyboard-like HID USB devices.



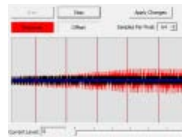
### DirectX Compatible Devices

Monitor button press responses and access axis information for DirectX Compatible controllers.



### Port Devices

Interpret code events from any supported port devices as button press responses.



### Sound Thresholding

Generating response events by thresholding audio input data. Audio data following events can optionally be saved to disk.



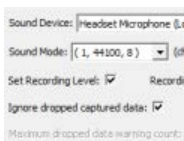
### Extension Devices

Plug-ins can be written to add built-in support for responses devices that Presentation does not natively support.

event_type	code	time	time
Key Input	104	-382.9	0.1
Key Input	101	-163.4	0.1
Key Input	108	-65.8	0.1
Key Input	109	64.2	0.1
Key Input	111	253.3	0.1
Key Input	119	393.4	0.1
Key Input	32	515.4	0.1
Key Input	119	753.3	0.1
Key Input	111	946.3	0.2

### Keyboard Input

Special functions allow collection of entered text, with optional automatic display, and detailed logging. (This is in addition to specific keyboard keys treated as response buttons.)



### Sound Recording

Record audio input data at specified times and store to disk.

## Programmability

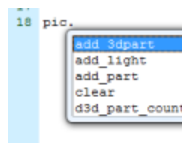
```

9 begin_pc1;
10
11 loop int i = 1 ur
12 begin
13 event1.set_eve
14 tr1.present();
15 i = i + 1;
16 end;

```

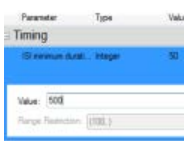
### Presentation Control Language

Simple programming language allows complete programmatic access to all Presentation features allowing arbitrarily complex experiments to be implemented.



### Full Featured Editor

The built-in multi-tab editor for programming experiments includes syntax highlighting, code completion, and method call hints.



### Customizable GUI

You may create a customized GUI so your research assistants, participants, or students can change the values of parameters within the experiment easily.



### Response Dependent Behavior

Automatic classification and pairing of stimuli and responses makes feedback easy, and programmability makes arbitrary response dependent behavior possible.



### Automated Features

Set up stimulus sequences with specified timing and run them automatically. Classification and pairing of stimuli and responses can be done automatically.



### Python Interfacing

The Presentation Python interface provides complete control over Presentation from your Python programs, including during the experiment. Thus, you can effectively program your experimental procedure in Python while still using all of Presentation's features.

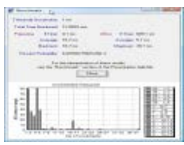
## Timing

### Designed for Precision

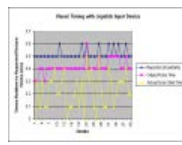
Presentation has a great variety of features for visual and auditory stimulus delivery, response

### Time Measurement Uncertainties

Presentation is architected from the ground up to detect system interruptions that affect the accuracy



monitoring, and interfacing. Presentation is specifically designed to allow simultaneous use of all of these features while still maintaining precise stimulus delivery and accurate event logging. For example, play multiple sounds, present multiple visual stimuli changing every vertical refresh, produce synchronized port output, and monitor responses occurring any time, all at the same time. If there *are* any performance issues for a particular system for a demanding experiment, you can address them *before* collecting data.



of time measurements. For all logged events, Presentation reports a system uncertainty value that reflects the accuracy of the time measurement for that event. This ensures that system interruptions or other Presentation functions do not compromise time measurement without notification. Typical uncertainty values are in the tenths of milliseconds range. Combined with knowledge of the display, audio, and response hardware timing characteristics, you can ensure accurate timing of your experiment.

## Data Reporting



**Detailed Logging**  
Data files contain time and system uncertainty measures for all logged events. You control what is logged and how it is identified. In addition to analysis, you can use this data to ensure the accurate presentation and monitoring of your stimuli and responses on every run.

Parameter	Value
Hits	4
Misses	0
Incorrects	1
False Alarms	0

**Event Classification**  
Automatically classify and associate stimuli and responses at run time, or use flexible post-processing features for general classification of events and computation of performance measures.

trial	colour	shape	time	time
1	yellow	circle	-999.9	0.1
2	red	square	-999.9	0.1
3	blue	square	-799.9	0.1
4	red	circle	-699.9	0.1
5	yellow	square	-599.9	0.1
6	blue	circle	-499.9	0.1
7	blue	square	-399.9	0.1

**Data Formatting**  
Flexible post-processing features allow you to produce custom formatted data files with only the information you need for input into your analysis tools.

trial	colour	shape	time	time
1	yellow	circle	-999.9	0.1
2	red	square	-999.9	0.1
3	blue	square	-799.9	0.1
4	red	circle	-699.9	0.1
5	yellow	square	-599.9	0.1
6	blue	circle	-499.9	0.1
7	blue	square	-399.9	0.1

**Custom Data Files**  
Programmatically write data files during an experiment to store whatever you like.

## Interfacing<sup>P</sup>



**Port Input Monitoring**  
Presentation will store values and times for signals received on supported ports. Those values can be accessed during the experiment as well as saved to data files.



**Event Port Output**  
Automatically send signals from supported ports synchronous with specified stimuli and responses.



**Standard Parallel Ports**  
Digital I/O port control (TTL value output, and polled input), and hardware interrupt handling for built-in parallel ports.



**Standard Serial Ports**  
Data output and input including programmatic sending and receiving of data strings.



**National Instruments Devices**  
Digital I/O port control (TTL value output, and polled input), monitoring of pulse triggered counters, single value analog output, and programmatic analog input monitoring.



**Measurement Computing Devices**  
Digital I/O port control (TTL value output, and polled input), and monitoring of some pulse triggered counters.



**Programmatic TCP/IP**  
Send and receive text data programmatically to a server over TCP/IP.



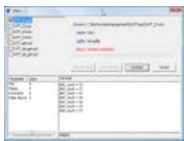
**Eye Tracker Interface**  
For eye tracking systems with a Presentation extension, receive real time eye tracking data and produce gaze dependent stimuli.

**Data Port Extensions**  
Immediately transmit detailed event information.





## Execution Options



### Experimenter Feedback

When using multiple monitors, or during non-visual experiments, Presentation can display customizable feedback to the experimenter. Report progress, performance, or anything you like.



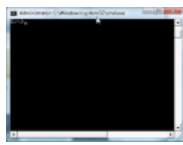
### Presentation IDE

The main Presentation application is used to develop and run experiments.



### Package Player

The Package Player offers a simplified interface for running securely packaged experiments (see below).



### Command Line and Shortcuts

The command line launcher application can be used to run experiments, or parts of experiments from the command line, scripts, or shortcuts.



### Programmatic Control

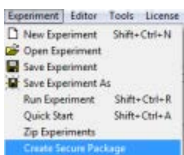
The Presentation Control API allows you to load and run Presentation experiments programmatically from your own program.



### PowerPoint

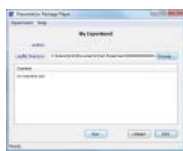
Run a section of an experiment directly from a button on a PowerPoint slide.

## Experimental Site Features



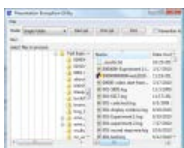
### Secure Packages

Presentation experiments may be packaged into a single file for distribution. Experiment control files are encrypted, and all files are validated each time the experiment is run.



### Package Player

The Package Player is a simplified application for executing securely packaged experiments.



### Data Encryption

Presentation can optionally encrypt experimental data files, including programmatically written text files. Graphical and command line decryption utilities are installed with Presentation.



### Automatic FTP Upload

Presentation can automatically upload all data files produced during an experiment to your FTP server.



### Experiment Licensing

An experiment license allows the execution of one experiment at a time on an unlimited number of computers connected to the internet. The licensed experiment can be switched at any time during the duration of the license.

## Extensibility



### Extensions

Presentation extensions are "plug-ins" which add functionality to Presentation. Some Presentation extensions are available from 3rd parties, such as device manufacturers. In addition, anyone can write a Presentation extension using the Presentation SDK. Presentation can interface with several types of extensions described in this section.



### Response Device Extensions

This type of extension adds built-in support for response devices that Presentation does not natively support.



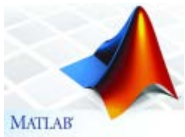
### **PCL Extensions**

This type of extension adds custom data types to the Presentation Control Language (PCL). This allows completely customized interfaces to external software or hardware.



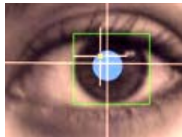
### **Workspace Extensions**

This type of extension implements a generalized interface to external software. It allows sending commands to external software and the importation of graphic or sound data into Presentation at run time.



### **Workspace Extension for Matlab**

The workspace extension for Matlab allows running arbitrary commands or scripts in the Matlab engine, and importing graphic or sound data into Presentation at run time. This extension is also open source.



### **Eye Tracker Extensions**

This type of extension provides access to eye tracker data at run time, and interaction with the eye tracking system.