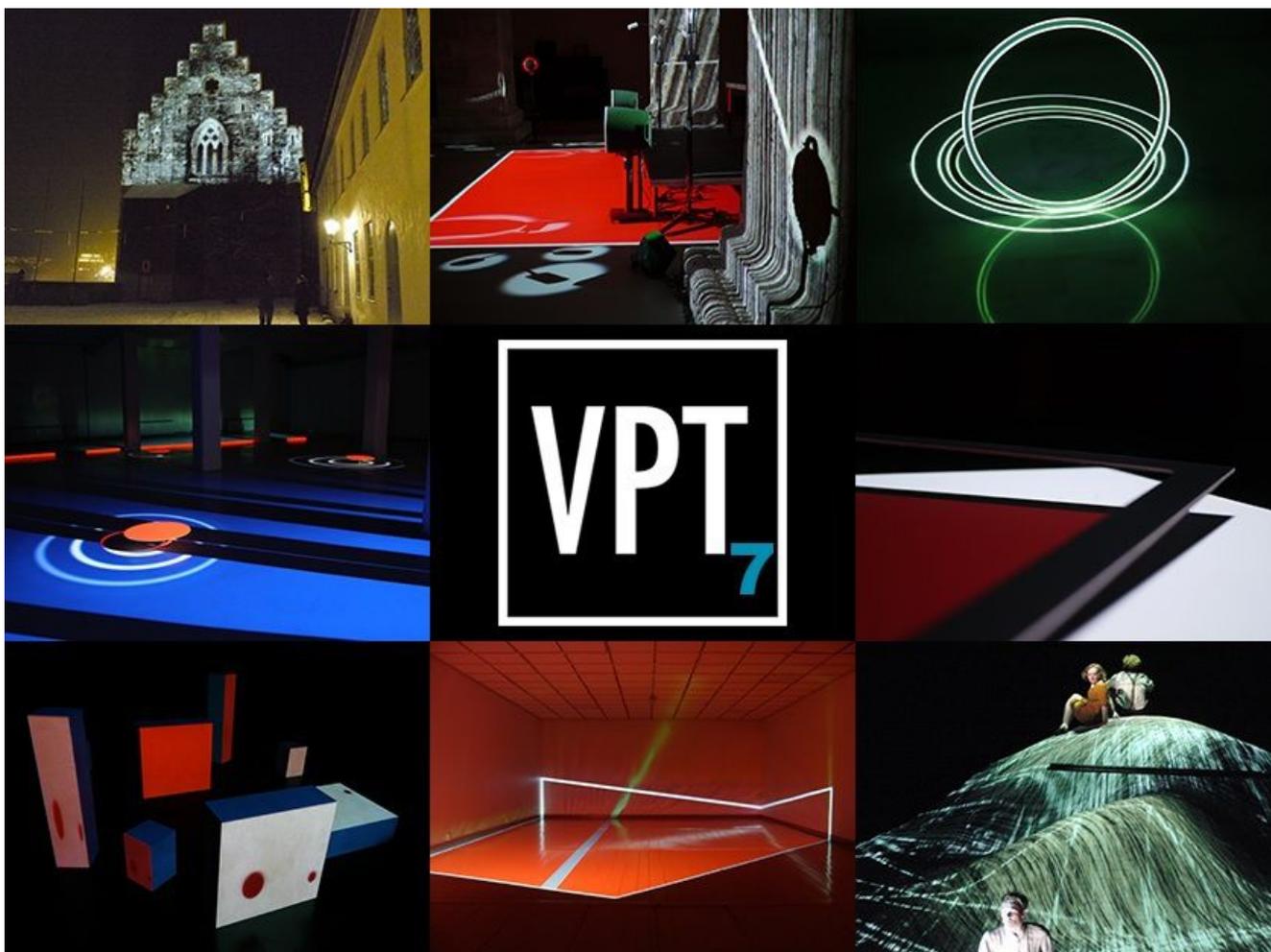


日本語 VPT7 ドキュメンテーション v0.3

Original document: [vpt7_Documentation.pdf](http://vpt7_documentation.pdf)

日本語訳 by 高橋亘 2015/09/19



VPT7 ドキュメンテーション

HC Gilje は、VPT7 を 2013 年 11 月にリリースしました。

ビデオプロジェクションツール VPT7.0、映像投影用強力パッケージソフト:

VPT (VideoProjectionTool) は、無料の多目的リアルタイム投影ソフトウェアで、Mac 用と Windows 用があります。

VPT は、複雑な形状に動画映像を投影したり、特定の空間/表面に投影を適合させたり、記録映像とライブ映像を混在させたり、マルチスクリーンに HD 映像を再生させたり、Arduino (アルディーノ) のセンサーやカメラトラッキングを使用した対話型組み込みソフトなどの用途に使われます。

VPT は、劇場や組み込み用ソフト向けに人気のあるツールとなりましたが、VJ (ビデオジョッキー) からも利用されています。以前のバージョン VPT6 は 20000 回以上ダウンロードされました。VPT7 は以前のバージョンよりも、より速く、より安定で、より簡単に柔軟に使用できるよう再構築されています。そして、まだ無料です!

新機能の一部を以下に示す:

追加可能なレイヤー: プロジェクトに必要な複数のレイヤーを追加できます。

柔軟な映像ソースのセットアップ: quicktime、HAP、still、mix モジュールなどの混合映像ソースをプロジェクト (ライブ、ソリッドおよび syphon ソース上) の必要に応じて作成できる。

Mac 版では VPT は syphon フレームワークをサポートし続けますが、HAP コーデック (超高解像度ビデオのオープンソースのコーデック) のサポートも追加されました。

新しいマスクエディタがあります。これは必要に応じてできるだけ多くのマスク・ポイントを追加することができます。

レイヤーのコーナーピン調整機能が改善され、より良い歪訂正を実現しています。

メッシュエディタもまた新しくなり、コーナーピン歪み (前のバージョンでは非対応) とメッシュ歪みとを一緒に対応できます。コントローラーとキューリストの能力が拡張されました。

ソースのプリセットとソースのプレイリストを使う伝統的 VPT のプリセットに対応するため、オプションの代替手段があります。

System Requirements

Mac

Mac Intel machine running OS X 10.5 or later, and 1 GB RAM. Jitter requires QuickTime 7.1 (or later), an OpenGL-compatible graphics card, and OpenGL 1.4 (or later). For syphon and hap: minimum OS X 10.6

Windows

Windows XP, Vista, or Windows 7 machine with a Pentium 4R or CeleronR compatible processor or higher and 1 GB RAM. Jitter requires QuickTime 7.1 (or later), an OpenGL-compatible graphics card, and OpenGL 1.4 (or later).

VPT7 インターフェース

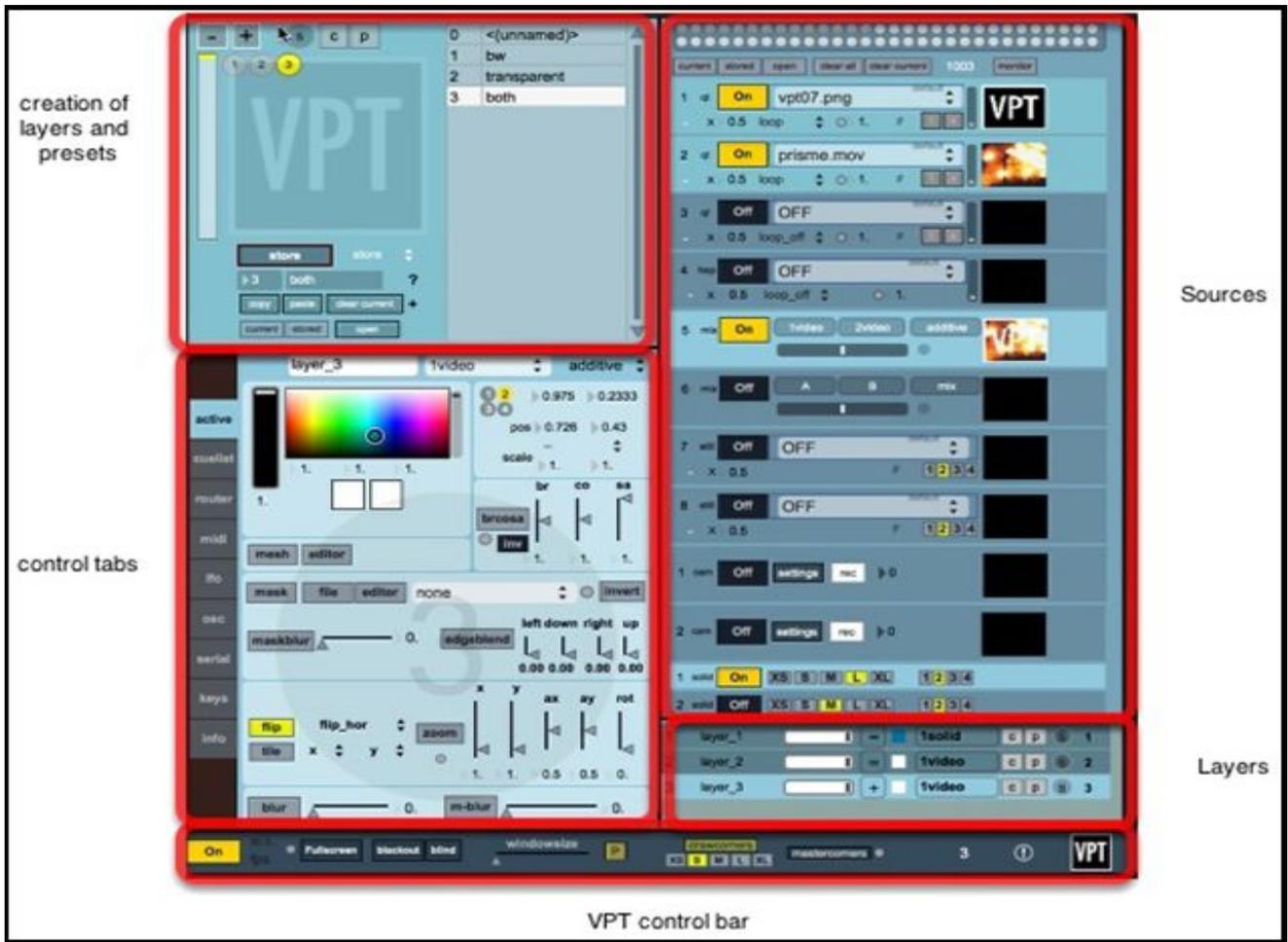
VPT7は、インターフェースウィンドウ、プレビューウィンドウと出力ウィンドウで構成されています。出力ウィンドウは、お使いのプロジェクターの出力画面となります。フルスクリーンにするためには、インターフェイスウィンドウの下部にある” fullscreen” ボタンをクリックするか、キーボードの[ESC]キーを押してください。



インターフェースウィンドウ概要

レイヤーとプリセット作成

ソース

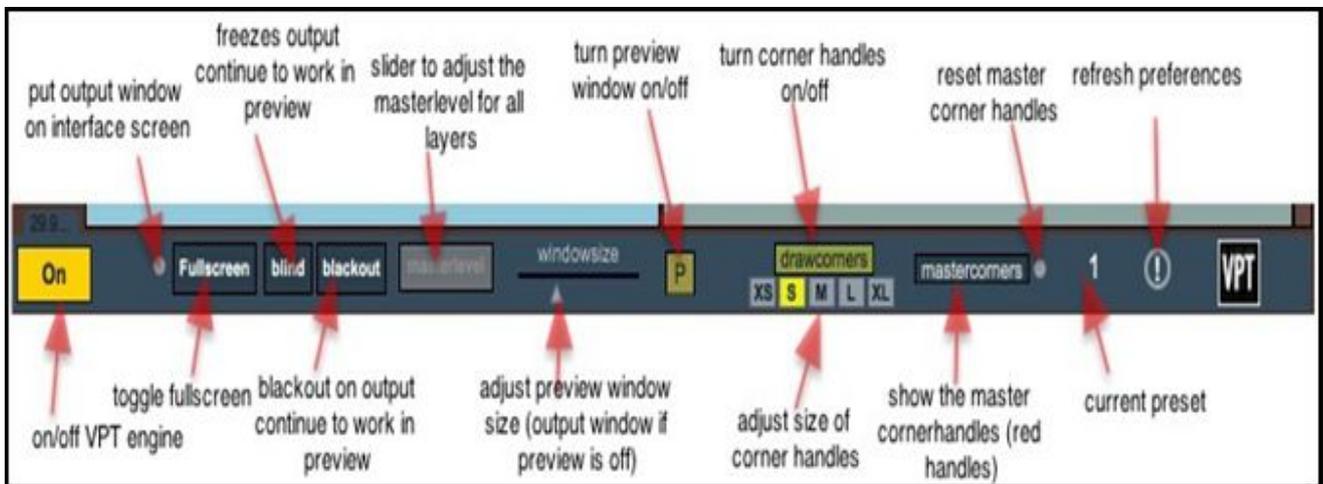


コントロール タブ

レイヤー

VPT コントロールバー

VPT コントロールバー



クイックスタート

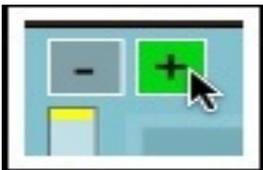
初めて VPT を開くと、たぶんほとんど動いておらず、インターフェイスウインドウの左上の部分は以下のようになっているでしょう。VPT7 ではレイヤー数は固定ではなく、プロジェクトに必要なレイヤーを追加できます。



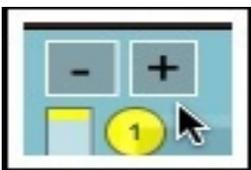
レイヤーとプリセット作成部

レイヤーの作成

+ボタンをクリックしてレイヤーを作成します。



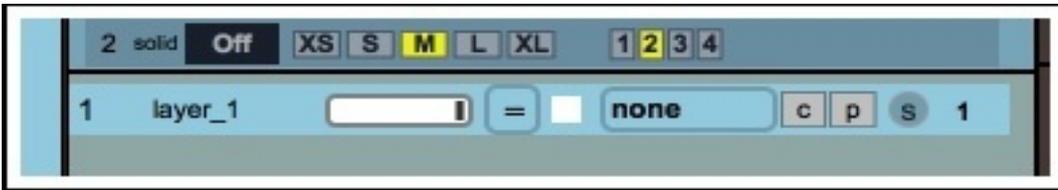
レイヤー番号が書かれた黄色い円が、+ボタンのすぐ下に表示されます。



プレビューと出力ウィンドウでは、レイヤーのコーナーハンドルが表示されますが、レイヤーはまだ映像ソースを持っていないので、真っ黒になっています。



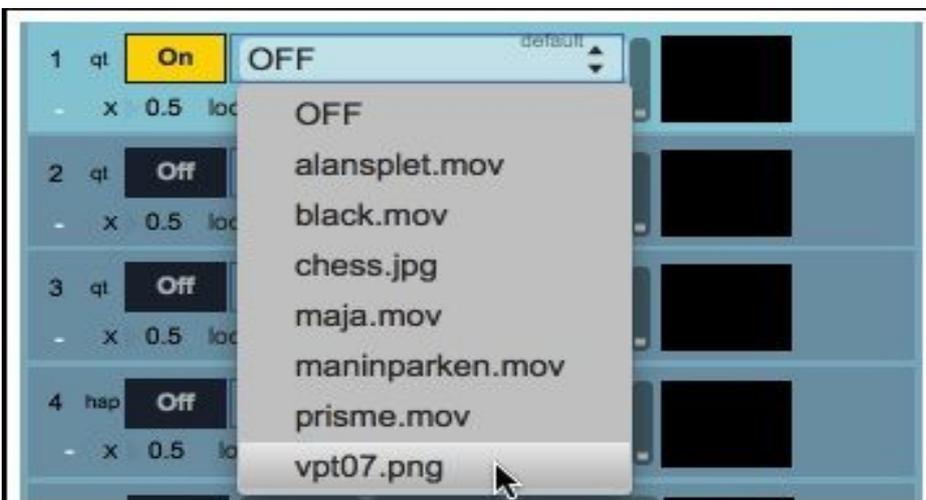
VPT インターフェースウィンドウの右下のセクションでは、レイヤー概要として、最初は新規レイヤーに関する情報だけです。



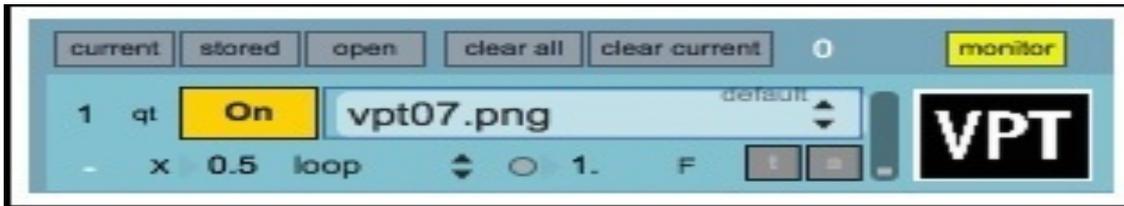
レイヤーに映像ソースを追加する

レイヤーに映像コンテンツを埋め込むためには、最初に、ソースセクション（右上部分）にある映像ソースのいずれかを有効にする必要があります。

まず、最初のソース（1 qt のラベル）の on ボタンをクリックして開始し、on ボタンの右側エリア（そこはおそらく OFF と表示）をクリックしてください。利用可能な映像ディレクトリのメニューが表示されます。いずれかのアイテムを選択します。メニューのアイテムは、フォルダ内の特定の映像ソースに割り当てられたビデオを指しています。デフォルトでは、このフォルダは VPT の初期デフォルトプロジェクトフォルダ内のビデオフォルダの中にあります。



選択した映像ソースを見るにはモニターボタンをクリックすると、映像ソース名横の小さなウィンドウには、現在選択されている映像メディアが表示されます。



新規レイヤーで利用選択した映像ソースを使うために、レイヤーに使用する映像ソースがどれか示する必要があります。レイヤーセクションでは、nonの部分をクリックすると利用可能な映像ソースのメニューが表示されます。最初の8つの映像ソースは1video、2video等のラベルが付いています。これらの映像ソースは異なるタイプのビデオソース（QuickTime、still、hap、ミックス）を含むことができるが、常に、より一般的な名前前の1video、2video等で参照される。下図では1videoを使用しています。この選択で、VPTアイコン（または他の選択映像メディア）がプレビューと出力ウィンドウに表示されます。



コーナーハンドルをドラッグしてレイヤーを配置する

プレビューまたは出力ウィンドウのいずれかでハンドルをクリックやドラッグしレイヤーを配置します。もしコーナー上のコントロールを失った（レイヤーの中間点を越えてドラッグした時に発生する可能性があります）場合は、**アンドゥ機能 Ctrl-Z**を使用して前のコーナー位置に戻せます。（注）アンドゥ機能はプレビュー/出力ウィンドウの内側のみで動作する。レイヤーを移動するには、レイヤー上で **Shift-click** しながらドラッグします。レイヤーを拡大縮小するには、**ALT-click** しながらドラッグします。



作成したものを保存

これで、出力の上にレイヤーを作成し、コンテンツを与え、配置しました、あなたは出力を保存したくなるでしょう。多くのプログラムとは異なり、VPT では、ファイル保存は作業を保存する方法ではありません。VPT の基本は、プリセットを保存することです。プリセットは VPT の現在の状態についての情報を保存するので、基本的には、出力で見るものを保存することになります。VPT は、リアルタイムで出力をレンダリングするリアルタイムアプリケーションなので、プリセットを保存するときに出力の画像またはビデオファイルを保存しませんが、レイヤーの位置や使用されている映像ソース等に関する情報が保存されます。プリセットを保存するには、ユニークな番号と記載名称（ただしオプション）を与える必要がありますので、この例では、プリセット番号に 1（0 は特殊なケースである、プリセットの保存番号に使わない）を使用し、プリセット名に VPT-layer を使用しました。store ボタンをクリックしてプリセットを保存します。もし、別のプリセットを同じ番号・別の名前で作成した場合は、古いプリセットを上書きします。これを回避するために、store ボタンの右にあるメニューをクリックして、store ボタンの動作を変更することができます。storenext を選択した場合、VPT は既存のプリセットの上書きを避け、次の利用可能なスロットにプリセットを保存します。しかしながら既存のプリセットを編集したい場合は、おそらく通常の store モードに戻したくなるでしょう。



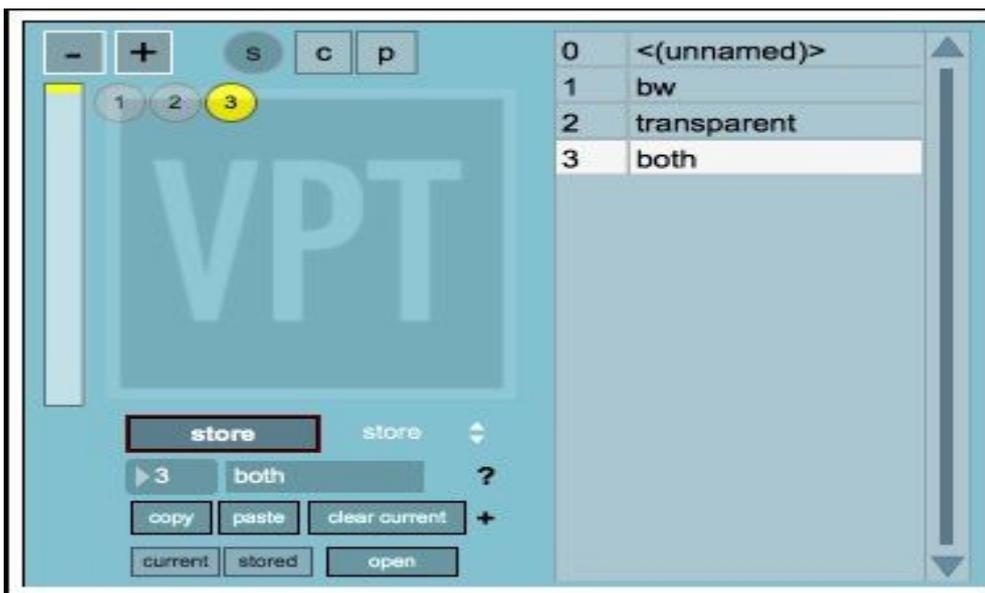
保存されたプリセットは、プリセットリストに表示されます。これで、VPT を閉じた後、次に VPT を開いたときには、プリセットリストでクリックによりプリセット 1 を選択することができます。



より多くのレイヤーを追加する

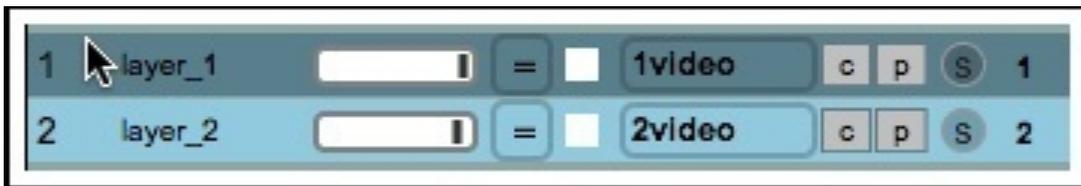
より多くのレイヤーを追加することは、最初のを追加するのと同じく簡単で、+ボタンをクリックするだけです。

注意事項： 新たなレイヤーは常に以前のものの上に追加されているので、新しいレイヤの位置調整、透明度調整、コンテンツ付与を行う前は、は、下の層が不明瞭になります。もし、プリセットを保存した後にレイヤを追加した場合は、それに応じて調整した（通常は新しいレイヤーの透明度を 0 に設定すること）レイヤーとプリセットをを再保存する必要があります。これは、古いプリセットは後から追加されたレイヤに関する情報を持っていないからです。



アクティブレイヤー

アクティブレイヤーの概念はVPTにおいて重要です、一度に1つのレイヤーでしか作業できないので、あなたが作業しているレイヤーを明確にする必要があります。この情報は、多くの場所にありますが： アクティブレイヤーは、プレビュー/出力ウィンドウでコーナーハンドラを持つレイヤーである。左上のセクションにある黄色い丸は、アクティブレイヤーを示し、レイヤーセクションでは水色の背景がアクティブレイヤーを示している。レイヤーの間を移動する（アクティブレイヤーが何であるかの変更）には、レイヤーセクションの左側の数字か左上のセクションで、円をクリックする、あるいはレイヤー（1-9）間でレイヤーを切り替える数字キーを使用する。 また、レイヤーをサイクリックに巡回する[Tab]キーを使用することもできます。



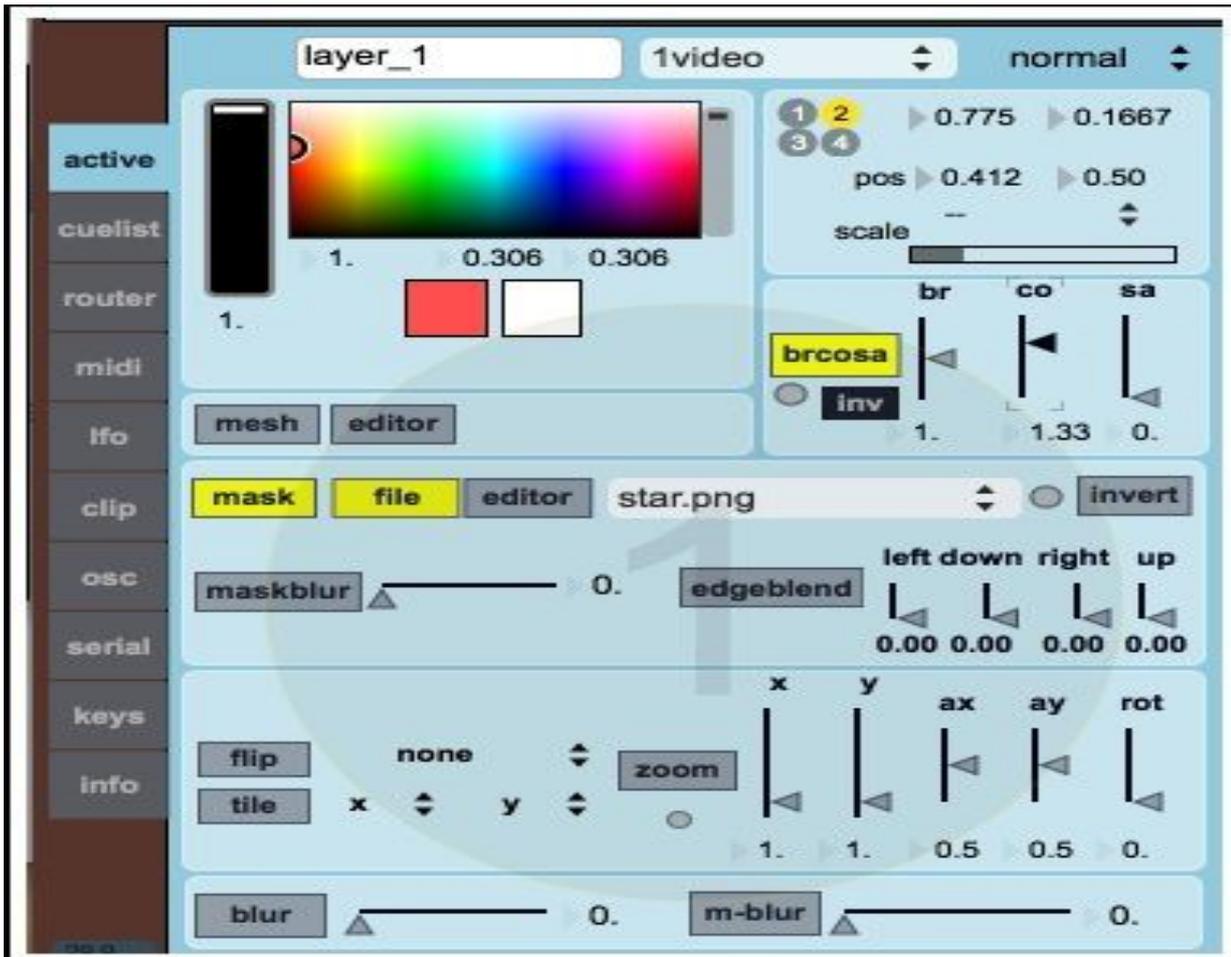
レイヤーリストでより多くのレイヤーを見る

いくつかのレイヤーを表示するために起動した場合、右下のレイヤー概要セクションは、多少混雑しているかもしれません。このセクションの右側で黒い線をドラッグすることで、簡単にソースセクションとレイヤーセクション間の分割位置を調整することができます。



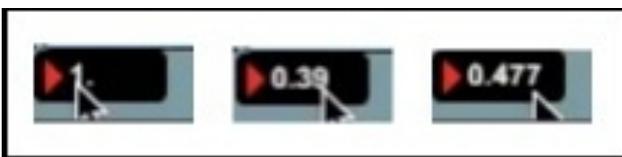
active タブ

VPT 左下部分のコントロールタブセクションには、レイヤー、映像ソースおよびプリセットのパラメータを制御するためのオプションがたくさんあります。タブの名前をクリックして選択します。active タブはアクティブレイヤーに関連付けられており、レイヤーに関連する透明度、色、幾何学紋様やマスクなどのパラメータを設定したり調整すること（マスクエディタを使用するように簡単に）ができます。また、明るさ/コントラスト/彩度、メッシュの歪み、縮尺、フリップとにじみ、モーションにじみとともに紋様の回転を調整する方法も多数あります。



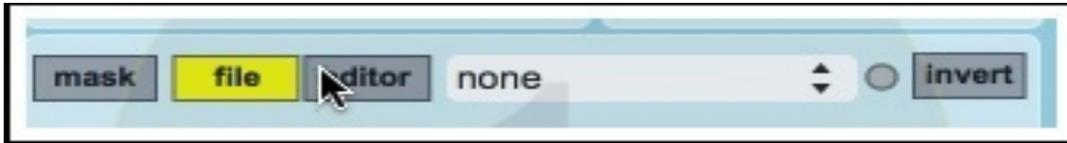
VPT で数値を調整することについての注意

数字をクリックすると、それがアクティブなパラメータであることを意味する赤い三角形が表示されます。これで、値を入力するキーボードを使用することができます。マウスを使用して値を変更するには、数字の上でクリックとドラッグを行います。クリックして、値を減少させるには下にドラッグ、値を大きくするには上にドラッグします。クリックする数字の場所が数字の精度/分解能を決定する。さらに、小数点の後ろではより多くの数字をクリックできます。

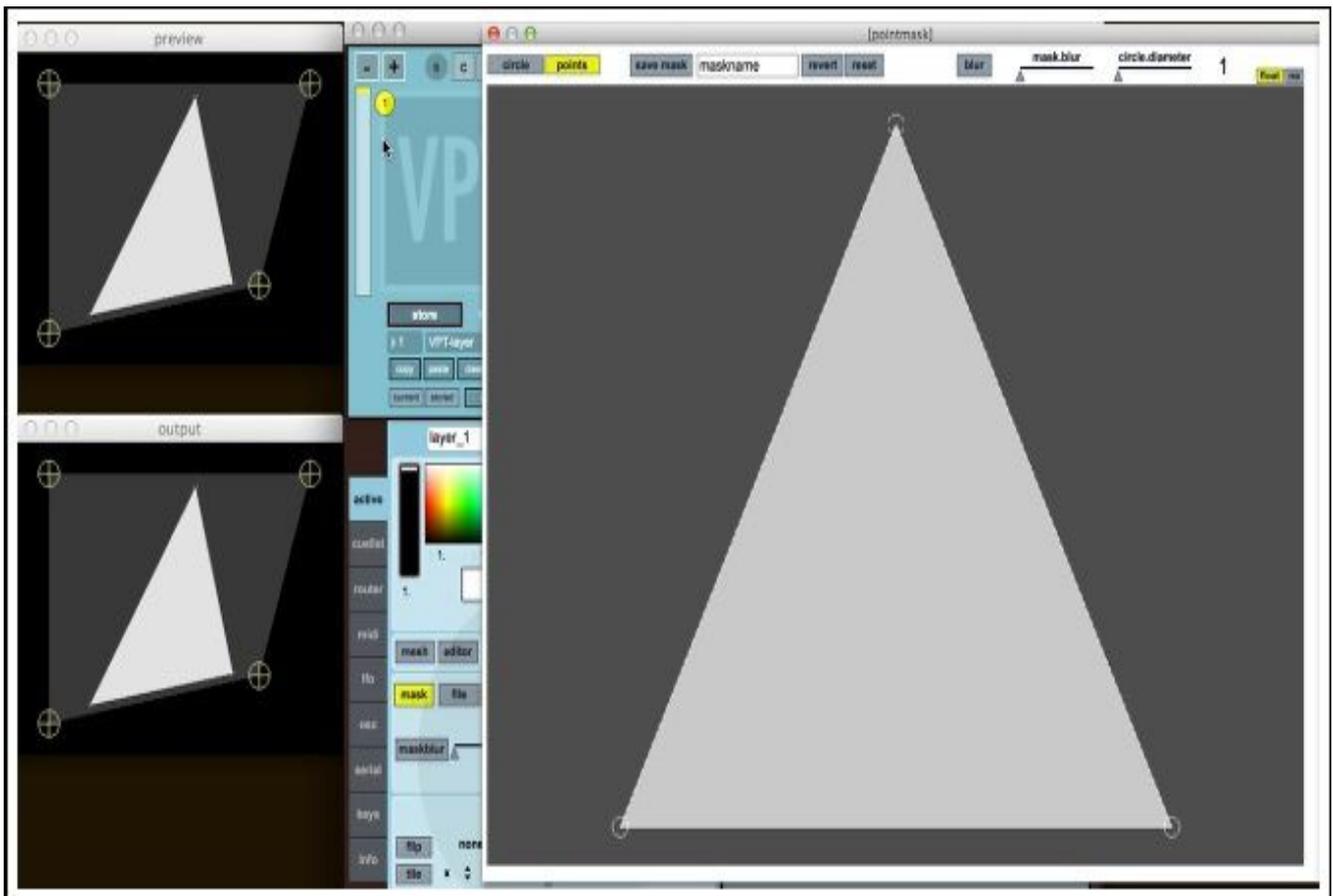


レイヤーのマスクを作成する

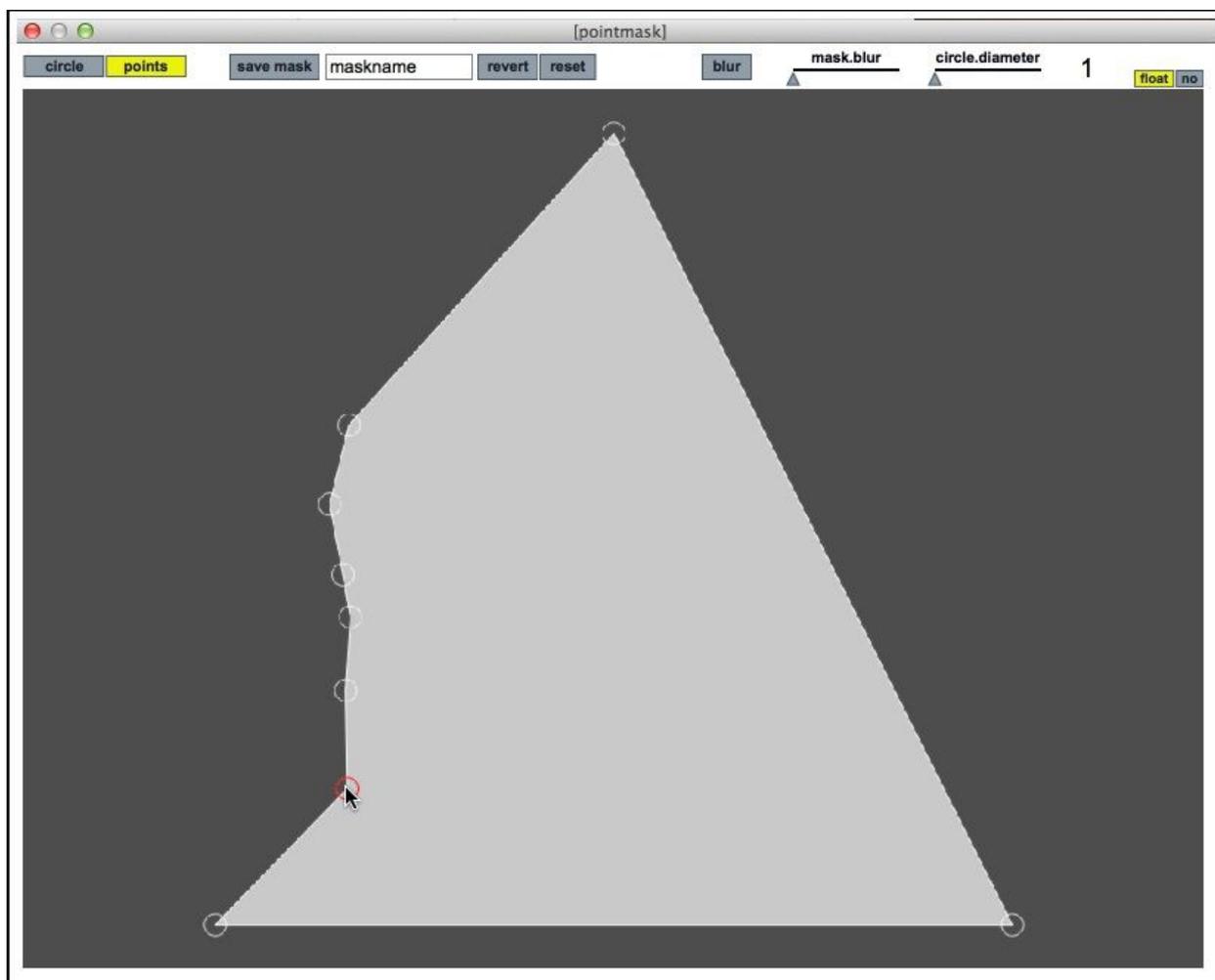
まず、現在のレイヤーがマスクを適用する対象であることを確認してください。また、マスクで簡単にアウトラインを表示するにはマスクのソースとして白色を使用すると便利かもしれません。マスクを作り始める前に、使用するレイヤーはマスクする領域をカバーしていることを確認してください。マスクエディタを開くには、editor ボタンをクリックしてください。



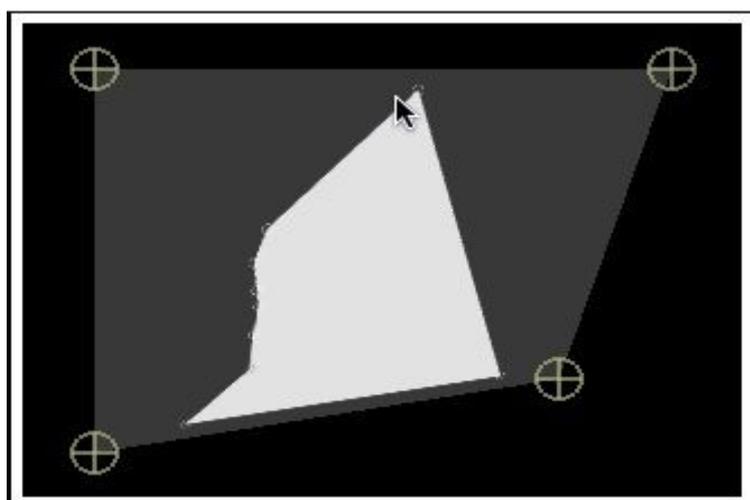
マスクエディタは VPT インターフェースの上面に表示されます。これがあなたの作る最初のマスクの場合は、コーナーハンドルを持つ灰色の三角形が表示されます。これは、リセットボタンをクリックしてに帰ることができ、デフォルトのマスクです。お使いのプレビュー/出力ウィンドウ上では、現在のアクティブレイヤーのマスクが表示されます（アクティブレイヤーはマスクエディタの右上のセクションに数字で示されます）。



既存の点の間をクリックすることにより、マスクアウトラインにより多くのポイントを作成できます。ポイントをクリック/ドラッグして、目的の場所に移動できます。可能な限りマスクを正確にするために必要なだけポイントを追加することができます。ポイントを選択して（赤くなります）、キーボードの Delete/Backspace キーを押すことでポイントを削除できます。



マスクエディタで行う変更はプレビュー/出力ウィンドウで更新されていることがわかります。



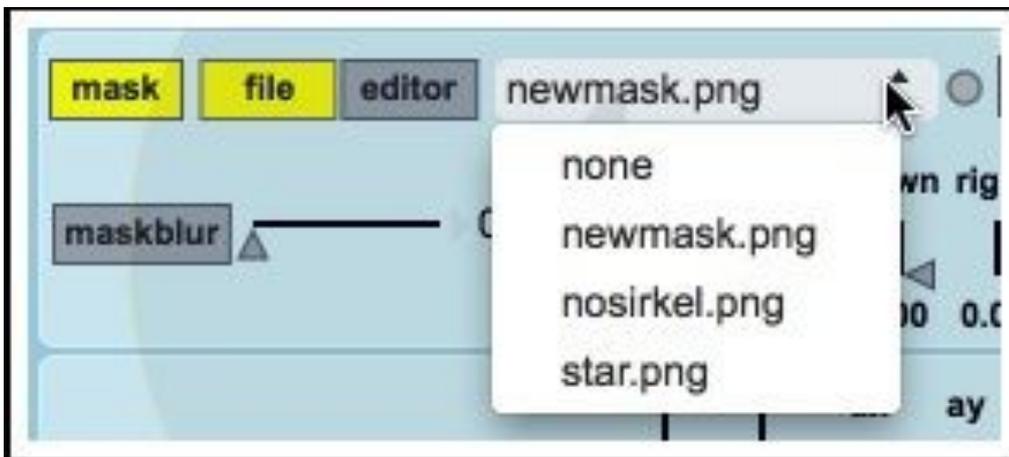
マスク形状に満足している場合には、マスクに名前（この例では newmask）を与えて、マスク保存ボタンをクリックします。マスクエディタウィンドウは閉じます。



アクティブレイヤータブで、新しく作成された newmask というマスクがアクティブであることがわかります

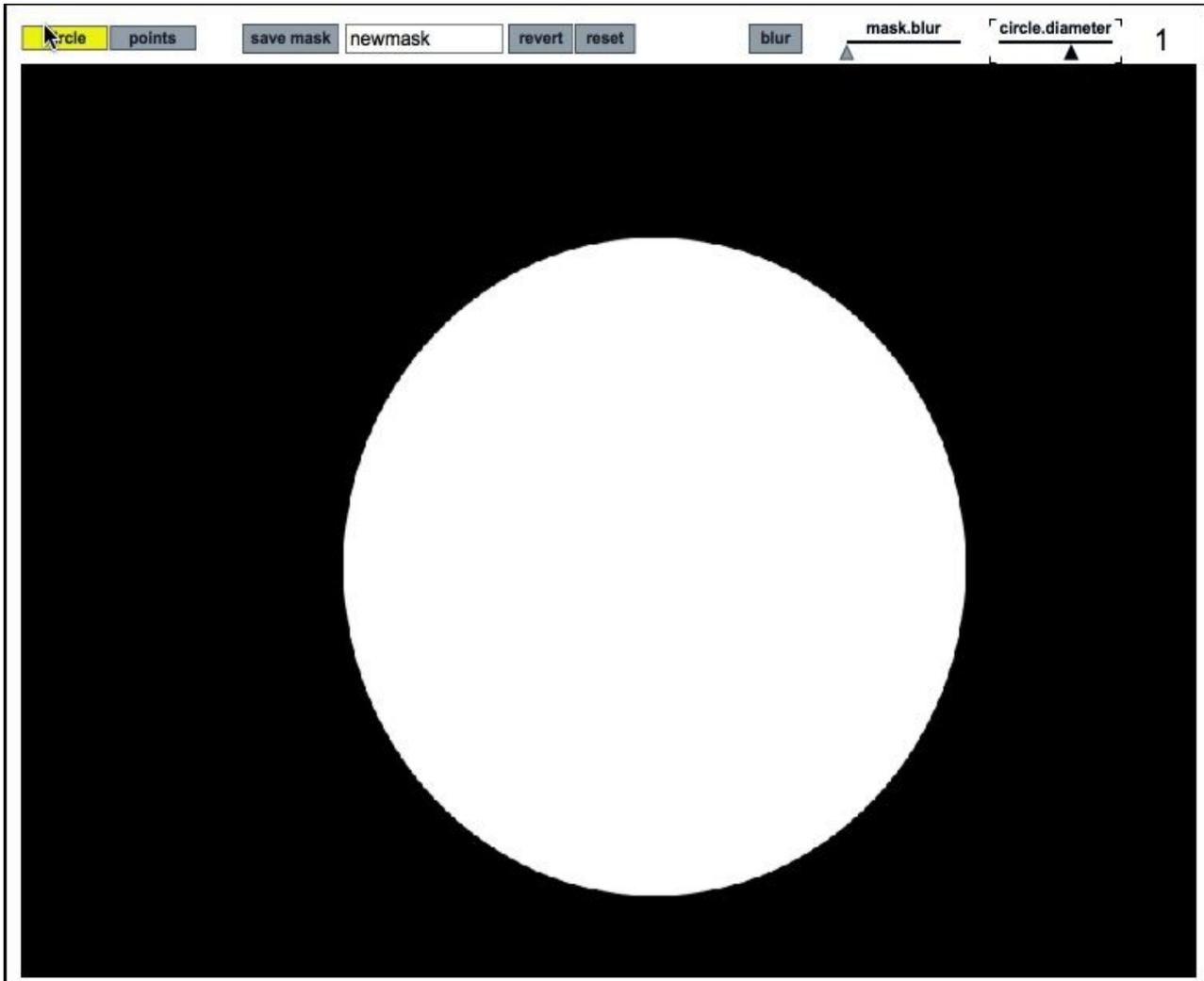


マスクを作成すると、それは黒と白の PNG ファイルとなり、プロジェクトフォルダ内マスクフォルダに保存されます。マスクメニューをクリックすることで利用可能なマスクを参照（選択も）することができます。



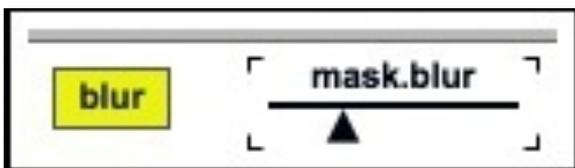
円形マスク

ポイント利用に代えて円形マスクを作りたいケースでは、マスクエディタで円を選択し右上のセクションのスライダーをドラッグして円の直径を変更します。このマスクの保存は、ポイント利用のマスクの場合と同様に行います。



マスクを保存する前にぼかしを追加する

マスク上にぼかしたエッジを持つようにしたい場合、後でぼかしを適用する（これはマスク・ブラー・セクションを使用して可能であるが）よりも、ファイルとして保存する前にぼかしを適用することがはるかに効率的です。単にぼかしボタンをクリックして、必要なぼかしを得るためにスライダーをドラッグします。

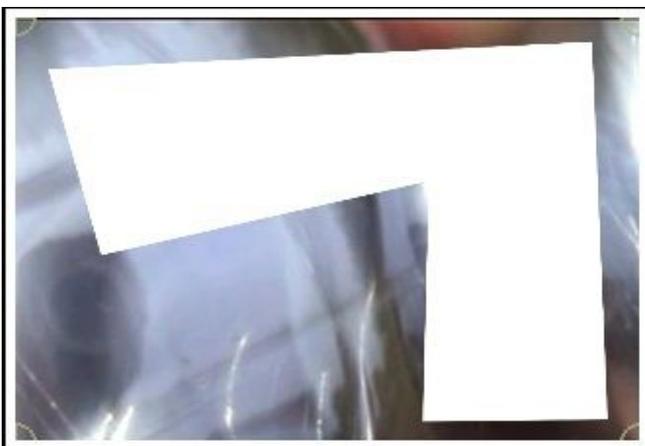
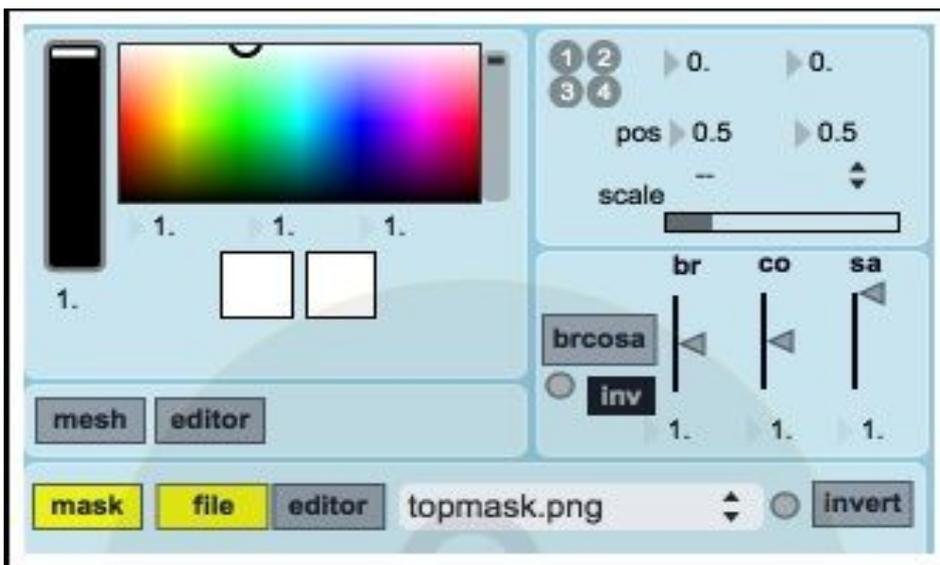


ぼかしはプレビュー/出力ウィンドウでのみ表示され、エディターでは表示されません。

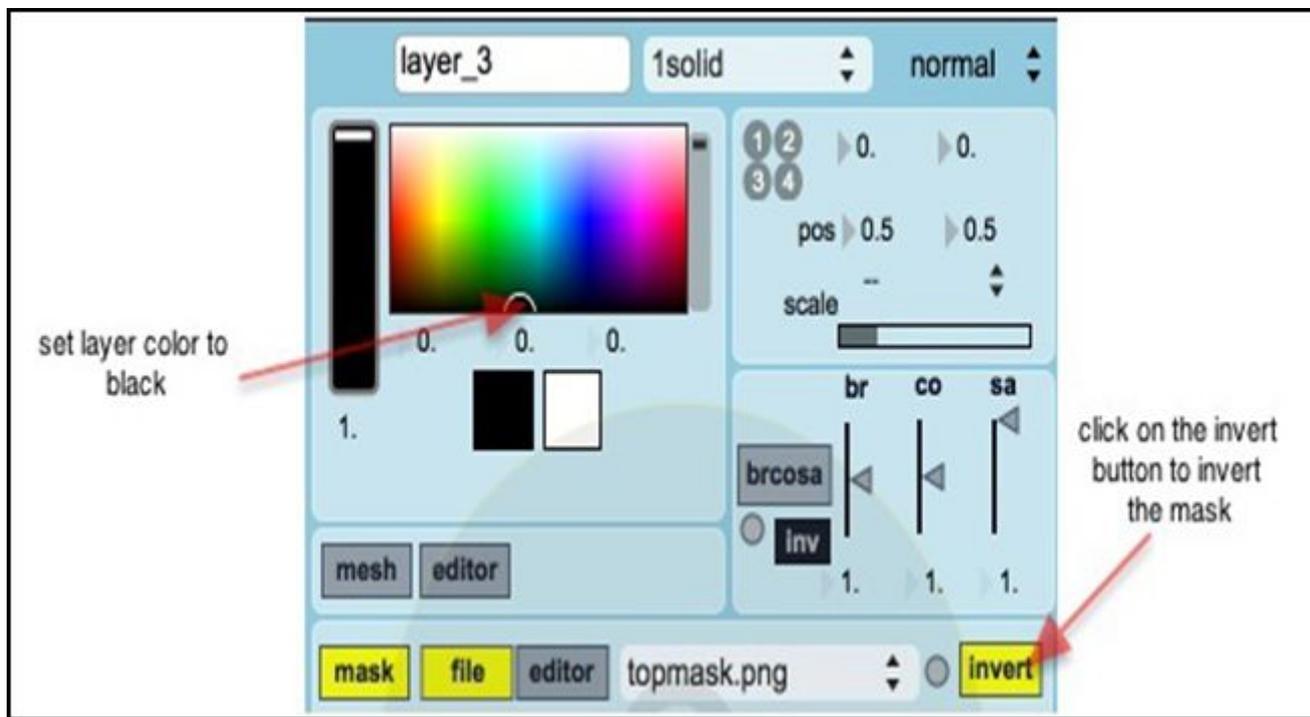


下層のために層をマスクとして使用する方法

まず、レイヤーを作成、それが最上位層であることを確認し、全出力映像をひとまとまりの映像ソースとして使用します。次にマスクエディタを使用してマスクを作成し、通常どおりに保存します。



トップのマスクは、通常のマスクとは反対に機能させる必要があります。マスク形状が透明で、周囲部分を不透明にする必要があります。このためには、2つのことを行う必要があります。まず、マスクを反転し、次にレイヤーの色をブラックへ変更します。

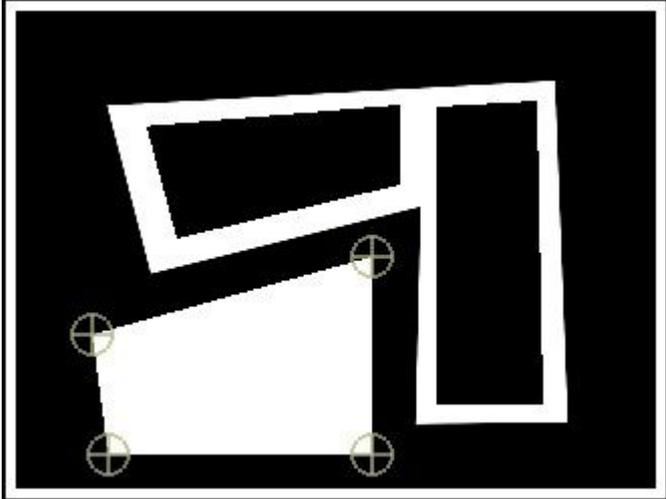


さあ、トップのレイヤーに切り抜きマスクが完成しました

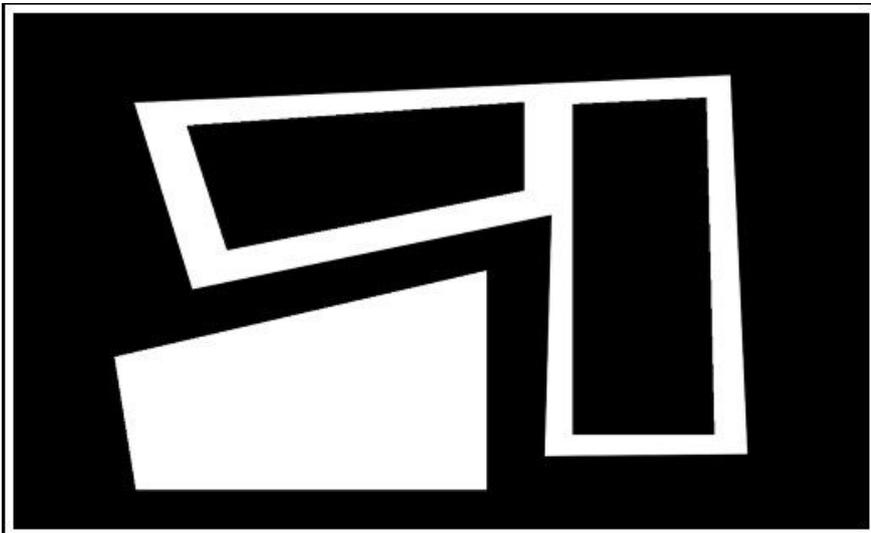


複雑なマスクの作成

時々、マスクエディタで作ることができるよりも、より複雑なマスクが必要となる。これを行う1つの方法は、1個のマスクを作成するために複数のレイヤーを組み合わせることである。そのばあい各レイヤーには、マスクの有り無しを含み黒または白色で固定されている。プリセットとしてすべてのレイヤーを保存しておくことで、後で簡単にマスクを適合させることができます。



別のレイヤーのマスクとしてこれを使用するためには、まず、出力ウィンドウをフルスクリーンにして、スクリーンショットを取る必要があります。



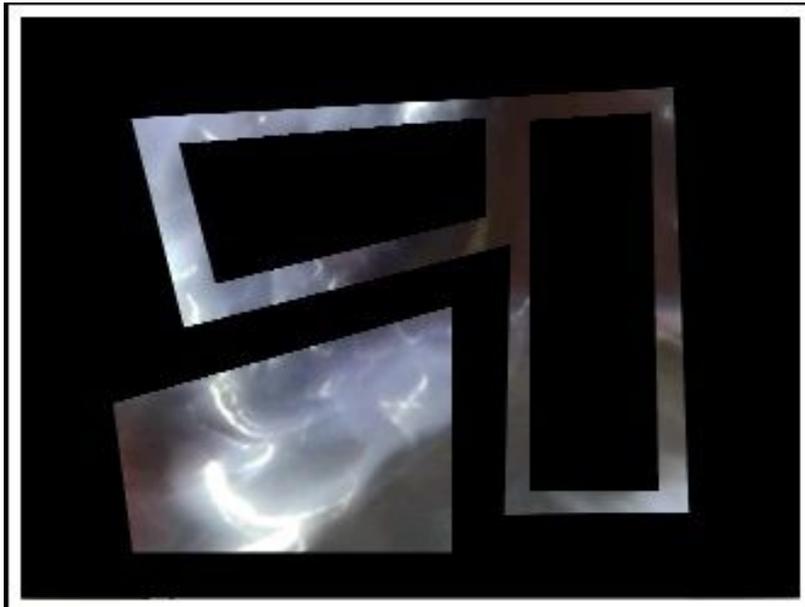
出力ウィンドーの png 画面キャプチャの名前を変更し、マスクフォルダに画像ファイルを保存します。



メニュー内容をリフレッシュするためにマスクメニュー横の円をクリックすると、メニューに新しく作成したマスクが見つかります。



マスクを他のマスクと同じように使用することができますが、マスクが使用しているレイヤーが全出力レイヤーをカバーしている（あるいはうまくマスクに収まらない）ことを確認してください。



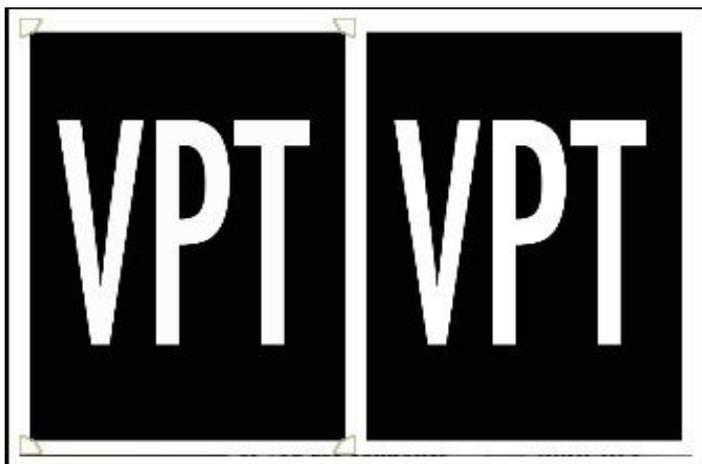
画像編集アプリケーションにおける新しいマスクの調整や作成

GIMP や Photoshop などの画像編集アプリケーションに新しいマスクを作ることができますが、作成したマスクはプロジェクトフォルダのマスクフォルダに入れてください。**マスクファイルは 1024×768 ピクセルの PNG 形式**の画像です。

レイヤー内映像ソースのサイズと位置を調整

レイヤーの位置決めとサイズを調整した後に、レイヤー内映像ソースの大きさ、位置および回転を調整したい場合があります。レイヤー内映像ソースの調整を行っても、元の映像ソースへは影響を与えていません。レイヤーに映像コンテンツが与えられたとき、このコンテンツは**テクスチャ**と呼ばれます。我々はいくつかのレイヤーに同じ映像ソースを使用することができますが、それらはそれぞれのレイヤーでの個別テクスチャとして扱われます。

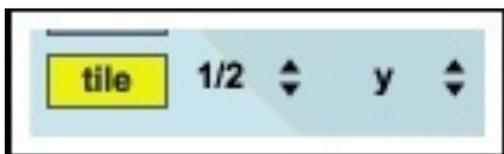
時には、テクスチャの一部だけを使用したいときがあります。異なるレイヤー上に映像ソースの異なる部分を使用したい場合です。以下の例では、2つのレイヤーがあり、最初のものが左半分 of 出力をカバーし、他方のものが右半分 of 出力をカバーしています。



第1レイヤー

第2レイヤー

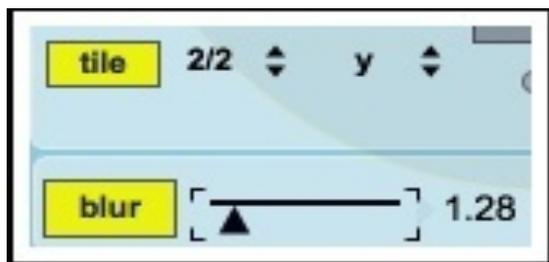
それぞれのレイヤー上でテクスチャの半分のみを使用したいのであれば、`active` タブの `tile` オプションを使用することができます。左レイヤーで水平方向でテクスチャの前半を使用します。`tile` は半分、3分の1、四分の1、五分の1に、`x` および `y` 方向の両方に分割できます。たとえば、`x` 方向の `2/3` という表記はテクスチャの **ミッドサード** <約注: **3分割の真ん中**>のみを使用するという意味です。



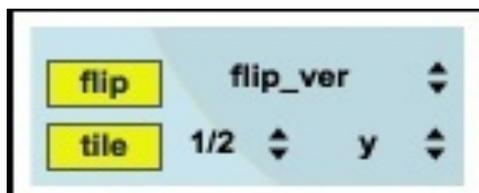
ここでは、第1レイヤーのテクスチャが変更されています



その後第2レイヤーに同じことを行いますが、今度はテクスチャの後半を使用したいとおもいます。また、2つのレイヤーの違いを示すために第2レイヤーにぼかしを追加しました。

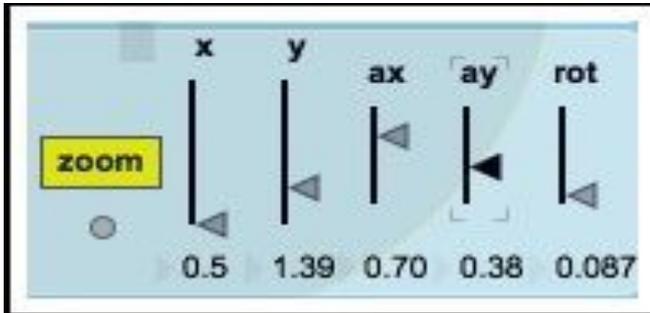


flip セクションでは、テクスチャを水平、垂直または両方に反転できます。



テクスチャのズーム、パンおよび回転

tile はいくつかの状況で正常に動作しますが、時にはより多くのコントロールが必要となります。ズームコントロールを使用すると、レイヤー境界の内側で、テクスチャへのズーム、移動、回転のコントロールが必要です。

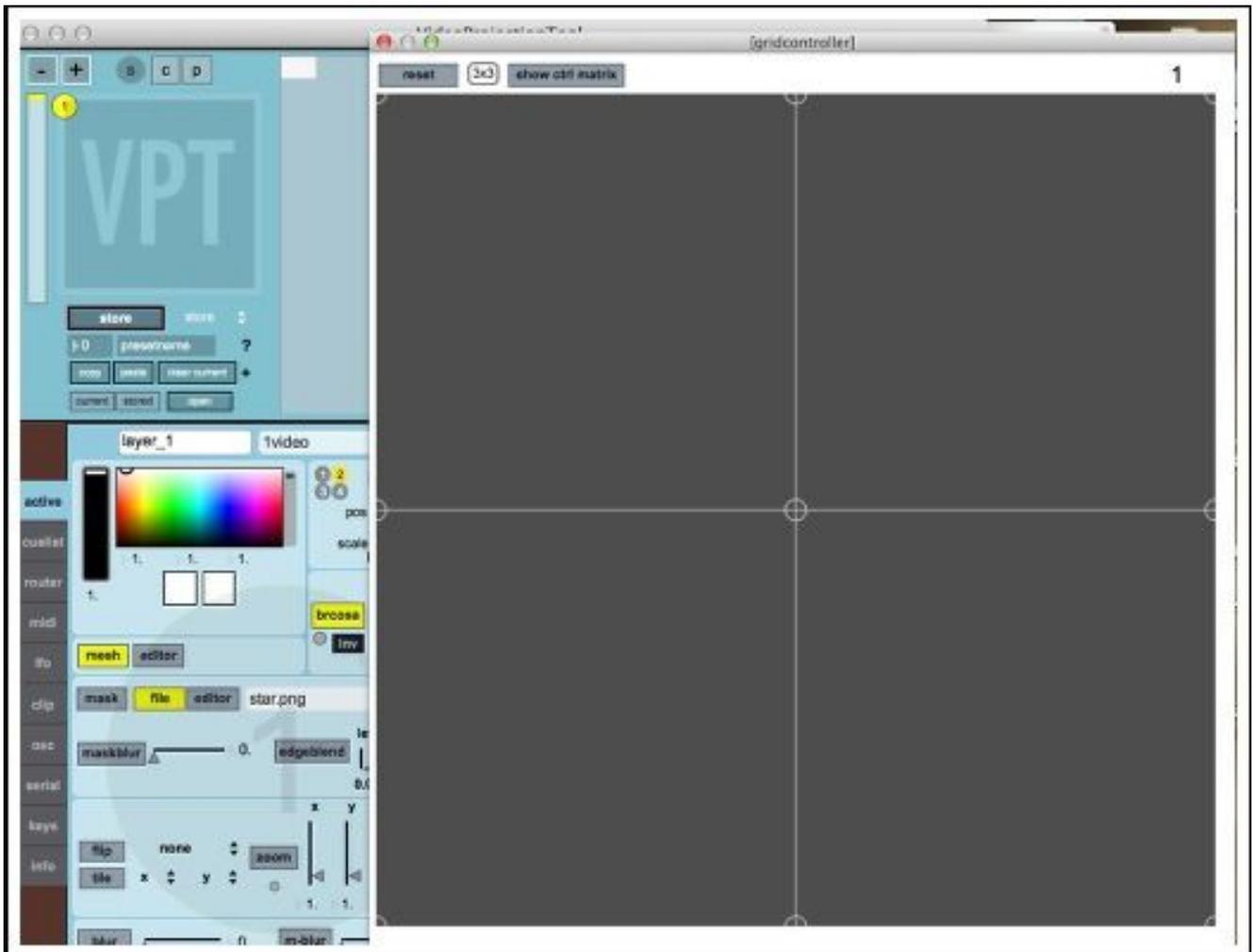


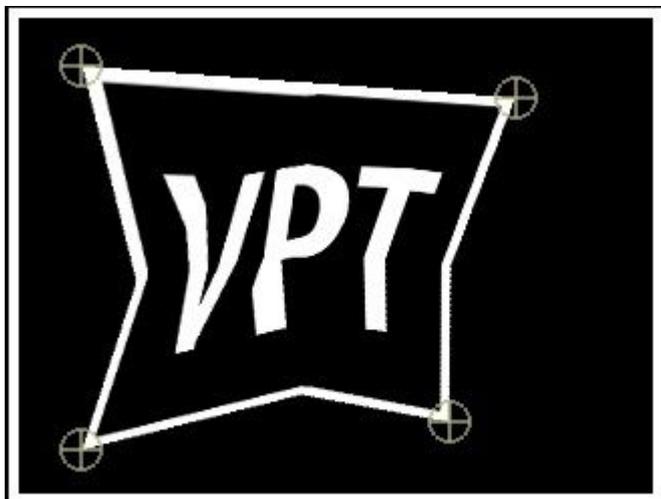
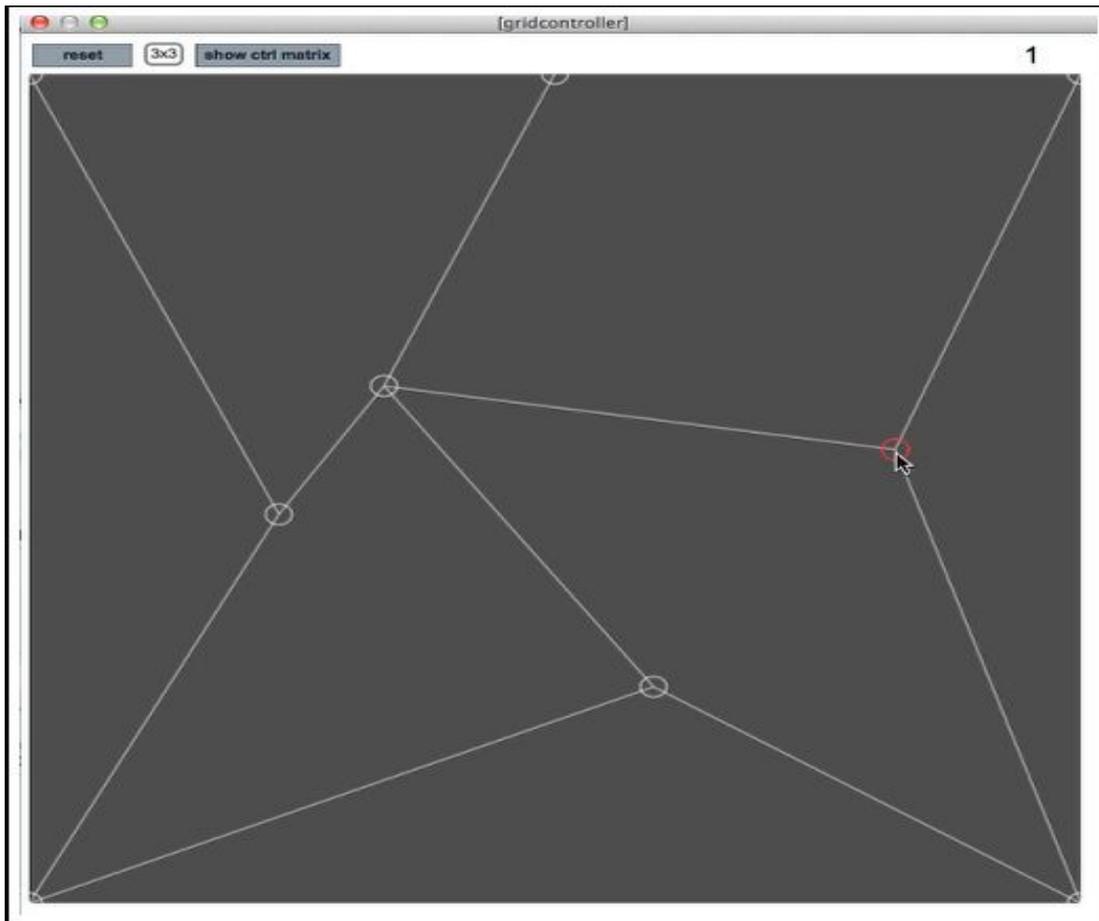
マスク内側にテクスチャーを合わせたい場合、非常に便利です。



メッシュエディタ

VPT7は、メッシュエディタを持っており、activeタブで利用可能です。メッシュエディタを開くために、mesh ボタンをクリックすると、マスクエディターとあまり変わらないインターフェースが表示されます。最初にメッシュをリセットしたり、グリッドのサイズを選択することが重要です。この例は3×3に設定されているので、合計9個のコントロールポイントがあります。グリッドを出力ウィンドウで表示したい場合、"show ctrl matrix"をクリックしてください。単にコントロールポイントをドラッグすることで、画像を歪められます。

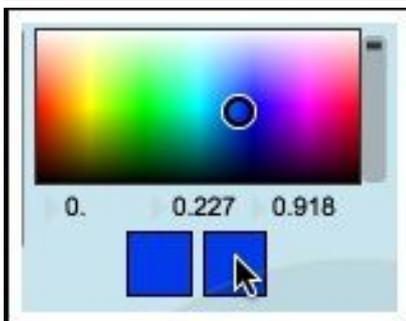
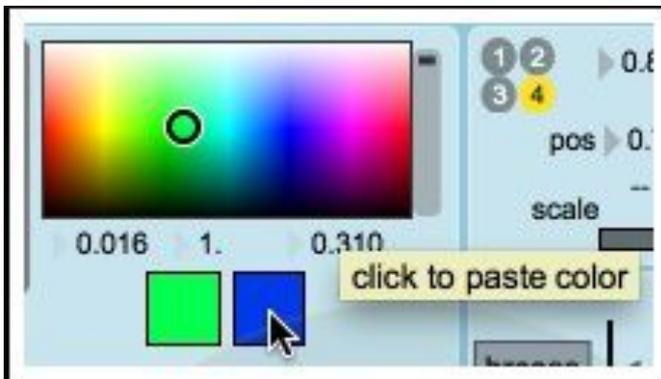
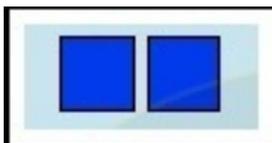
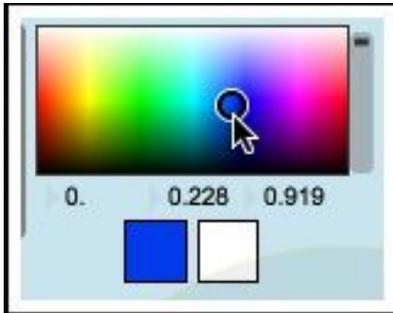




メッシュグリッドを保存するには、メッシュエディタを閉じます。プリセットが格納される際、メッシュ設定がレイヤーと一緒に保存されます。

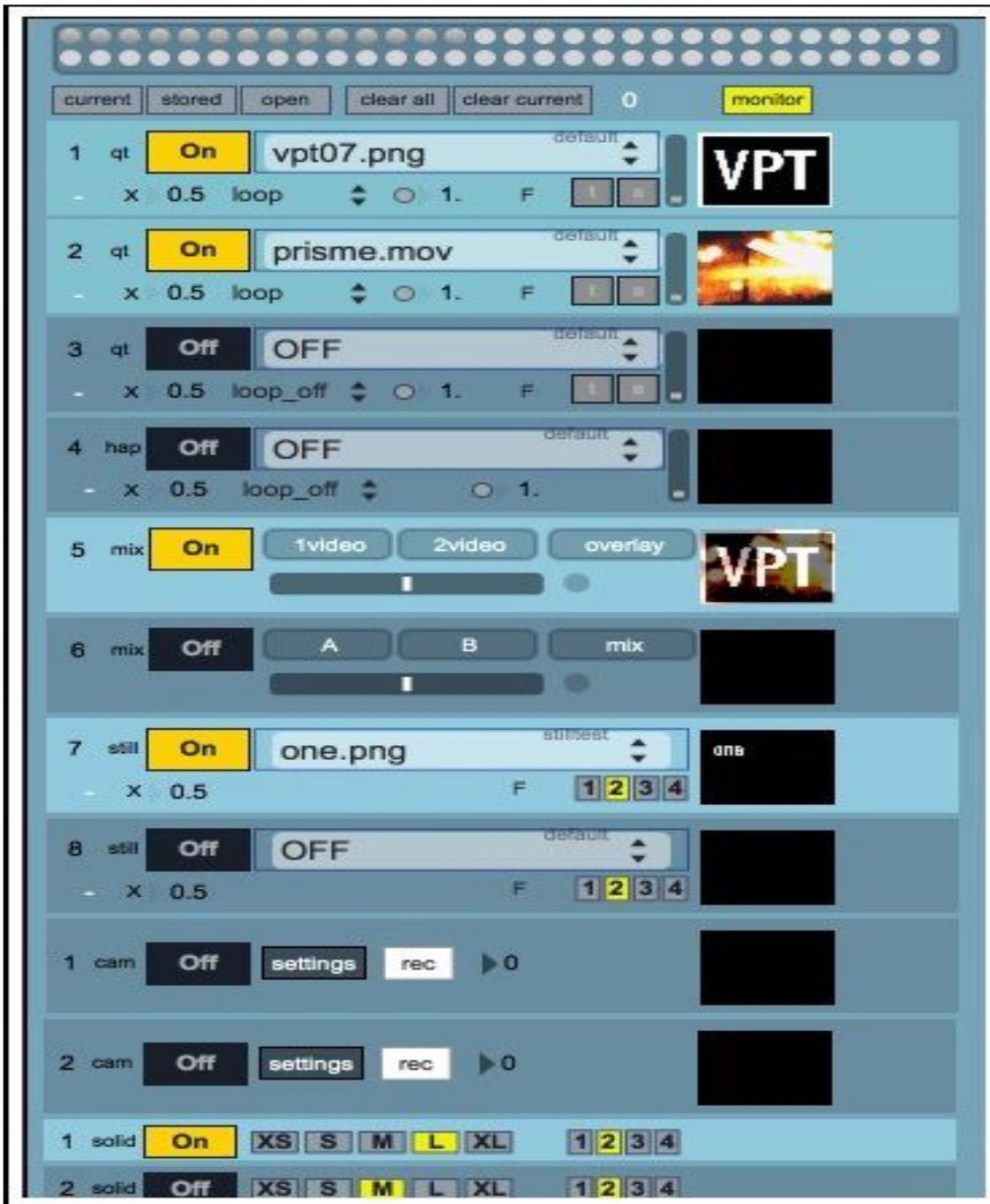
カラー値のレイヤー間コピー&ペースト

異なるレイヤーで同じ色を使用したい場合があります。active タブで色をコピーするのは簡単です。



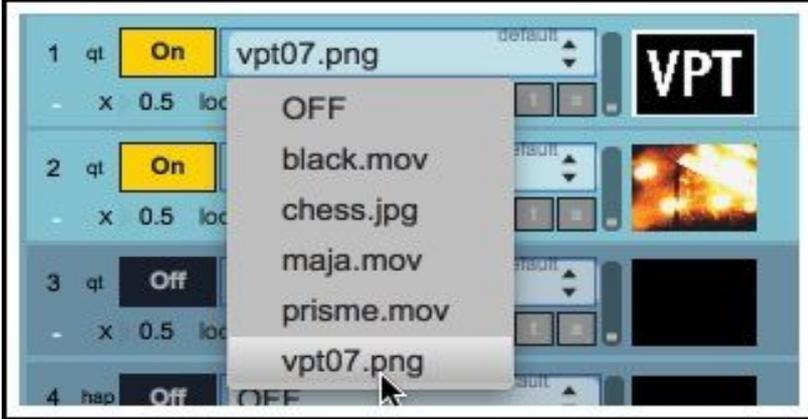
映像ソース

VPTには、幅広い映像ソースの選択肢があり、最初の8つの1-8videoソースはプロジェクトの必要性に合わせてカスタマイズすることができるので、たとえば8つのqt(QuickTimeビデオ)ソースや8つのstill(Still image 静止画)ソースを持つたりできます。1-8videoソースで利用可能な映像ソースの種類は、QuickTimeモジュール、HAPモジュール(現在はMacのみ)、still imageモジュールとMIXモジュールである。QuickTimeのモジュールは最も使用されるでしょう、HAPモジュールはHAPコーデックを使用して動画エンコードに最適化されます、静止画像のみを扱う場合は、still image 静止画モジュールが最良の選択です。MIXモジュールはさまざまなブレンドモードを使用して2つの映像ソースをミックスします。



QuickTime のモジュール

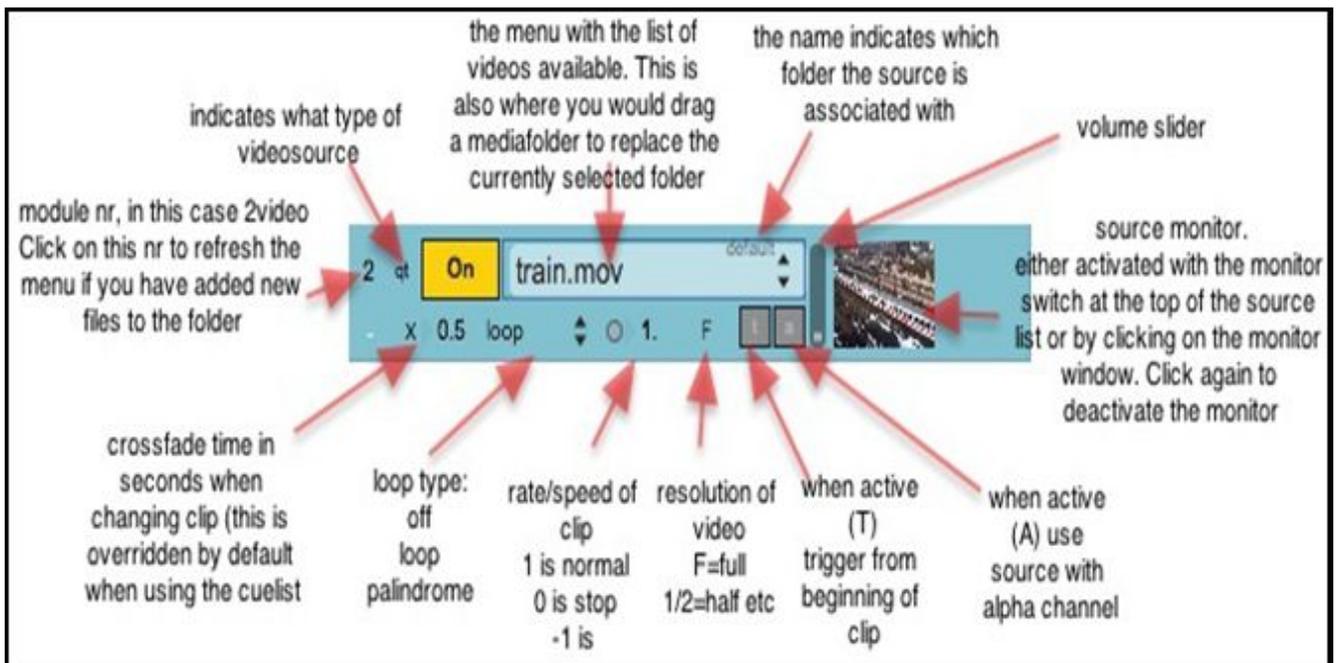
QuickTime のモジュールは、ビデオのフォルダ（ビデオと静止画が混在）を参照します。デフォルト設定では、プロジェクトフォルダ内の video フォルダです。映像ソースを使用するには、On/Off ボタンをクリックして On にし、メニューから映像ソースファイル（ビデオ・フォルダのファイル）を選択する必要があります。



VPT は QuickTime を基にしており、好ましいコーデックは、75%品質の JPEG やアップルの ProRes コーデックなどです。インターネットからダウンロードした MP4/H264 ビデオは、使用する前に変換する必要があります（付録B VPT trimmer のアドオンをチェックしてください）

お使いのコンピュータに依りますが、同時にいくつかの大規模なビデオファイル再生は、全てが遅くなるので、適切なサイズにビデオをレンダリングするとか、（たとえば、投影出力の四分の一のみをカバーするレイヤーだけの場合、フルHD 解像度を使用する意味がありません）QuickTime モジュールの解像度メニューを使用して VPT をスムーズに再生できる解像度に下げます。

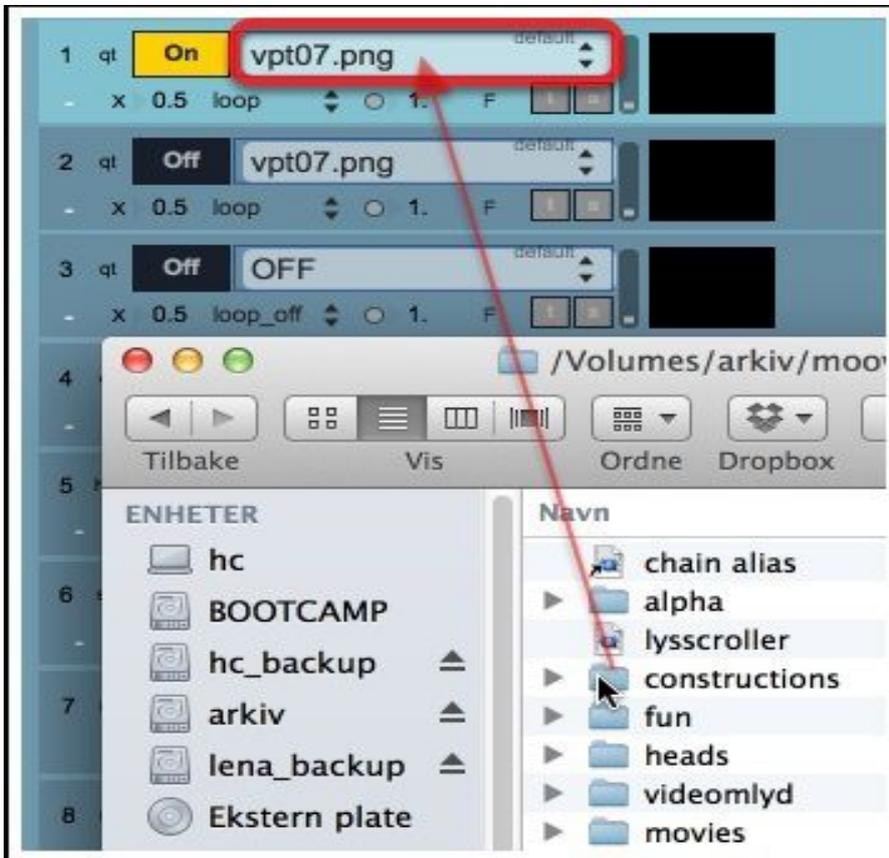
映像ソース モジュール番号	映像ソース タイプ	ビデオファイル名	ビデオフォルダー名	ボリューム
------------------	--------------	----------	-----------	-------



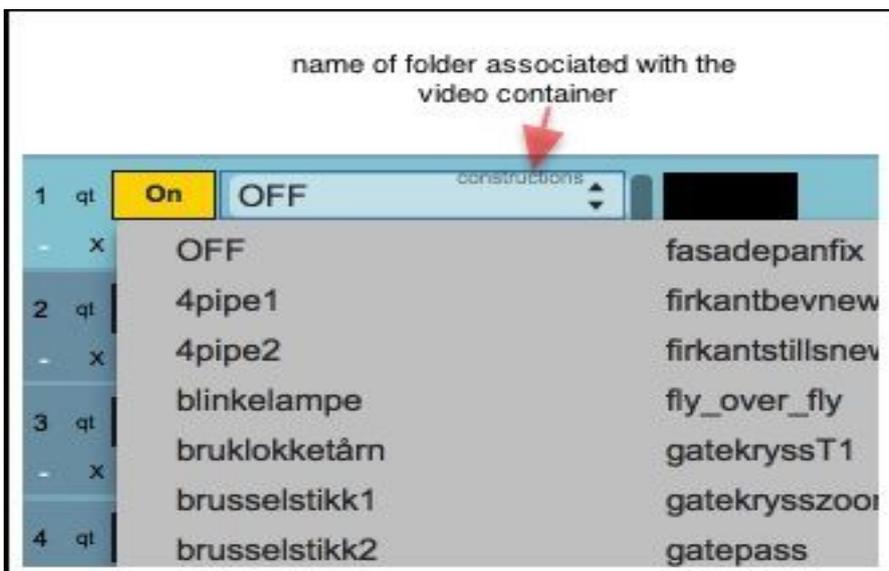
クロスフェード時間	ループタイプ	クリップ速度	解像度	トリガー	アルファチャネル (透明)
-----------	--------	--------	-----	------	---------------

独自の映像ソース(ファイル、フォルダー) の追加

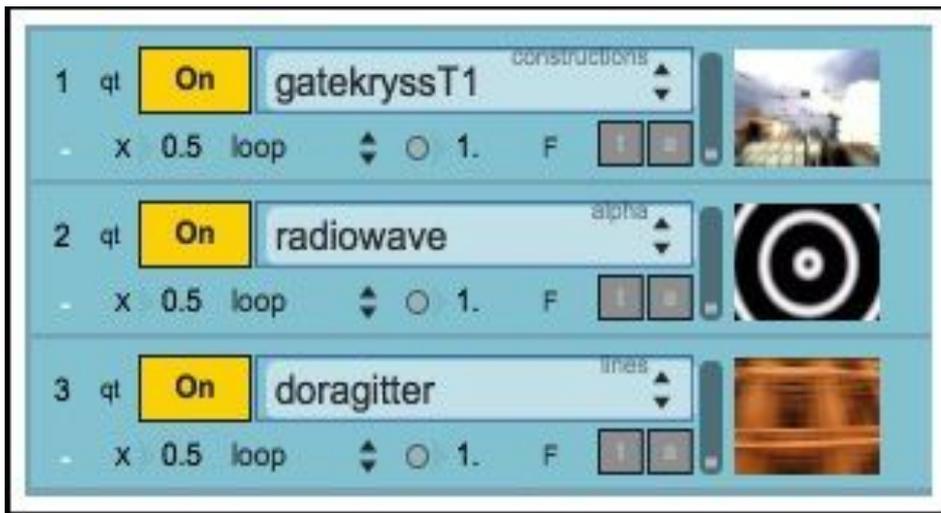
VPTに独自の映像ファイルを追加することは簡単です。一つの方法は、単にプロジェクトフォルダ内のビデオフォルダに独自の映像ファイルをコピーすることです。また、独自の映像ソースのフォルダー（個々のファイルではない）を異なるビデオコンテナへドラッグして行うことができます。



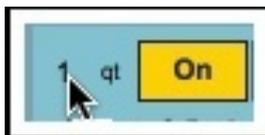
このページの例の場合、最初のビデオコンテナは、1videoにはフォルダ"constructions"がドラッグされ、その映像ソースファイルが更新されます。ビデオコンテナにドラッグで関連付けられたフォルダ名は、メニューの右上隅に見ることができます。



別のビデオコンテナが、別のビデオフォルダを持つことができます。



注意！VPT が開かれている間にフォルダにファイルを追加した場合は、ソース番号をクリックしてそのフォルダに関連付けられたビデオコンテナを更新する必要があります。

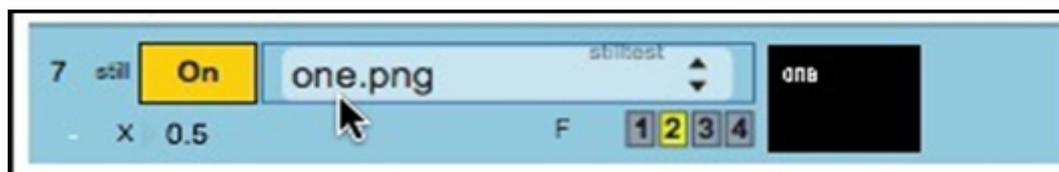


静止画像 (still image)

静止画像モジュールは、QuickTime のモジュールと同様に機能します。QuickTime モジュールで説明されているのと、同様の方法で解像度を調整できます。また、リフレッシュレート (数値 1-4) を調整することができます。毎秒多分 30 回更新される動画とは違い、静止画像は自身では更新されませんので、VPT がリフレッシュレート (数値 1-4) に基づいて静止画像を更新します。

- F 1 毎秒 30 回の更新 (静止画ではめったに必要ない)
- F 2 毎秒 10 回秒の更新
- F 3 毎秒 2 回の更新
- F 4 2 秒毎に更新

リフレッシュレートの数値決定は、どのように静止画像を使用するかに依存するので、試してみる必要があります。ただ、高解像度画像を頻繁に更新する場合は、フレームレートの大きな犠牲に気づくでしょう。



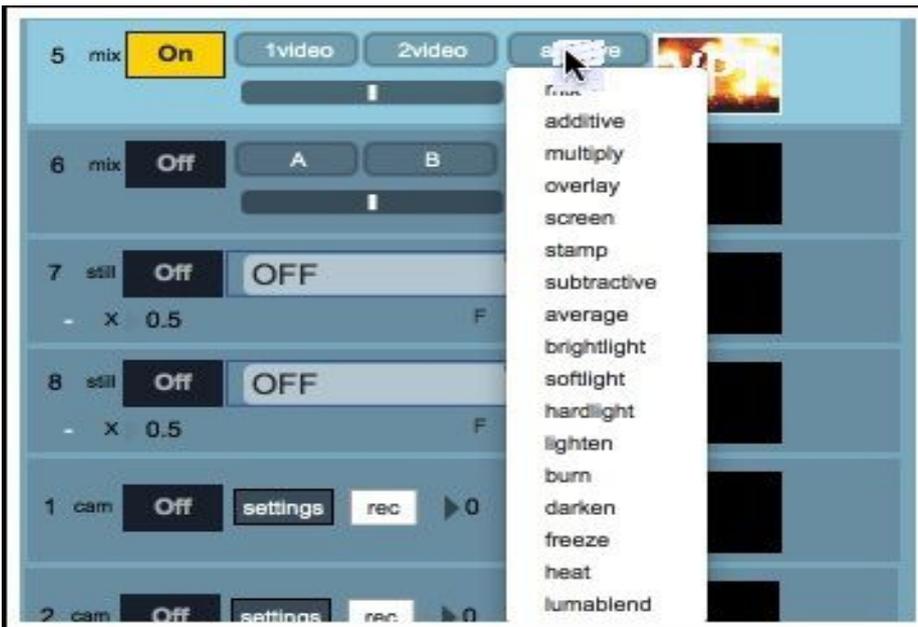
解像度 リフレッシュレート

MIX モジュール

MIX モジュールは、異なるブレンドモードを使用して、任意の2つの映像ソースをミックスすることができます。AとBの各メニューで各々の映像ソースを選択します。

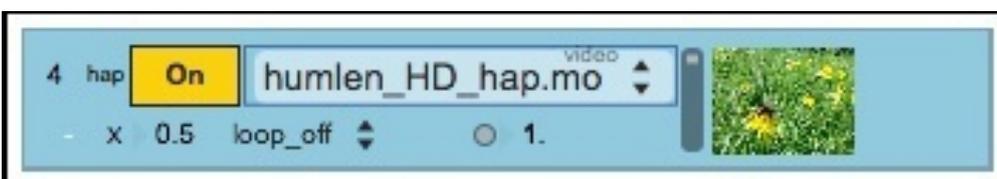


MIX メニューに幾つかのブレンドモードがあります。そしてミックススライダを使用して映像ソース A と B を混在させることができます。



HAP モジュール (Mac のみ)

HAP モジュールは、通常の QuickTime モジュールと似ていますが、高解像度ビデオに適している。ただ、VPTを実行しているコンピュータにインストールしたHAPコーデックで高解像度ビデオ圧縮する必要があります。

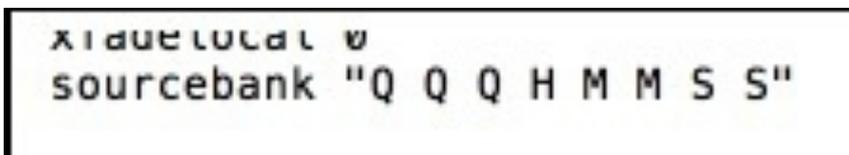


ビデオモジュールの独自の選択

プロジェクトフォルダ内の prefs.txt というテキストファイルに含まれている VPT の環境設定には、映像ソースモジュール (1video-8video) の設定情報があります。sourcebank という語を含む行の後に、引用符の中にアルファベット大文字が書かれています。テキストエディタで、必要性に応じこの行を変更することができます。これらの文字はそれぞれ、モジュールタイプを示しています。

- Q QuickTime ビデオ
- S 静止画像
- H HAP
- M ミックス

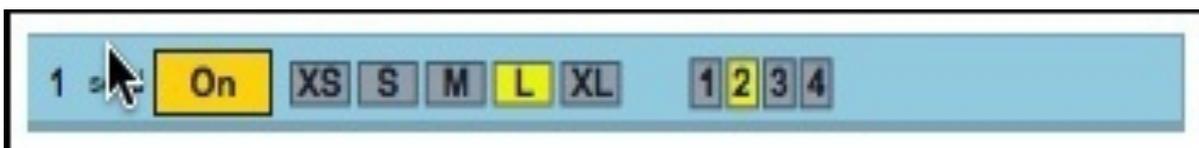
たとえば、4つの QuickTime ソースと 4つの静止画像ソースにしたいのであれば、次のような行に変更します。 sourcebank"Q Q Q Q S S S S"



solid モジュール

solid モジュールは白色の solid モジュール (レイヤーでソースとして使用される場合、色付けできる) が作成できます。また、solid モジュールは主に背景として又はマスクとの組み合わせで使用されます。(マスクを作成する際に非常に有用である) 固定画像の still ソースのように、solid モジュールは異なるリフレッシュレートを持てます。そして、固定の解像度を選択するのに、XS-S-M-L-XL の記号を使います。
XS 4×4 ピクセル S 80×60 M 320×240 L 640×480 XL 1024×768

マスクなしでレイヤーを色で埋めるために、solid モジュールのみを使用する場合には、XS の解像度を使用することができます。マスクと一緒に solid モジュールを使用する場合は、solid モジュールの高い解像度によりマスクがスムーズに見えます。(違いを見るために試してみてください)



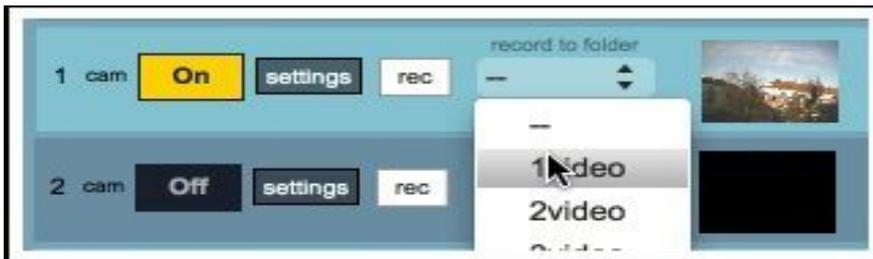
ライブ入力 (モジュールタイプ cam)

VPT は2つのライブ入力を備えており、コンピュータに接続されたビデオカメラや内蔵されたカメラを操作できます。他の映像ソースのように、On/Off ボタンをクリックすることでライブソースをアクティブにします。通常、VPT は、カメラが接続され機能している場合は、自動検出してモニタウィンドウに入力を表示します (モニターがアクティブな場合)。自動検出しない場合は、settings ボタンをクリックして、カメラを探してみてください。VPT を開始する前にコンピュータに入力ソースを接続すべきでしょう、そうしないと VPT を再起動する必要があります。



ライブ入力を記録

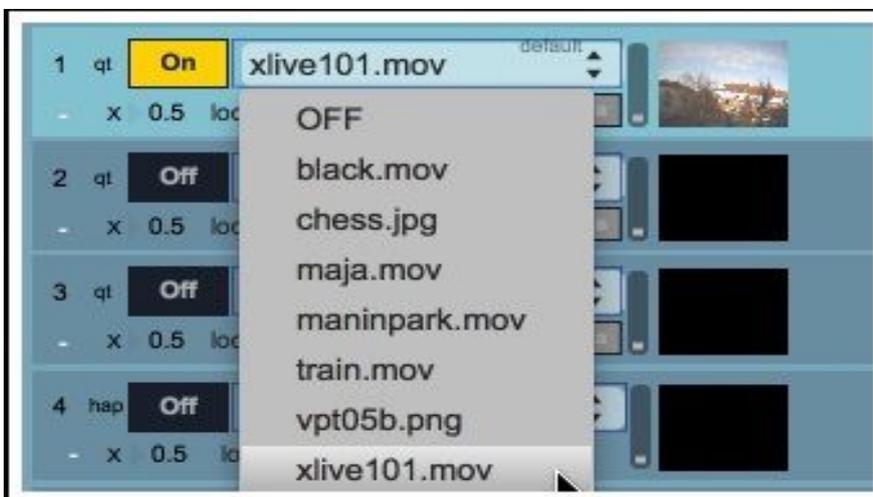
ビデオフォルダにライブソースからの入力を直接記録できます。 "record to folder"メニューで 1video を選択した場合、ライブ入力のビデオファイルは 1video ソースに対応するビデオフォルダに記録されます。



録画を開始するには、"rec"ボタンをクリックしてください。録画中"rec"ボタンは赤くなります。録画を停止するには、もう一度"rec"ボタンをクリックしてください。



録画ファイルには xlive101, xlive102 などの名前が自動で付きます。



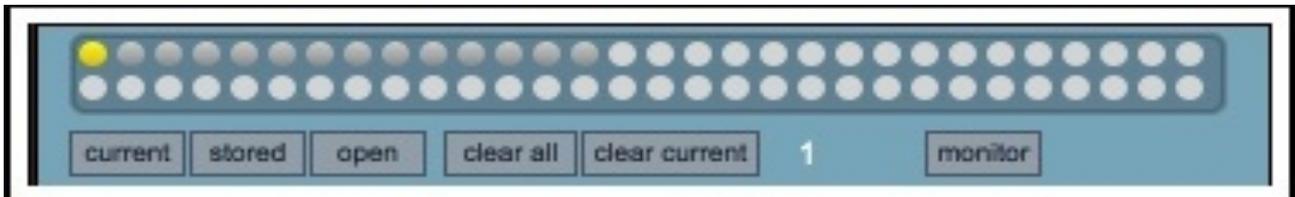
source プリセット

ソースセクションの上部に、source プリセットコントロールがあります。VPT のデフォルト動作では、プリセット保存時に source プリセットが自動的に作成されるので、このことは言及しませんでした。しかし、VPT プリセットと source プリセットを独立して維持したほうが役に立つ使い方もあります。

VPT の通常利用では、特定のセットアップのためにレイヤを作成しマップするので、レイヤーを周りに移動したり、そのプロパティを変更する必要は全くありません。通常、作成者はレイヤーのセットアップ作成やコンテンツの変更に最も興味があるでしょう。したがって、作成者は1つの VPT プリセットのみで作業を行い、その後、残りの作業を source プリセットで行います。

source プリセットはすべてのソースについての情報を保存します。どのビデオが各ソースで選択された状態か、再生速度、ループのタイプ、クロスフェード時間などの情報です。

ソースバンク上部にある円の1つを **shift-click** して source プリセットを作成します。円を **click** することで source プリセットを選択します。黄色い円は現在のプリセット、濃い灰色の円はプリセットが保存されていること、ライトグレーの円は空を示しています。



注意してください。VPT のデフォルトでは、VPT プリセットを保存するたびに毎回 source プリセットを作成しています。

通常の VPT プリセットとは別の source プリセットで作業するには、プロジェクトフォルダー内の `prefs.txt` でいくつかの設定を変更する必要があります。(ドキュメントの設定セクションで、設定の詳細を参照してください) デフォルトの設定は次のとおりです。

```
include_sources 1
autosource 1
xfadelocal 0
```

この設定の指示内容は、VPT プリセットを保存するときに source プリセットを含める、VPT プリセットを保存するときには自動的に source プリセット (1001 から始まる) を作成して格納すること、それと、`cuelist` のクロスフェード設定で、個々のソース内のクロスフェードの設定を上書きすること。

以下に示すように、設定を変更したいと思うでしょう。source プリセットの自動保存を無効にし、ローカルで `crossfade` 値を設定可能にする。

```
include_sources 0
autosource 0
xfadelocal 1
```

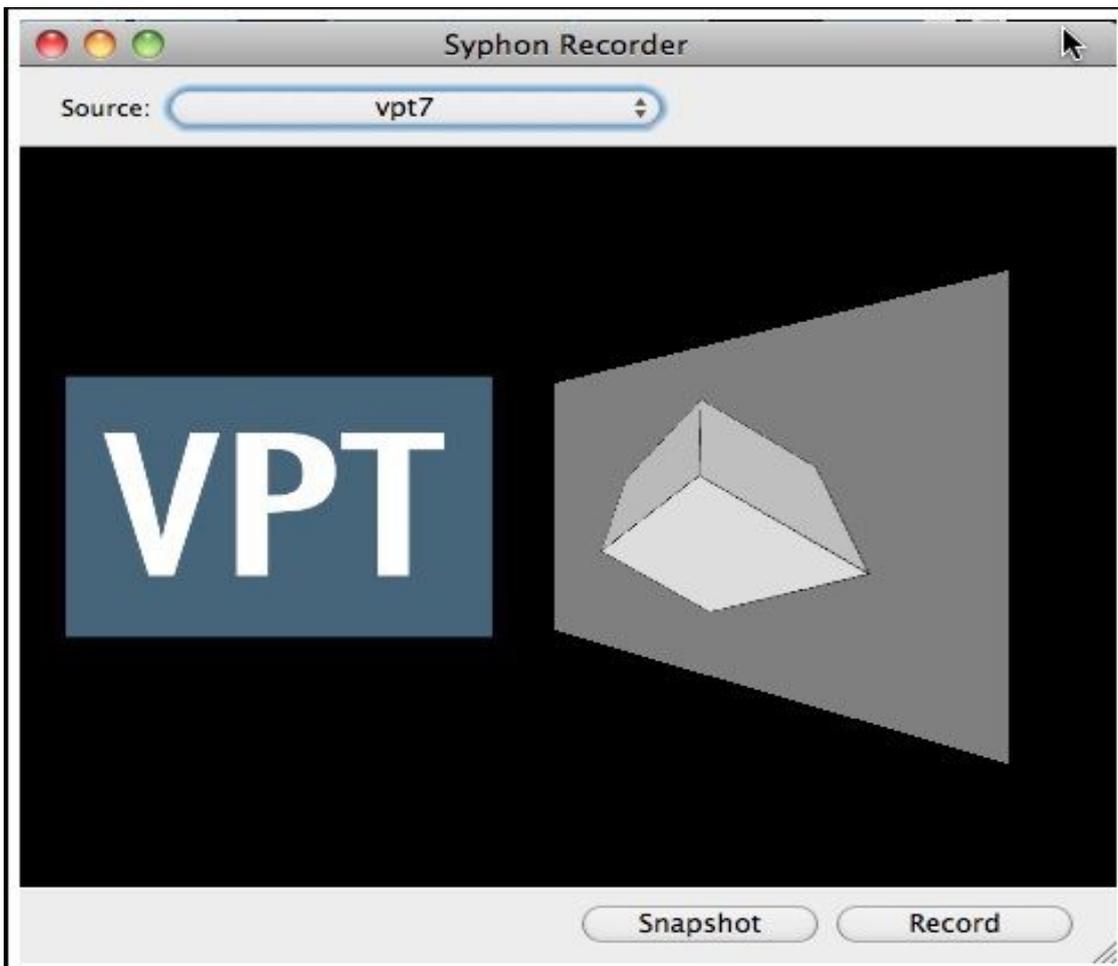
syphon (サイフォンと発音、Mac のみ)

VPT は、アプリケーション共有のビデオストリームである syphon フレームワークをサポートしています。これにより、VPT のソースとして、Processing、OpenFrameworks、Max-Jitter、VDMX、Modul8++のような他のアプリケーションの出力を使用できます。



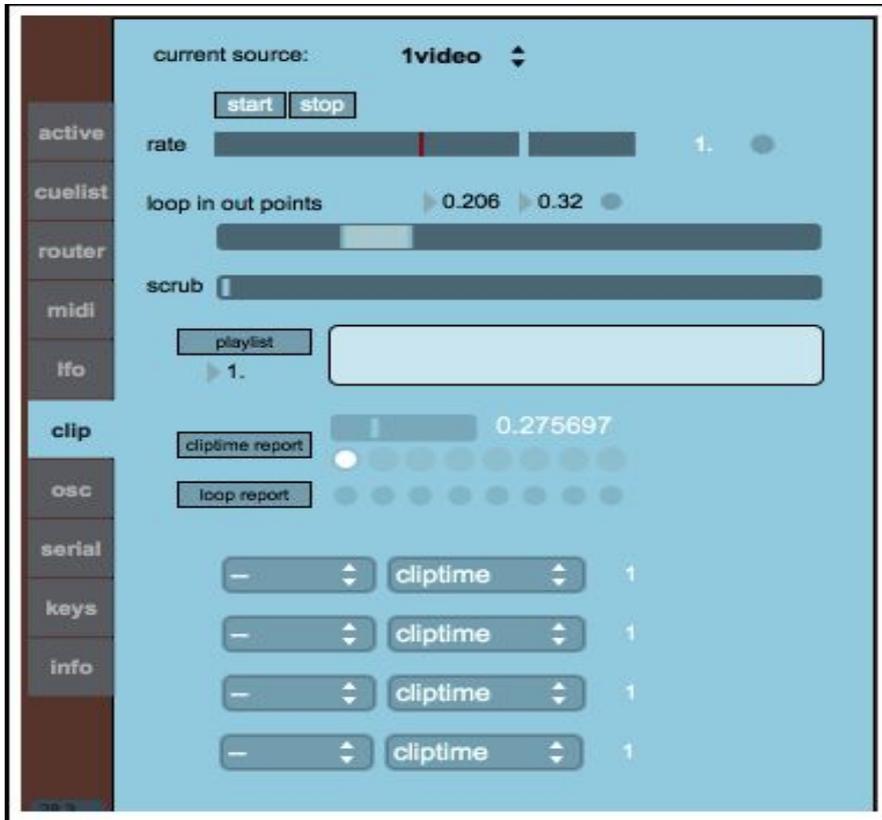
syphon 出力

VPT は出力を syphon 出力で送ることもできるので、他のアプリケーションが VPT の出力を使用できる。syphon 出力ボタンはサイフォンモジュールの下、ソースリストの一番下にあります。サイフォン出力は、サイフォンレコーダーとの組み合わせで特に有用です。サイフォンレコーダーは受信したサイフォンビデオストリームを記録できるので、VPT からの出力を記録するための方法として特に有用です。



clipコントロールタブ

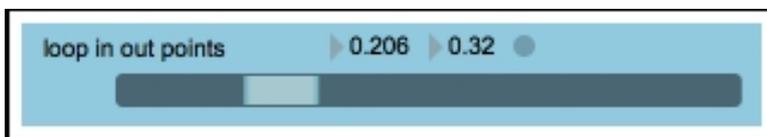
このタブにあるコマンド群は、routerまたはoscコントロールタブでも利用できますが、clipコントロールタブでは、直接的にビデオソースのより細かな制御が行えます。(主にQuickTimeビデオソース)
まず、コントロールしたいビデオソース (1video-8video) を選択してください。



ループポイント

スライダーエリアでクリックやドラッグでループポイントを設定します。cmd+click-drag (Mac) または Ctrl+click-drag (Win) で選択したループを移動 (ループ長は同じまま) し、また、選択中のループ両端いずれかでのshift+clickにより、端は延伸する。

ループポイントは、0. と1. の間の値に関連しており、プリセットの保存時にこれらの値が保存されている。

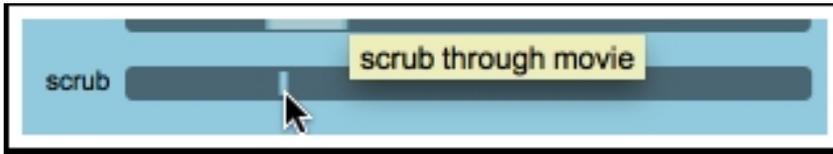


Scrubbing (再生位置飛ばし)

ビデオを scrub する (再生位置を飛ばす) には、ビデオを停止するか、またはレートを0に設定する必要があります。



そして、スライダーをクリックしてドラッグするだけで、フレームを通してビデオの scrub (再生位置飛ばし) ができます。

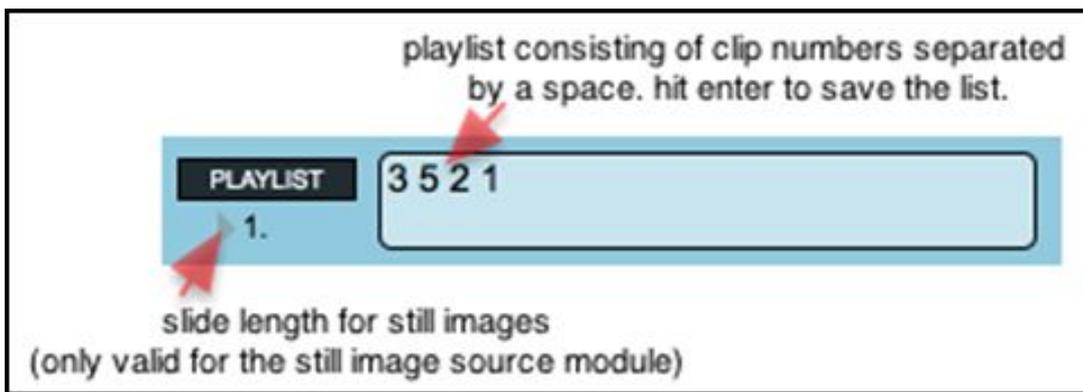


Playlist

プレイリストとは一つの選択肢です。それは、クリップを最後まで再生する各ビデオソースに対応した個別のプレイリストを持って (含んで) いる (または、指定されたスライド長の固定画ソースに対応している) したがって、次項目へのクロスフェードがリスト内にあります。

数字は、指定したソース内のクリップ番号を表します (スペースで区切られる)、

そして、プレイリストのボタンがオンになっている限り、以下のプレイリストは、clip 3を演奏してclip 5へフェード移動、clip 5を再生してclip 2へフェード移動、clip 2を再生してclip 1へフェード移動、clip 1を再生してclip 3へフェード移動、などと再生が続きます。



プレイリストボタンをクリックすると、対応するソースの左下隅に "p" の文字が表示されます (また、直接アクティブにすることができます)。

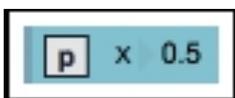
クロスフェード時間は、ソース内で指定したもので、この場合0.5秒です。

プレイリストが機能するためには、クリップがクロスフェード時間の2倍よりも長いことを確認する必要があります (プレイリストでは1つのクリップから、別のクリップにフェードするので)、さもないと、プレイリストは次のクリップに移動しません。

プレイリストのクロスフェード時間が0.5秒であれば、リスト内のクリップ長の最小値は1秒です。

プレイリストに動画と静止画を混在できません。

静止画像ソースを使用して、静止画像のプレイリストを作成できるが、次の静止画像にフェーディングする前に、各静止画像が表示される時間を指定する必要があります。クリップコントロール内のプレイリストボタン下にある数を調整することで、時間を秒単位で設定できます。



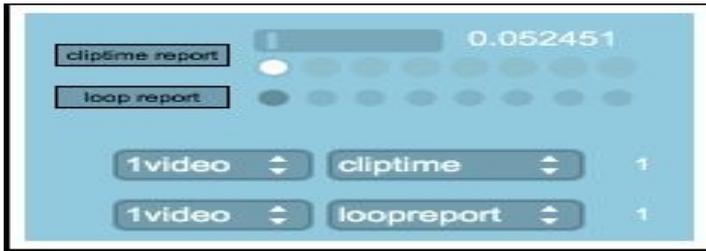
cliptimeレポートとloopレポート

クリップ制御タブの一番下で、ソースのクリップタイムとループのレポートをアクティブにできます。

“clicktime report” をクリックして、特定のソースのオン/オフを切り替えます。白丸 (1-8) は、クリップタイムのレポートがオンになっていることを示します。

“loop report” をクリックして、特定のソースのオン/オフを切り替えます。ループがレポートされたときには、対応する円が白く点滅します。

クリップタイムレポートとループレポートは、[ルータを介して他のイベントをトリガするために使用](#)することができます。



サウンドの追加

サウンドファイルをQuickTimeムービーに変換することにより、VPTプロジェクトにサウンドを追加できて、それをビデオ・モジュールのひとつとして再生することができます。QuickTime Proを使ってムービーにサウンドファイルを変換することもできます。あるいは、VPT内側から開けるように私が作った小さなヘルパーアプリVPTtrimmerをダウンロードできます。このアプリは、ビデオを扱いやすい他のフォーマットや解像度に変換することができます。

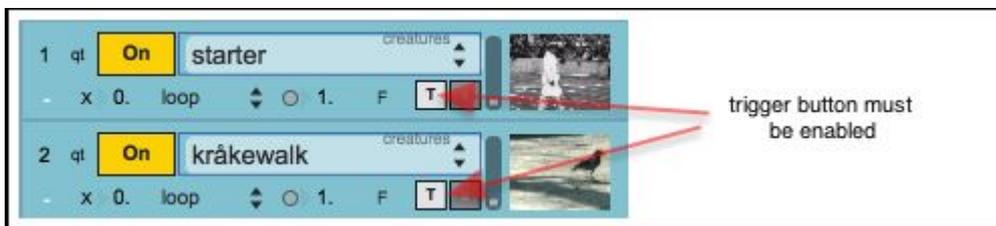
http://nervousvision.com/vpt/xtension/VPT_trimmer-mac.mxf.zip

http://nervousvision.com/vpt/xtension/VPT_trimmer-win.mxf.zip

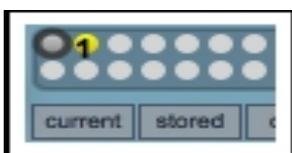
詳細は付録B VPTtrimmerを参照してください。

複数クリップを同時にトリガ

これを行うには、いくつかのやり方があります。VPTプリセットとsourceプリセットの利用、あるいは コントローラと組み合わせたルーターの利用、または キューリストからコマンド利用、などがあります。クリップがVPTプリセットで最初から始まることを確認するには、各ソースの T (トリガー) ボタンをクリックする必要があります。



その後、通常のVPTプリセット (ソースがプリセットに含まれている場合) を保存するか、sourceプリセットを作成することができます。プリセットを選択すると同時に、QuickTimeの動画は開始されます。



他のアプローチは、ルータでトリガを設定する（その際、トリガボタンを有効にする必要はありません）方法が有りますが、同じコントローラで各ソースのトリガパラメータを制御させることを忘れないください。

ctrl-nr	destination	parameter	range (min-max)
17	video	1 trig	0. 1.
17	video	2 trig	0. 1.

この場合、ボタンが組み込まれていてテストすることが容易なので、17番のコントローラを選択しました。



最後に、ルータにメッセージをフォーマットし、OSCを使用してcuelistからビデオをトリガすることができます。

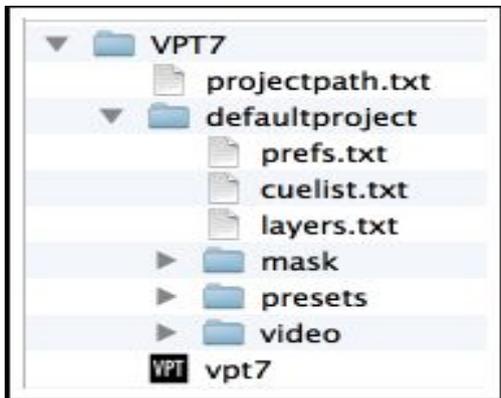


17はコントローラ番号、1は、ひとつのイベントにトリガーをかけるという意味の数値です。

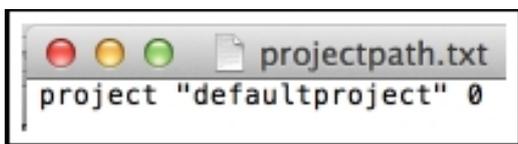
VPTフォルダー構造

VPT7をダウンロードし解凍すると、フォルダ構造は以下のようなになります。(Windowsの場合は、余分にいくつかのファイル、サポートフォルダがあります)

このフォルダには、VPTアプリケーションファイル、デフォルトのプロジェクトフォルダとprojectpath.txtが含まれます。projectpath.txtがVPTにVPTアプリケーションが使用するproject folderが何処かを伝えるので、projectpath.txtが同じフォルダ内に存在することが非常に重要です。



projectpath.txtを開くと、一行のテキストが表示されます。引用符で囲まれた名前は、現在のproject folderの名前です。番号0は、フォルダがVPTアプリケーションと同じフォルダにあることを示しています。何らかの理由で、どこか他の場所にproject folderを持ちたい場合は、全体のファイルパスを指定し、番号を1に変更する必要があります。

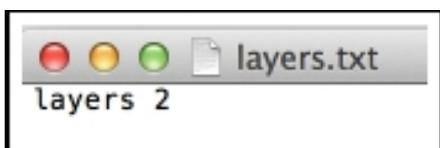


新しいプロジェクトを作成する最も簡単な方法は、defaultprojectフォルダを複製し、新しい名前を与えます。プロジェクトを切り替えるにはprojectpath.txt内のプロジェクトパスを変更してください。

プロジェクトフォルダには、デフォルトのビデオ(video)とマスク(mask)とプリセット(presets)のフォルダ、cuelist(cuelist.txt)、設定(prefs.txt)やレイヤー使用数情報(layers.txt)が含まれています。



layers.txtにはVPTが閉じられた時に使用されていたレイヤー数の情報が記録され、次にVPTが起動時した際、レイヤーを作成するためこのファイルを使用しています。



VPT 設定: prefs.txt

VPTの設定は、メッセージとそれに続く値のリストで構成されます。ここでVPTのいくつかのカスタマイズが行えます。値のみを変更し、メッセージは変更しないことが重要です。

fsaa 1/0 use hardware accelerated rendering. Default 1 //ハードレンダリングの利用

framerate in frames per seconds. Default 30 fps //フレームレート

previewframerate Default 15 fps (to make the general performance of VPT better) //プレビューのフレームレート

number_of_screens Default 1, maximum 3 when used in combination with a matroxtriplehead2go //スクリーン数

screenratio Default 4:3. Accepts anything as long as it follows the number:number format //スクリーン横縦比

screendivider 0/1. Default 0. If using more than 1 screen the screen divider can indicate the division between each screen. //スクリーン分割の有無

preview_width in pixels. Default 320 //プレビュー画面幅

menubar 1/0 Default 1. Menubar on when starting VPT (see appendix for Mavericks issue with menubar in fullscreen) //起動時メニューバーの有無

cursor_off 0/1 Default 0. Cursor off when starting VPT //起動時カーソルの有無

preview 1/0 Default 1. Preview window on or off. //プレビュー画面の有無

fullscreen 0/1 Default 0. Start VPT in fullscreen. //起動時フルスクリーン on/off

autostart preset/cue/off preset/cuenumber. Start automatically at a preset or cue number. //プリセット、キューの自動起動

drawcorners 1/0. Default 1. Show corner handles on active layer. //コーナーハンドルの表示

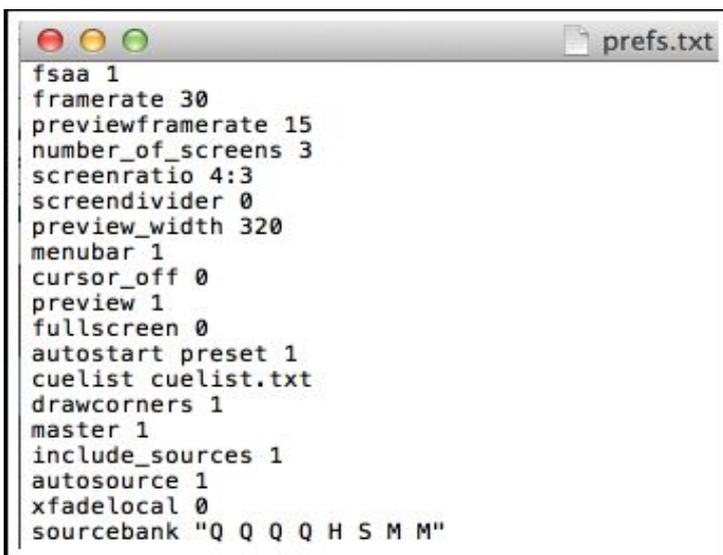
master 1/0. Default 1 VPT render engine on/off at startup //起動時のレンダラーエンジンon/off

include_sources 1/0. Default 1. Include sources when storing a VPT preset //VPTプリセット保存時にソース含めるon/off

autosource 1/0. Default 1. Automatically create a new source preset when storing a VPT preset(numbered from 1001 and up) //VPTプリセット保存時にsourceプリセットの自動作成on/off

xfadelocal 0/1. Default 0. Use crossfade times associated with the individual sources. If set to 0 crossfade times in the cuelist overrides //個別ソースでのローカルなクロスフェード時間の利用

Sourcebank list of 8 letters enclosed by quotes. This lets you customize your sourcebank according to the need of your project. Q=quicktime, H=hap, S=still, M=mix. //8文字でソースバンクの構成定義



```
fsaa 1
framerate 30
previewframerate 15
number_of_screens 3
screenratio 4:3
screendivider 0
preview_width 320
menubar 1
cursor_off 0
preview 1
fullscreen 0
autostart preset 1
cuelist cuelist.txt
drawcorners 1
master 1
include_sources 1
autosource 1
xfadelocal 0
sourcebank "Q Q Q Q H S M M"
```

cuelistタブ (VPTプリセットを使い cuelistを作成)

cuelistは、保存した複数のVPTプリセットの遷移のシーケンスを作成します。VPTプリセット（レイヤと映像ソースに関するすべての情報を意味する）はVPTの状態を保存するもので、動画や画像をレンダリングするわけではありません。2つのプリセット間の遷移を作成すると云うことは、VPTが（キーフレームと少し似てます）これらの2つの状態間を補間することを意味します。

cuelistは、コマンドとVPTプリセットや時間に関する情報を含む複数の行から成ります。

コマンド例:

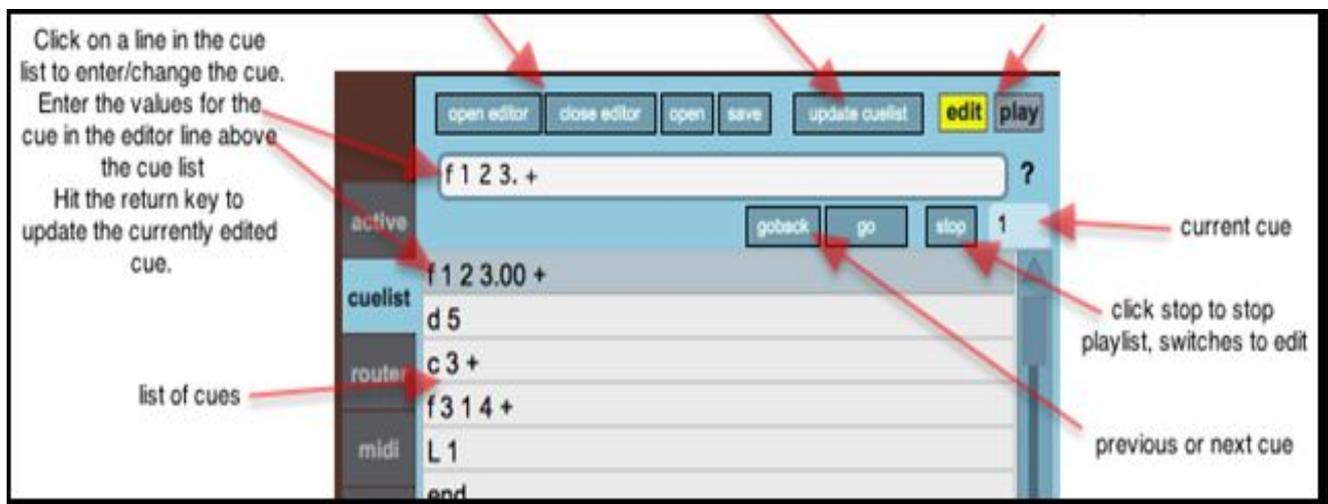
F/f : fades //あるプリセットから他のプリセットへ指定の時間（秒）でフェード移行

F 1 2 3 プリセット1から2へ3秒でフェード

C/c: cuts //あるプリセットへ直接移行

C 3 プリセット3にジャンプ

(手動でそれぞれのキューをトリガする時は、ほとんどの場合この二つのコマンドを使用します。)



cuelistを開始するにはそのキューの上でクリックします。

cuelistを停止するには、停止ボタン（または、編集ボタン）をクリックします。

現在のキューが終了した後、自動的に次のキューに行くには、スペースで区切られた + を追加します。

例:プリセット1から2でへ3秒でフェードし、次のキューに移り、プリセット3にジャンプします。

F 1 2 3 +

C 3

D/d: delay //次のキューに行く前に 指定時間（秒）の遅延を入れます

D 5 5秒遅延して次のキューに移動

例: (遅延時間が終わった後は、常に次のキューに移動しますので、後に+がありません)

F 1 2 3 +

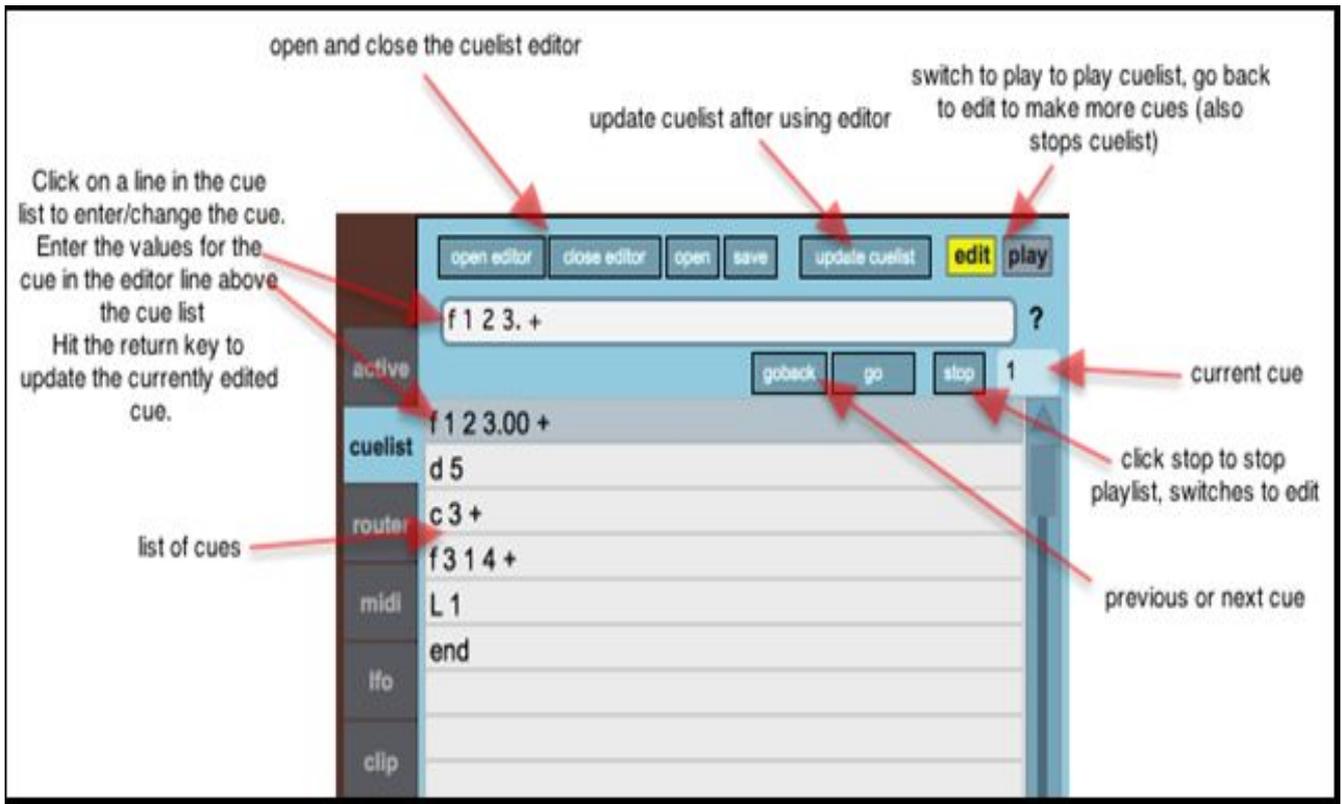
D 5

C 3

cuelistで継続的なループをしたい場合には、cuelist内の行にジャンプすることをVPTに伝えます。

L/l: jump //指定した行番号にジャンプします。

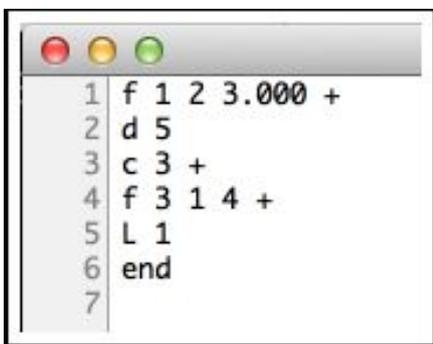
L 1 1行目にジャンプします。



変更したり、多くのキューを追加する場合には、エディタを使用する方がはるかに簡単です。open editor ボタンをクリックしてください。



リストの最後に"end"を置くことを忘れないでください。



テキストエディタを閉じた後、update cuelistボタンをクリックしてください (close editor ボタンでエディタを閉じた場合は必要ありません)。任意のテキストエディタでcuelistを作成し、cuelistタブ内で開くことができます。



cuelistのより高度な機能

sourceプリセットのcuelistを作成

S/s: 指定したsourceプリセット

S 2 sourceプリセット2 をアクティブにします

VPTプリセットの代わりに、sourceプリセットのリストを作成することができます。

作成方法は、sourceプリセットのセクションを参照してください。sourceプリセットには既にクロスフェード情報が含まれているので、sourceプリセット番号を指定するだけです。

OSC形式メッセージ送信

O/o: VPTへ OSC形式メッセージ送信 (OSC : Open Sound Control)

例) レイヤー1のフェードレベルを0.5にセットします。

```
O /layer1/fade 0.50
```

X/x: VPTから他のアプリケーション/コンピュータへ OSC形式メッセージを送信

OSC形式メッセージ送信先は、OSCEditorタブで設定する必要があります。

ルータにコントローラのフェード情報を送信

R/r: ルータにコントローラのフェード情報を送信。ルータの設定方法はルータのセクションを参照。

4パラメーター: コントローラー番号、開始値、終了値、時間

例) コントローラー番号1、値の変化が0から1、時間は5秒間です。

```
R 1 0. 1. 5.
```

(もし、ルーター内でコントローラー1がレイヤー1のフェードレベルに割り当てられていたら、このレイヤーのフェードレベルをキューリスト内の1行で制御できる、ということになります。)

routerタブ : コントローラー(midi/lfo 等用) をVPTパラメーターに割付け

ほとんどのVPTパラメータを制御できます。ルータで、コントローラとパラメータ間の割付を設定することができます。

コントローラは、多くのものになれます。MIDIタブ内で見かけるスライダーやボタンであったり、外部MIDIコントローラ（MIDIタブで設定します）であったり、lfoタブの内蔵lfo波形ジェネレータになることができたり、また、cuelistで設定（Rコマンドを使用）されるコントローラになる事もあります。

コントローラ番号 伝送先 パラメーター 値の範囲

ctrl-nr	destination	parameter	range (min-max)
1	video	rate	0. 1.
2	video	scrub	0. 1.
3	video	in	0. 1.
4	video	out	0. 1.
5	OFF	-	0. 1.
6	OFF	-	0. 1.

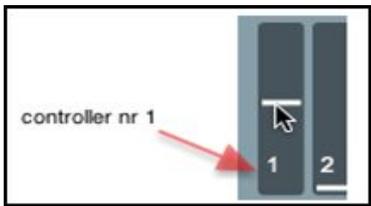
ctrl-nr (コントローラー番号)

コントローラは、常に番号で呼ばれています。ルータの最初の列は、コントローラ番号（ctrl-nr）です。デフォルトでは、コントローラ番号1、2、3等の順序にルーター行が設定されていますが、必要に応じてコントローラの番号を設定することができます。理解すべき重要なことは、ルータの最初の列に指定されたコントローラの番号は、使用しているコントローラに対応しなければならないということです。

例えば下図の場合、コントローラ1をMIDIタブのスライダ1とLFOタブのLFO1の両方が参照しており、両方の制御対象はコントローラ1のパラメータがマッピングしたものになります（この場合は、1videoのビデオ・レートです）。

ctrl-nr	destination	parameter	range (min-max)
1	video	rate	0. 1.
2	video	scrub	0. 1.

MIDI スライダ1



LFO1



destination(伝送先)

2番目の列では、伝送先タイプを指定します。制御用のデータを送信したい場合は、レイヤーまたはビデオソース（またはドロップメニュー内の項目）を選択します。

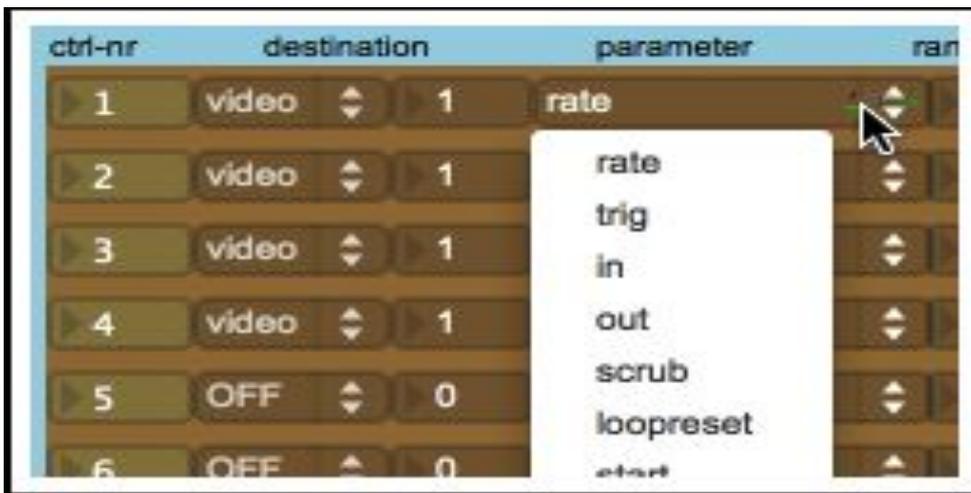


このドロップメニュー直後のナンバーボックスでは、具体的な伝送先を数値で指定します。もしドロップメニューからビデオを選択した場合、ナンバーボックスの数（1-8）を変えて、VPTに指示します。つまり、その数を1に設定したら、伝送先は1videoです。



parameter(パラメーター)

伝送先を設定した後、伝送先のどのパラメータかをドロップメニューで指定する必要があります。伝送先にビデオを選択しているので、ドロップメニューにはこれに関連する項目が出ます。下図の場合、レート（ビデオの速度）を選択しました。



range (範囲)

最後の2つの数字は、コントローラの入力値と伝送先パラメータとの関連を特定します。

通常は、VPT内の値は最低値が0で最高値は1に定義されており、数値は0と1の間になります。

たとえば、レイヤーのフェードが設定0のときには透明になり、設定0.5で半透明、設定1では不透明になります。

ビデオのレート（速度）の場合では、範囲を拡大することができます。0は、ビデオの停止を意味し、1は通常の速度です。-1は逆回しですが通常は速度です。したがって、範囲の下限値を-1、上限値を1になるように設定できます。

この場合、コントローラに割り付けられたスライダーが一番下にあるときには-1の値にマッピングされ、それが上にあるときは1にマッピングされることを意味します。



一つのコントローラで、複数のパラメータを制御する

同一のコントローラによって、ルータの複数行を制御できます。下図の場合、コントローラ1はレイヤ1とレイヤ2 両方のフェードレベルを制御します。

ctrl-nr	destination	parameter	range (min-max)
1	layer 1	fade	0. 1.
1	layer 2	fade	0. 1.
3	video 1	in	0. 1.

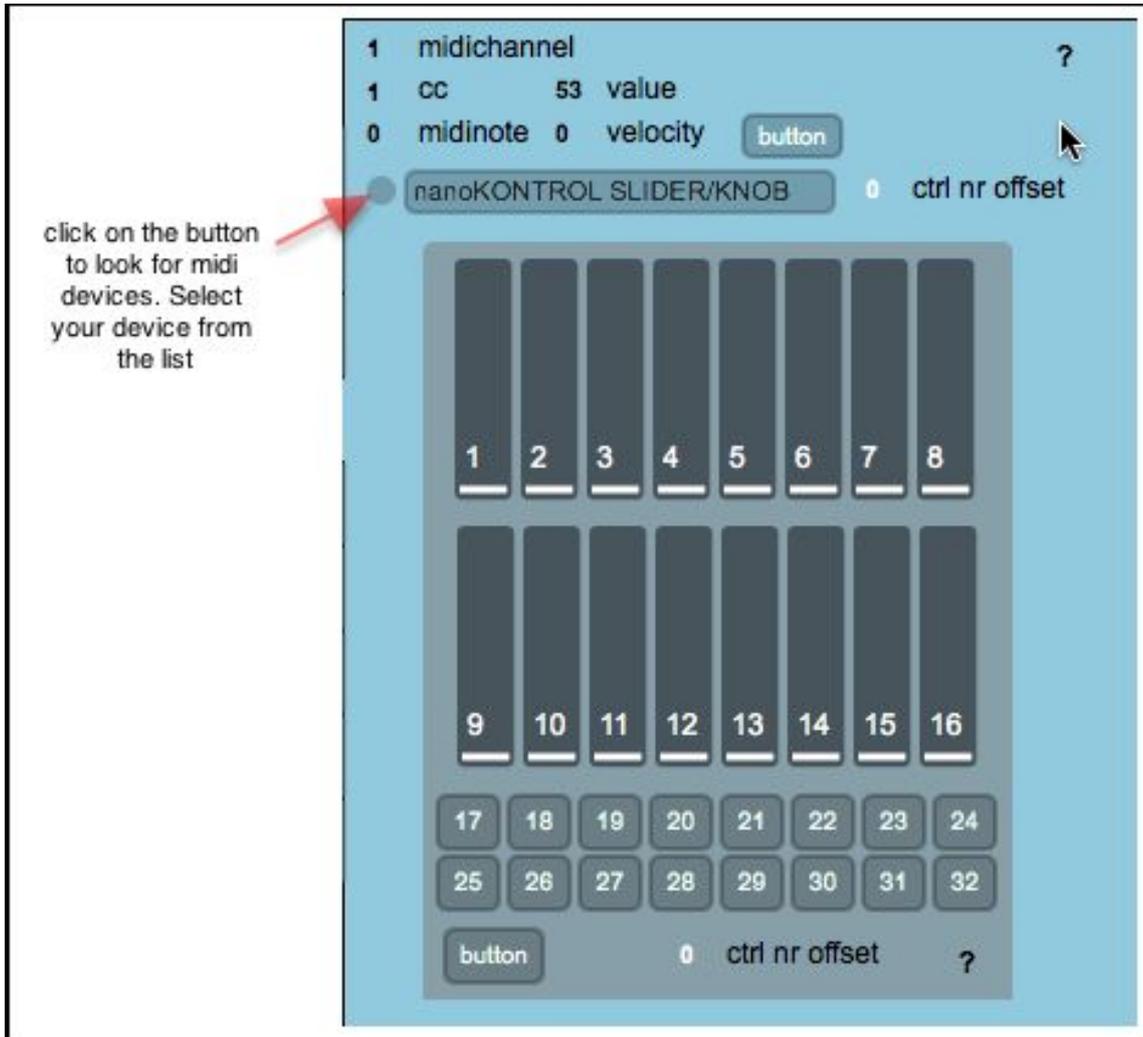
もし2行目のrange(範囲)を逆にすると、2つのレイヤが1つのコントローラを使用してクロスフェードになる、つまりレイヤ1が完全表示（1：不透明）になっている時は、レイヤ2が表示オフ（0：透明）になっています。

ctrl-nr	destination	parameter	range (min-max)
1	layer 1	fade	0. 1.
1	layer 2	fade	1. 0.
3	video 1	in	0. 1.

midiタブ : midi用 スライダーとボタン

VPTは、簡単に外部MIDIコントローラに接続することができます。メニューからデバイスを選択します (VPTを開始する前に、デバイスを接続します)。通常、ccの左にある数字はコントローラ番号に対応します (値は、valueの隣に、0から127の範囲内の数字で表示されます)。

VPTは、また、いくつかの組み込型スライダーやボタンを持っており、ルータと一緒に使用できます。



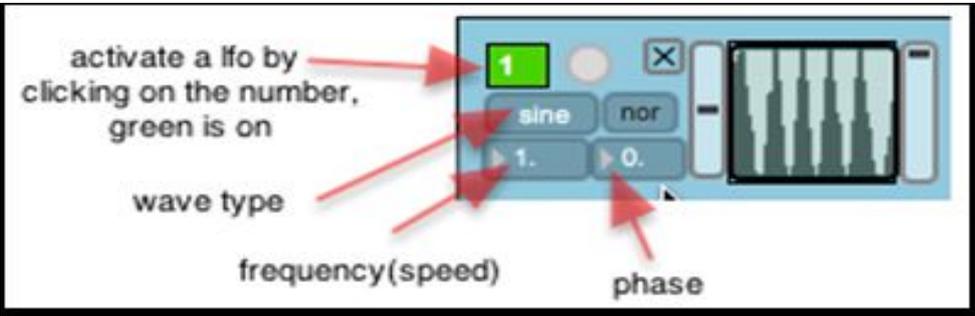
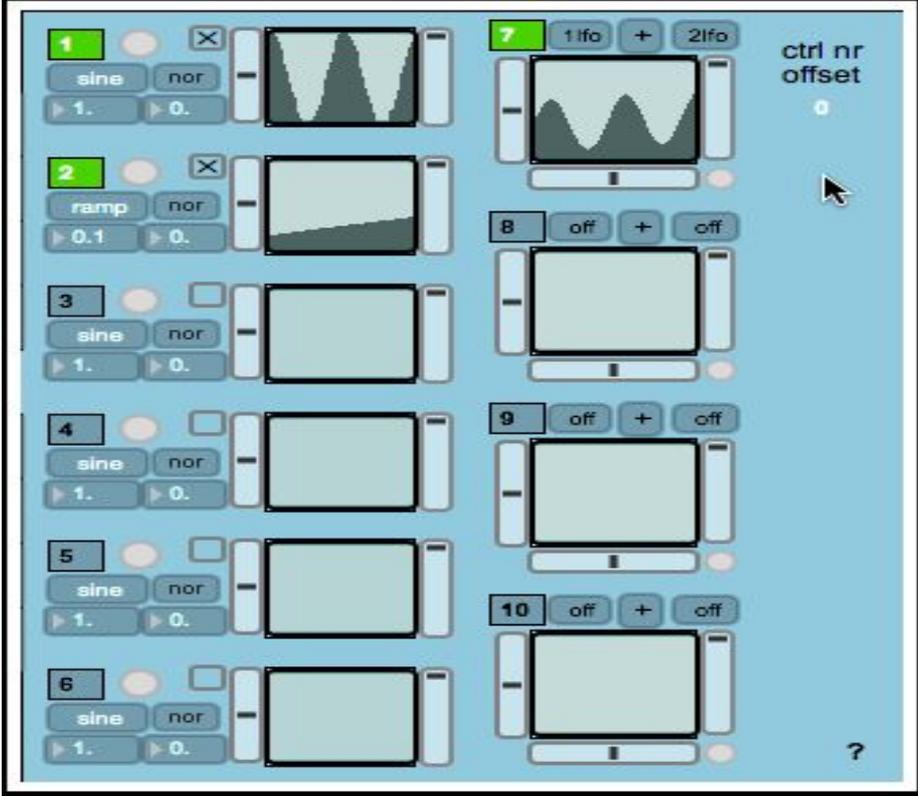
lfoタブ： 波形を使ったVPTパラメータの制御

LF0(Low Frequency Oscillator)つまり低周波発振器は、基本的には、ルータを介してVPTのパラメータ値の変化を自動化するものです。

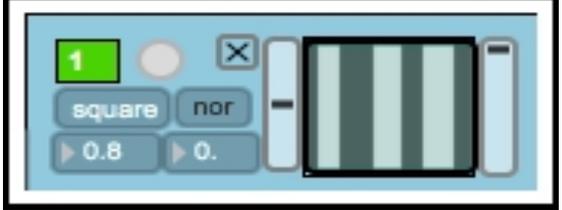
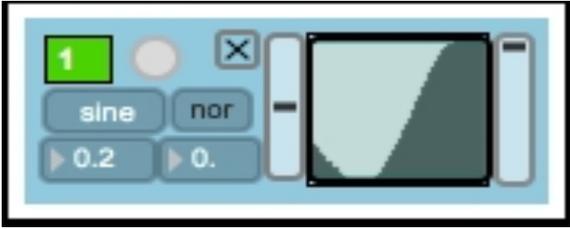
単一波形

合成波形

コントローラ番号とのオフセット値



単一波形の種類



OSCタブ : メッセージ送信 to and from VPT

VPTでは、OSC (Open Sound Control) と呼ばれるメッセージ形式を、VPT内部でのメッセージ送信だけでなく、他のアプリケーションや他コンピュータとのメッセージ送信/受信に使用されています。使用されるプロトコルはUDPです。

OSCは、他のアプリケーションの多くが使用するアドレス方式であり、かつ、パラメータにアクセスする非常に整然とした方法で、階層的なメッセージ構造を持っています。

メッセージの最初の部分が伝送先を示し、第二の部分は伝送するパラメータ値です。宛先アドレスは、左から右により詳細な情報になります。

簡単なサンプル :

レイヤー1のフェードレベルを制御する場合、制御パラメータは、最も一般的な部分であるレイヤーから始め、最も具体的な部分で終了させます。各メッセージレベルは/で区切られます。

```
/1layer アクセスしたいレイヤー
```

```
/fade アクセスしたいパラメータ
```

これらを一緒にすると /1layer/fade

そしてフェードパラメータの後に値を送信するので、最終的なOSCメッセージは次のようになります :

```
/1layer/fade 0.5
```

レイヤー2の輝度値にアクセスする場合は、以下の様に1つ深いレベルになります :

```
/2layer/brcosa/brightness 1.2
```

最初のビデオソースの速度を2に設定する場合は、最初に /sources、次にソース・モジュール /1video、そしてパラメータ(rate)およびその値(2)を追加します :

```
/sources/1video/rate 2.
```

メッセージ構造が、ルータでコントローラーの割付を作成する方法と非常によく似ていることがわかります。実際、VPT内部でルータがルータリストからOSCメッセージを作成しています。

OSC エディター

アプリケーション間でメッセージを送受信できるようにするには、受信側で受信ポートを設定(VPTでは6666)する必要があります。同じコンピュータ内で送受信している場合は、localhostのIPアドレス127.0.0.1を使用できます。他の場合は、コンピュータのIPアドレスを調べる必要があります、かつ同じネットワーク上にある必要があります。

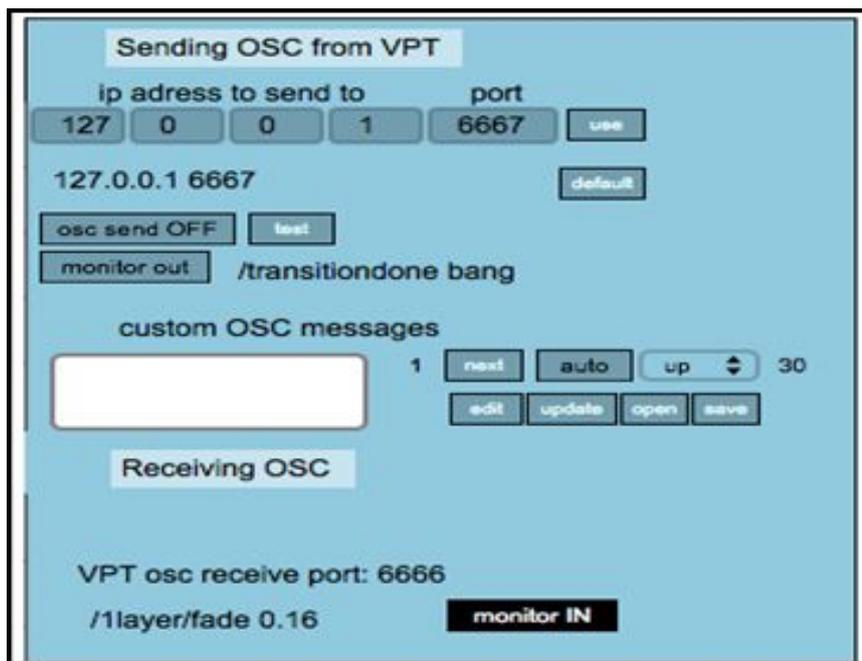
OSCエディタタブでは、設定および OSCメッセージのinとoutトラフィックを監視することができます。

VPTは、OSCメッセージを送信するデフォルトのアドレス(127.0.0.1)とポート(6667)を持っています。VPTと同じコンピュータ上の別のアプリケーションがポート6667で受信している場合、OSC送信ボタンがONならば、それがVPTからのOSCメッセージを受信します。(デフォルトはOFF)

VPTから送信されるメッセージ：

```
/preset i
/cue i
/cuetrig t
/transitiondone t
/loopreport i t sourcenr bang
/cliptime i f sourcenr relative cliptime (0. to 1.)
```

入りと出両方のトラフィックをモニターできます。



VPT内部からのOSCメッセージ送信

VPTがOSCを使用するいくつかの方法があります：

cuelistでは、VPTの任意のパラメータにOSCメッセージを送信するために、0/ oコマンドを使用できます
0 /1layer/fade 0.5

またcuelistでは、X/ xコマンドを使用してVPTからOSCメッセージを送信することができ、ここでは基本的に任意のコマンドを送信することができます。

外部からVPTへのOSCメッセージ送信

OSCをサポートする多くのアプリケーションやデバイスがあります、例えば、Processing、MaxJitter、PD、OpenFrameworks。また、ArduinoからもOSCメッセージを送信できます。(または、シリアルポートを介してデータを送信することができる他のデバイス、詳細はシリアルセクションを参照してください)

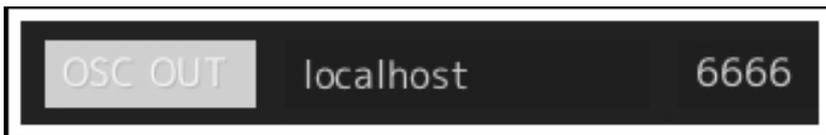
VPTで動作するOSCメッセージの設定やタブレット上のインターフェイス作成が可能なスマートフォンやタブレットのアプリがいくつかあります。

Duration: OSCを使用する無料タイムラインツール

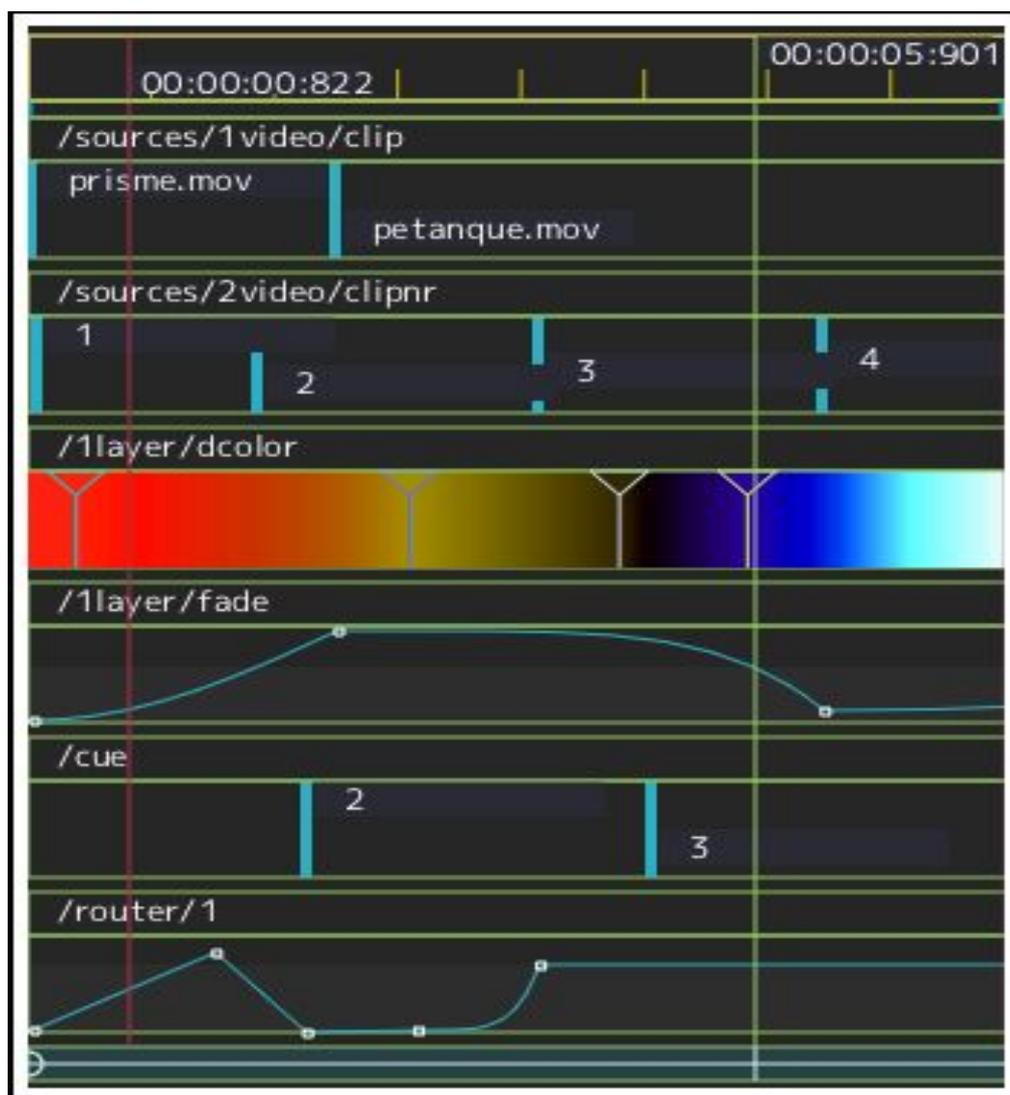
私は、ジェームズ・ジョージ作成のアプリ [Duration](#) を試してみることをお勧めします、このアプリはOSCを使用したVPTを制御する別の方法を提供します。このアプリの使用法のドキュメントはウェブサイトをご覧ください。



DurationをVPTと一緒に動かすためには、VPTがOSCメッセージ受信を待っているポート番号（それは6666）と通信できるように設置する必要があります。また、ipアドレスも特定する必要があります。(DurationとVPTが同じコンピュータ上ならば、localhost (127.0.0.1)です。)



VPTでどのようにDurationを使うことができるか、設定の例をいくつか示します。



VPT7用の VideoTriggerとSoundtrigger

VPTに制御データを送信するためにOSCを使用するVPT用アプリケーションが他に2つあり、セットアップ方法はルーターと非常に良く似ています。

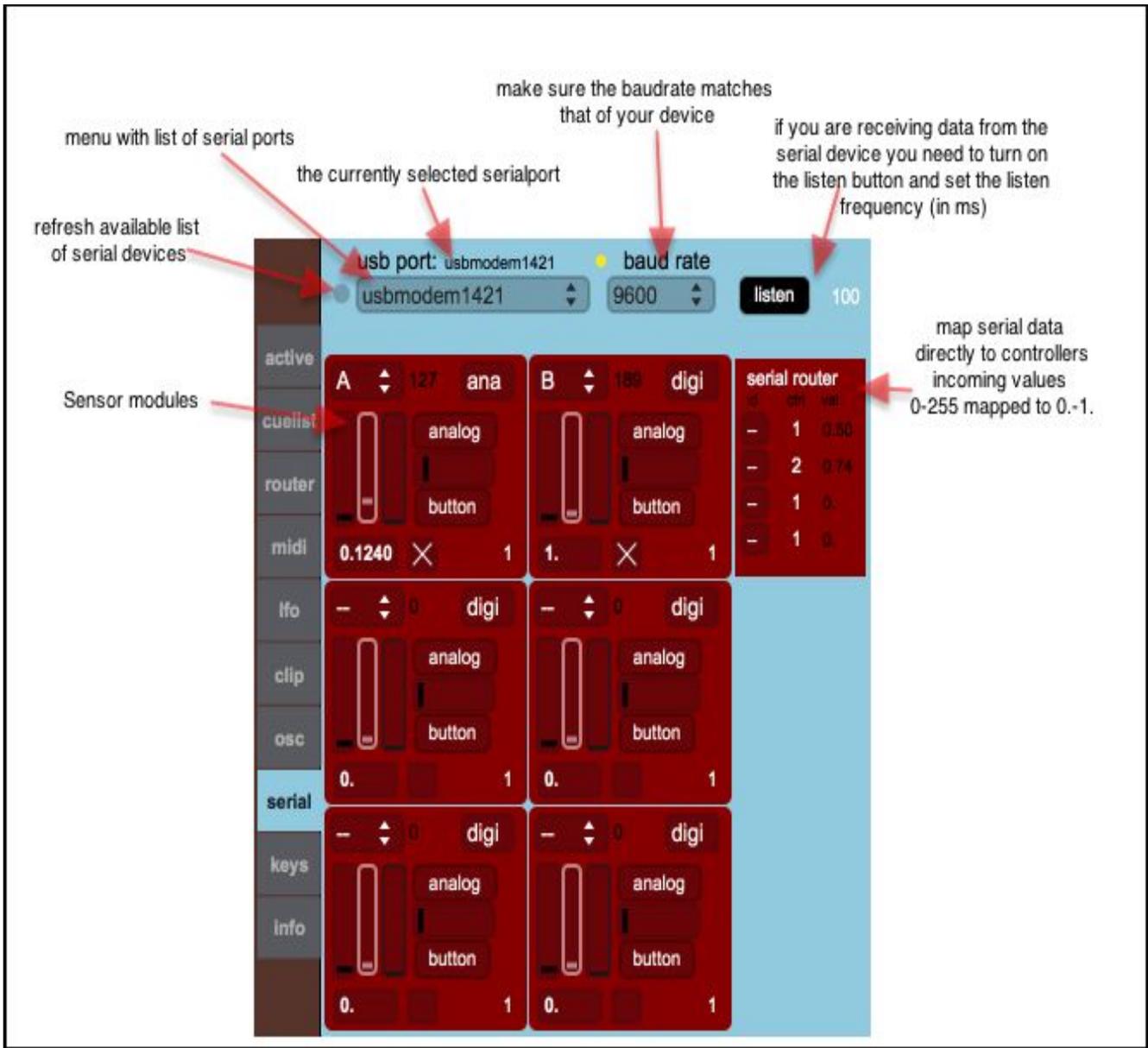
VideoTriggerは、VPTのイベントのトリガとしてカメラを使用することができます、SoundTriggerは、音声入力を使用したVPTイベントを制御することができます。詳細については、付録を参照してください。

serialタブ : シリアル通信

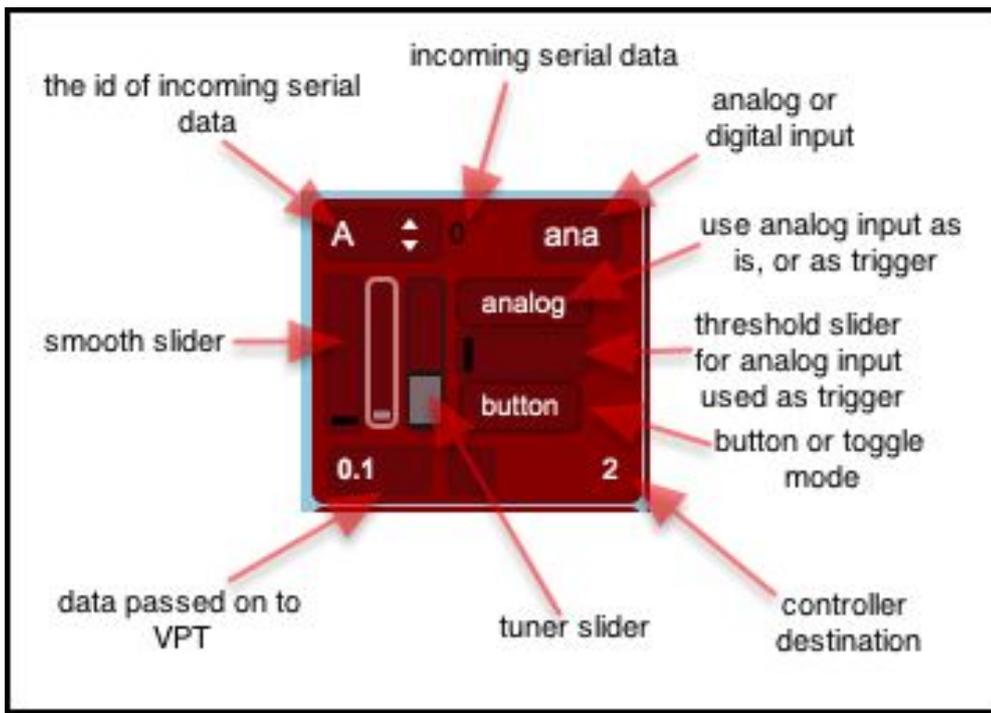
VPTは、Arduinoマイクロコントローラのようなシリアルデバイスと通信することができます。これは、センサーやスイッチがVPTの動作を制御できることだけではなく、VPTがLEDライト、モーター、サーボモータなどを制御できることを意味しています。

シリアルタブで、シリアルデバイスへの接続を設定出来ます、Arduinoはこの例に該当します。メニューからシリアルポートを選択することなどMIDIデバイスと同様の方法でシリアルデバイスを設定します。

VPTを開始した後デバイスを接続した場合には、メニューの左側にある更新ボタンをクリックします。



センサーモジュール



smooth slide(スムーズスライダ) :

センサーの出力は、しばしば急変することがあります。このスライダは、あまり急激でない変化を提供します。

tuner slider(チューナーズスライダ) :

利用出力値の範囲が、センサー全出力範囲を使わないないことがあります。チューナーズスライダは、実用範囲を調整するために使用します。

クリックし垂直方向にドラッグして、範囲を調整します。

shift+clickを現最大値の上側で行うと、最大値を変更できますが、最小値は変わりません。

shift+clickを現最小値の下側で行うと、最小値を変更できますが、最大値は変わりません。

option+clickオして、現在の範囲の内側で垂直方向にドラッグすると、範囲のサイズを変更できます。

ctrl(win)/cmd(apple) +click して、垂直方向にドラッグすると、現在の範囲を移動できます。

Arduino(アルデュイーノ)からVPTへのセンサ入力

VPTにArduino(アルデュイーノ)からのセンサデータを送信するためには、VPTが理解できるようにデータをフォーマットする必要があります。

データの前に識別子 (A-H) を追加します。

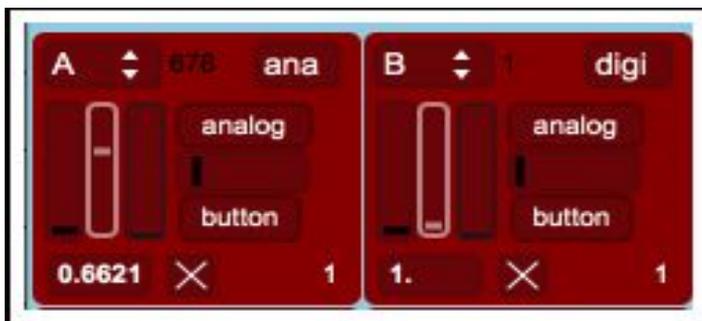
したがって、下図のArduinoのスケッチ(プログラム) ではanalogSensorデータサンプルは、A 678 (アナログセンサデータは、0から1024の値である) にフォーマットされ、digitalSensorのデータは、例えば B 1 になります。

```
int digitalSensor, analogSensor;

void setup()
{
  // start serial port at 9600 bps:
  pinMode(7, INPUT); //configure digital pin as input
  digitalWrite(7, HIGH);
  Serial.begin(9600);
  delay(100);
  Serial.flush();
  delay(1000);
  Serial.println(-1, DEC); //send -1 to let VPT serial port is ready
}

void loop()
{
  analogSensor = analogRead(0);
  digitalSensor = digitalRead(7);
  // delay 10ms to let the ADC recover:
  delay(10);
  Serial.print("A ");
  Serial.println(analogSensor, DEC);
  Serial.print("B ");
  Serial.println(digitalSensor, DEC);
}
```

VPTでは、センサモジュールに適切なIDとデータ型を設定する必要があるので、下図の様に、最初のモジュールはidがAでデータタイプがアナログ入力に設定され、一方、第2のモジュールはidがBでデータタイプがデジタル入力に設定されています。



VPTからのシリアルデータ送信：VPTからライト、モーターなどを制御する方法

VPTからのシリアル出力を利用して簡単にLEDやモータを制御できます。

下図の様に、ルータで行う方法があります。シリアルを選択し、後S1-S8を選び、出力範囲を決めます。コントローラ1は、0から255までの範囲で、一方、コントローラ17は、0から1までの範囲に設定されていることに注意してください。その理由は、コントローラ1では、電力制御可能なPWM(PulseWideModulation)アナログ出力を制御し、コントローラ17は（オン/オフ）のデジタル出力を制御するためです。

VPTは入力データ受信と同様の方法でシリアル出力をフォーマットします：最初にID、その後データ。

Arduinoに来るときには、S1、S2は1と2に変換されるので、データは次のようにフォーマットされます。

1 127 (半調光)

2 1 (オン)

コントローラ番号	伝送先	パラメーター	値の範囲
----------	-----	--------	------



Arduinoのスケッチ（プログラム）は次のようになります。

```
int id = 0; //the two bytes from the incoming data from VPT
int data;
int led1 = 9; //the led hooked up to the PWM output
int led2 = 13; //the led hooked up to the digital output

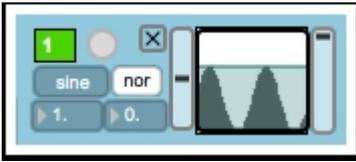
void setup()
{
  pinMode(led2, OUTPUT); //configure digital pin as output
  Serial.begin(9600);
  delay(100);
  Serial.flush();
  delay(1000);
  Serial.println(-1,DEC);
}

void loop()
{
  while ((Serial.available() > 1)){

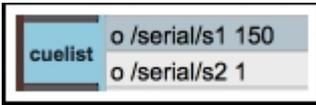
    id = Serial.read();
    data = Serial.read();

    if (id == 1) {
      analogWrite(led1,data);
    }
    if (id == 2) {
      digitalWrite(led2,data);
    }
  }
}
```

このセットアップで、例えば、LEDの調光を（コントローラ 1 に割付けた）LF0波形で制御できます。



また、0/oコマンドを使用してcuelistからシリアルデータを送信することができます



サーボモータをVPTから制御

シリアルデータを送信するために前節と同じ設定を使用しますが、今回のArduinoスケッチ（プログラム）は少し異なったやり方を用います。

このスケッチでは、データを無視してデータのIDだけを見てサーボモータを制御します。サーボ位置1に移動する場合は、 /s1 1 （ idは s1、データは任意の数）で、サーボ位置2に移動する場合は、 /s2 1 （ データは任意の数）をVPTから送信します。

```
//Example of how to control a servo motor from VPT

#include <Servo.h>

Servo myservo1; // create servo object to control a servo

int id = 0; //the two bytes from the incoming data from VPT
int data;

void setup() {
  myservo1.attach(3); // attaches the servo on pin 3 to the servo object
  Serial.begin(9600); // opens serial port, set baud rate
  delay(100);
  Serial.flush();
  delay(1000);
  Serial.println(-1,DEC);
}

void loop() {
  while (Serial.available() > 1) { // send data only when a byte is received
    id = Serial.read(); // read the incoming byte
    data = Serial.read(); //we are ignoring this byte
    switch (id) {
      case 1:
        myservo1.write(0);
        break;
      case 2:
        myservo1.write(150);
        break;
    }
  }
}
```

VPT用のArduinoのスケッチをダウンロードすることができます：

http://nervousvision.com/vpt/xtension/VPT7_arduino.zip

シリアルルータ

ルータに直接シリアルデータを送る場合は、シリアルルータが役に立つでしょう。VPTは、0-255（1バイト）の範囲の値を、0-1の範囲の浮動小数点の数値に変換します。

メニューからシリアルデータID（A-H）を選択し、このIDをコントローラ番号に関連付けることで、着信データをマッピングできます。

serial id	controller nr	controller value
A	1	0.50
B	2	0.74
-	1	0
-	1	0

ArduinoからVPTにOSC形式のメッセージを送信

以下はシリアルポートを介してOSCメッセージを送信するための簡単なスケッチです、これは、簡単に、たとえばArduinoからcuelistを制御できることを意味しています。

```
void setup()
{
  // start serial port at 9600 bps:
  Serial.begin(9600);
}

void loop()
{
  Serial.println("0 /1layer/fade 0.");
  delay(500);
  Serial.println("0 /1layer/fade 1.");
  delay(500);
}
```

VPTプリセットに関する その他機能

VPTプリセットを削除するには、インターフェース画面のレイヤーとプリセット作成セクションのプリセットリストから削除対象を選択し、"clear current" ボタンをクリックすることで可能です。



また、"+" 印をクリックすると、次に、すべてのVPTプリセットを削除するオプション機能が現れます。



"current" ボタンをクリックすることで、VPTパラメータの現在の状態の詳細を見つけることができます。VPTパラメータの全ての値を含む長いリストのウィンドウが開きますが、これらはプリセットを保存するときに保存されるものです。



Client Objects [vpt]					
	Name	Priority	Interp		Data
<input checked="" type="checkbox"/>	sources	-1	⇄ thr...	0.00	1001
<input checked="" type="checkbox"/>	1layer		⇄		
<input checked="" type="checkbox"/>	mesh		⇄		
<input checked="" type="checkbox"/>	gridsize	0	⇄ linear		0
<input checked="" type="checkbox"/>	position	0	⇄ linear		0
<input checked="" type="checkbox"/>	on	0	⇄ linear		0
<input checked="" type="checkbox"/>	zoom		⇄		
<input checked="" type="checkbox"/>	rota	0	⇄ linear		0.08377
<input checked="" type="checkbox"/>	yanchor	0	⇄ linear		0.051239
<input checked="" type="checkbox"/>	xanchor	0	⇄ linear		0.890487
<input checked="" type="checkbox"/>	yzoom	0	⇄ linear		1.190599
<input checked="" type="checkbox"/>	xzoom	0	⇄ linear		0.5
<input checked="" type="checkbox"/>	on	0	⇄ linear		1
<input checked="" type="checkbox"/>	edgeblend		⇄		
<input checked="" type="checkbox"/>	inv	0	⇄ linear		0
<input checked="" type="checkbox"/>	down	0	⇄ linear		0.
<input checked="" type="checkbox"/>	right	0	⇄ linear		0.
<input checked="" type="checkbox"/>	up	0	⇄ linear		0.
<input checked="" type="checkbox"/>	left	0	⇄ linear		0.
<input checked="" type="checkbox"/>	on	0	⇄ linear		0
<input checked="" type="checkbox"/>	brcosa		⇄		
<input checked="" type="checkbox"/>	saturation	0	⇄ linear		1



"stored" ボタンをクリックすると別ウィンドウが現れ、あなたが作ったVPTプリセットの保存データ値に関する情報が明らかになります。

Storage Slots [vpt]			
	Name	0	1: pres...
<input checked="" type="checkbox"/>	sources	1001	1001
<input checked="" type="checkbox"/>	1layer		
<input checked="" type="checkbox"/>	mesh		
<input checked="" type="checkbox"/>	gridsize	0	0
<input checked="" type="checkbox"/>	position	0	0
<input checked="" type="checkbox"/>	on	0	0
<input checked="" type="checkbox"/>	zoom		
<input checked="" type="checkbox"/>	rota	0.086...	0.08377
<input checked="" type="checkbox"/>	yanchor	0.379...	0.051...
<input checked="" type="checkbox"/>	xanchor	0.697...	0.890...
<input checked="" type="checkbox"/>	yzoom	1.38921	1.190...
<input checked="" type="checkbox"/>	xzoom	0.5	0.5
<input checked="" type="checkbox"/>	on	1	1
<input checked="" type="checkbox"/>	edgeblend		
<input checked="" type="checkbox"/>	inv	0	0
<input checked="" type="checkbox"/>	down	0.	0.
<input checked="" type="checkbox"/>	right	0.	0.
<input checked="" type="checkbox"/>	up	0.	0.
<input checked="" type="checkbox"/>	left	0.	0.
<input checked="" type="checkbox"/>	on	0	0
<input checked="" type="checkbox"/>	hbonea		

同様にソースプリセットのセクションでも、現在または全てのソース・プリセットをクリアするためのボタンが表示されます、もちろん情報も現在のアクティブな値と格納された値です。



Client Objects [sources]				
Name	Priority	Interp		Data
<input checked="" type="checkbox"/> videobank01		⇄		
<input checked="" type="checkbox"/> videopath	-1	⇄ none		arkiv:/moovs/constructions/
<input checked="" type="checkbox"/> loop	0	⇄ none		1
<input type="checkbox"/> xfade	0	⇄ linear		0.5
<input checked="" type="checkbox"/> video	0	⇄ none		gatekryssT1
<input checked="" type="checkbox"/> rate	0	⇄ linear		1.
<input checked="" type="checkbox"/> on	0	⇄ none		1
<input checked="" type="checkbox"/> alpha	0	⇄ none		0
<input checked="" type="checkbox"/> resolution	0	⇄ none		F
<input checked="" type="checkbox"/> refreshrate	0	⇄ linear		1
<input checked="" type="checkbox"/> volume	0	⇄ linear		0.
<input checked="" type="checkbox"/> autotrig	0	⇄ linear		0
<input checked="" type="checkbox"/> in	0	⇄ linear		0
<input checked="" type="checkbox"/> out	0	⇄ linear		0
<input checked="" type="checkbox"/> videobank02		⇄		
<input checked="" type="checkbox"/> videopath	-1	⇄ none		arkiv:/moovs/alpha/
<input checked="" type="checkbox"/> loop	0	⇄ none		1
<input type="checkbox"/> xfade	0	⇄ linear		0.5
<input checked="" type="checkbox"/> video	0	⇄ none		radiowave

Storage Slots [sources]					
Name	1001	1002	1003	1004	
<input checked="" type="checkbox"/> videobank01					
<input checked="" type="checkbox"/> videopath	default	default	default	<i>default</i>	
<input checked="" type="checkbox"/> loop	1	1	1	<i>1</i>	
<input type="checkbox"/> xfade	3.	0.5	0.5	<i>0.5</i>	
<input checked="" type="checkbox"/> video	prism...	prism...	prism...	<i>prism...</i>	
<input checked="" type="checkbox"/> rate	1.	1.	1.	<i>1.</i>	
<input checked="" type="checkbox"/> on	1	1	1	<i>1</i>	
<input checked="" type="checkbox"/> alpha	0	0	0	<i>0</i>	
<input checked="" type="checkbox"/> resolution	F	F	F	<i>F</i>	
<input checked="" type="checkbox"/> refreshrate	1	1	1	<i>1</i>	
<input checked="" type="checkbox"/> volume	0.	0.	0.	<i>0.</i>	
<input checked="" type="checkbox"/> autotrig	0	0	0	<i>0</i>	
<input checked="" type="checkbox"/> in	0	0	0	<i>0</i>	
<input checked="" type="checkbox"/> out	0	0	0	<i>0</i>	
<input checked="" type="checkbox"/> videobank02					
<input checked="" type="checkbox"/> videopath	default	default	default	<i>default</i>	
<input checked="" type="checkbox"/> loop	1	1	1	<i>1</i>	
<input type="checkbox"/> xfade	3.	0.5	0.5	<i>0.5</i>	
<input checked="" type="checkbox"/> video	vpt07....	vpt07....	vpt07....	<i>vpt07....</i>	
<input checked="" type="checkbox"/> rate	1	1	1	<i>1</i>	

付録A : OSCコマンド

VPT用 OSCコマンドの一覧

(このリストは、[オンライン](#)も利用できます)

各コマンドは、コマンド名と、その後にくるメッセージのタイプで、表現されます：

i =整数 (-1,0,1,2,3 etc)

f=不動小数点 (0.1,0.45,2.22 etc)

s=文字列 (prism,black,chess etc)

t=command はトリガーなので、その後のメッセージは不要

構文の例

レイヤ3のフェードレベルを0.5 (半透明) に設定するには、OSCメッセージは次のようになります。

```
/3layer/fade .5
```

1sourceのレートをもと(通常2倍の速度) に設定するには、OSCメッセージは次のようになります。

```
/sources/1video/rate 2.
```

フルスクリーンにするには、OSCメッセージは次のようになります。

```
/fullscreen
```

1sourceのランダムなクリップを再生するには、OSCメッセージは次のようになります。

```
/sources/1video/random
```

コントローラ1の値を0.3することを、ルータに、送信するためには、OSCメッセージは次のようになります。

```
/router/1 .3
```

sources

```
/sources
```

```
/1video,/2video,/3video,/4video,/5video,/6video,/7video,/8video
```

quicktime module

```
/clip s name of video
```

```
/clipnr i (relative position in sourcemenue, 0:off, 1:first video, 2:second video etc)
```

```
/rate f (1. normal speed, -1 normal speed backwards)
```

```
/loop i (0:loop off, 1:loop,2:palindrome(loop backnforth),3:loop once)
```

```
/xfade f (0.,1.)
```

```
/resolution i (0:full,1:2/3,2:1/2,3:1/3,4:1/4,5:1/8,6:1/16)
```

```
/trig t
```

```
/on i (0,1)
```

```
/vol f (0.,1.)
```

```
/pan f (0.,1.)
```

```
/loopreport i (0,1)
```

```
/last t play last clip in the menu (often used together with recording live source directly
```

to a

source folder)

```
/random t
```

```
/scrub f (0.,1.)
```

```
/start t
```

```
/stop t
```

```
/in f (0.,1.) relative in-point
```

```
/out f (0.,1.) relative out-point
```

```
/looppreset t
```

```
/playlist i i i ... (list of clipnumber to be used in playlist)
/play i (0,1) activate playlist
/cliptime i (0,1) report cliptime
/alpha i (0,1) use video with alpha channel (default:0)
/com
```

still module

```
/clip s
/clipnr i
/xfade f
/resolution i
/refresh i (0:33ms,1:100ms,2:500ms,3:2000ms) how often still image is refreshed
/on i
/playlist i i i i...
/play i
/slide_length f (time in seconds)
```

mix module

```
/A i (0:off,1:1video,2:2video etc according to sourcelist in menu)
/B i (0:off,1:1video,2:2video etc according to sourcelist in menu)
/mix f (0.,1.)
/mixtype i (blendmode 0:normal alphablend, the others according to blendmode list in menu)
```

hap module

```
/clip s
/clipnr i
/rate f
/loop i
/xfade f
/on i
/trig t
/vol f
/pan f
/loopreport t
/last t
/random t
/scrub f
/start t
/stop t
/in f
/out f
/looppreset t
/playlist i i i i...
/play i
```

/1cam,/2cam

```
/on i (0,1)
/rec i (0:off,1:on)
/recdest (0:none, 1-8:1video-8video)
```

/1solid,/2solid

```
/on i (0,1)
/resolution i (0:4 3,1:80 60,2:320 240,3:640 480,4:1024 768) pixelsize
```

```

/refresh i (0:33ms,1:100ms,2:500ms,3:2000ms) how often solid source is refreshed
/1syphon,/2syphon,/3syphon,/4syphon
/on i (0,1)
/update t (check for new syphon servers)
/server s servername
/serverlist i (from server list)

```

Layer

```

/1layer,/2layer etc /0layer: active layer

/fade f (0.,1.)
/rgb f f f (0.,1.)
/color f f f f (rgba 0.,1.)
/dcolor i i i (rgb 0,255)
/red f (0.,1.)
/green f (0.,1.)
/blue f (0.,1.)
/blendmode i (0:normal, 1:additive, 2:multiply)
/layername s
/layerorder i
/scalex f
/scaley f
/posx f
/posy f
/cornerpin
  /upper_left f f (x y 0.,1.)
  /lower_left f f (x y 0.,1.)
  /upper_right f f (x y 0.,1.)
  /lower_right f f (x y 0.,1.)
  /com direct messages to jit.gl.cornerpin
/source s (1video,2video etc)
/flip
  /on i (0,1)
  /fliptype i (0: normal,1:hor flip,2:ver flip,3:hor+ver flip)
/tile
  /on i (0,1)
  /xtile i (0,14 0:normal, 1: 1/2, 2: 2/2, 3: 1/3,.....14: 5/5)
  /ytile i (0,14 0:normal, 1: 1/2, 2: 2/2, 3: 1/3,.....14: 5/5)
/zoom
  /on i (0,1)
  /xzoom f
  /yzoom f
  /xanchor f
  /yanchor f
  /rota f
/blur
  /on i (0,1)
  /blur f
/mblur
  /on i (0,1)
  /mblur f
/brcosa
  /on i (0,1)

```

```

/brightness f
/contrast f
/saturation f
/mask
/on i (0,1)
/source s (maskname)
/inv i (0,1)
/blur_on i (0,1)
/blur f
/switch i (0,1)
/moving i (0,1) use moving mask (requires a quicktime video in mask folder)
/points list (f,f,f,f...f,f) list ifof x y points of mask
/edgeblend
/on i (0,1)
/left f (0.,1.)
/down f (0.,1.)
/right f (0.,1.)
/up f (0.,1.)
/inv i (0,1)
/mesh
/on i (0,1)
/gridsize i
/position (f,f,f,f...f,f) list ifof x y points of mesh
/trig

```

presets

```

/preset i go to specified preset
/presetprev t
/presetnext t
/store0 store current state of VPT to temporary preset 0

```

cues

```

/cue i go to specified cue
/cuenext t
/cueprev t
/cueplay i (0:edit,1:play)

```

sourcepresets

```

/sourcepreset i go to specified source preset
/sourcenext t
/sourceprev t

```

router

```

/router
/1(/2,/3 etc) f let ^s you send controller data to the router
/router-preset i switch between the stored router presets

```

lfo

```

/1lfo /2lfo /3lfo /4lfo /5lfo /6lfo /7lfo /8lfo /9lfo /10lfo
/speed f 1. is normal speed
/phase f
/val f
/range f

```

```
/lfomix f (0.,1.)  
/on i (0,1)  
/wave (0:sine,1:ramp,2:triangle,3:square)  
/waveinv i (0,1)
```

serial communication

```
/serial  
/s1(/s2,/s3,/s4,/s5,/s6,/s7,/s8,/s9) f/i/s sends formatted messages out the serial port /s1 .5  
appears as 1 .5
```

osc out (default destination 127.0.0.1 port 6667)

OSCアウトがOSC-エディタタブで有効になっている場合、これらのメッセージは、OSCを使用してVPTから送出されます。ループレポートとcliptimeレポートは、対応するソースに対してオンにする必要があります（これは、クリップ制御タブ内で行うか、または /sources/1video/loopreport 1 /sources/1video/cliptime 1を使用して行うことができます）。

```
/preset i  
/cue i  
/cuetrig t  
/transitiondone t  
/loopreport i t sourcenr bang  
/cliptime i f sourcenr relative cliptime (0. to 1.)
```

div

```
/onoff i (0,1) turn VPT engine on/off
```

```
/fullscreen i (0,1)
```

```
/focus i (make layer i the active layer)  
/copy t copy from current active layer  
/paste t paste to current active layer
```

```
/cornerselect i (1 UL,2 UR, 3 LL, 4 LR)  
/corner_xy f f (xy position of active corner 0. to 1.)
```

```
/drawcorners i (0,1)
```

```
/mastercorner  
/enable i (0,1)  
/draw i (0,1)  
/reset t
```

```
/blind i (0,1) freezes video on the output window, let you continue to work on preview window  
/blackout i (0,1) makes output window black while you can continue to work on preview window
```

```
/masterfade f (0.,1.) a masterfader for all layers  
/framerate i framerate for output (default set to 30 fps in prefs)  
/pframerate i framereate for preview window (default set to 15 fps in prefs)
```

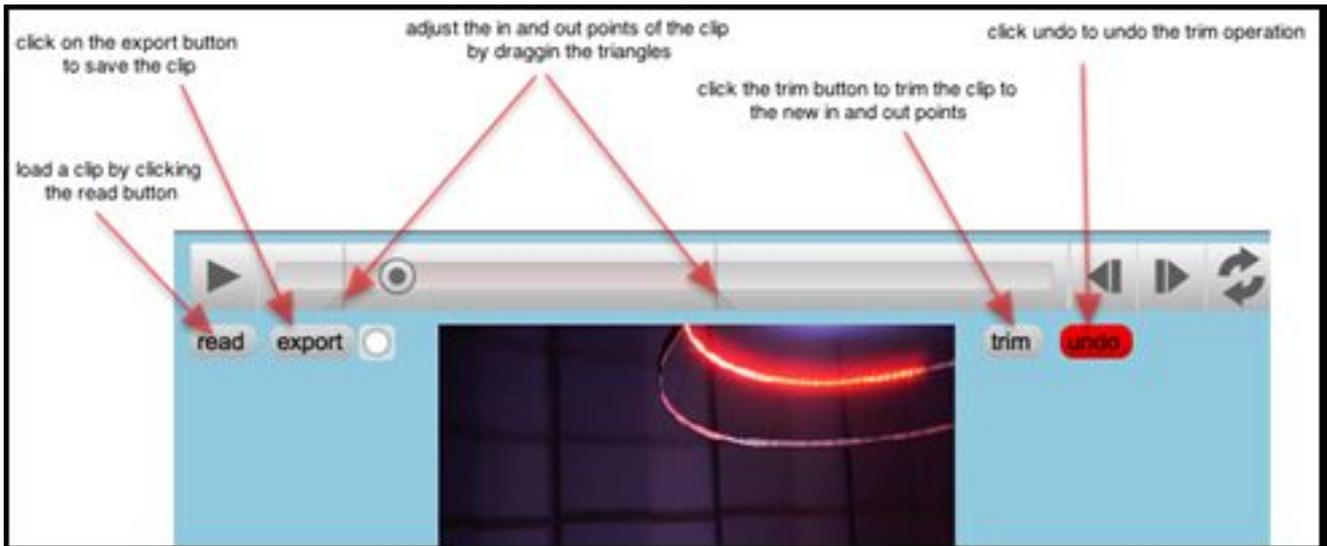
付録B： VPT trimmer

以下でダウンロードできます。

http://nervousvision.com/vpt/xtension/VPT_trimmer-mac.mxf.zip

http://nervousvision.com/vpt/xtension/VPT_trimmer-win.mxf.zip

VPT trimmer、次のことが容易に出来るようになります。ビデオを迅速に編集し新しいクリップとして保存、圧縮タイプの変更（例えば写真のJPEGやMP4またはH264の代わりにアップルのProRes）、クリップの解像度（小さなレイヤーのソースを使用するので、フル解像度のHDが不要なとき）



エクスポートボタンをクリックした際は、ビデオの名前と保存する場所を入力します。圧縮の種類や解像度を変更したい場合は、オプションボタンをクリックします。



設定やサイズをクリックすることで、エクスポートする前に必要に応じて調整することができます。エクスポートプロセスはクリップのサイズにより多少時間が掛かることがあります、我慢です。



付録 C : VPT 7 Video Trigger

VPT 7 video triggerは、Zach Poff's video triggerの改変です。(彼は、多くの非常に強力なソフトウェアを作ってきました、それらは彼のウェブサイトzachpoff.comから無料で入手できます)

VPTのイベントをトリガすることができるゾーンを、6ゾーンまで作成することができます。

以下でこのプログラムをダウンロードできます：

[VPT7 videotrigger mac](#) [VPT7 videotrigger win](#)

このソフトは、バックグラウンドのディテクションを使用して動作するので、まず空のバックグラウンドを記憶する必要があります。まずゾーンを作成して、そして、ゾーンのいずれかを選択し、それをビデオウィンドウ内にクリック&ドラッグします。ゾーンのON / OFFは、オン/オフボタンをクリックして切り替えます。

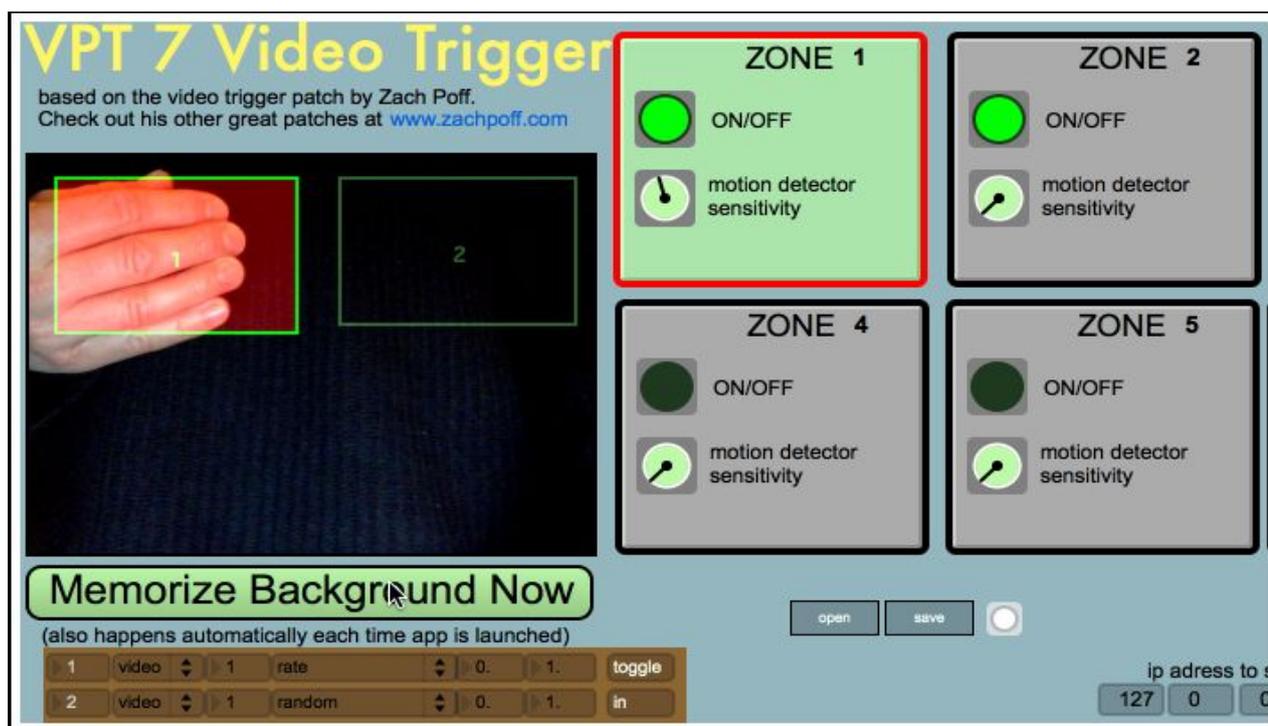
トリガーの下側部分はルータのセクションです、多少異なりますが、概ねVPTのルータと同様のものです：ここでのコントロール番号は、ゾーン (1-6) の番号になります。最後の列では、ルータは、トリガをどのように解釈すべきか("toggle", "in")を設定することができます。通常動作は、ゾーンにトリガをかけたときに1を取得、ゾーンから出たときに0を取得します。ゾーンに入るときにだけトリガーを動作したい場合は、トリガー解釈に "in" を設定する必要があります。

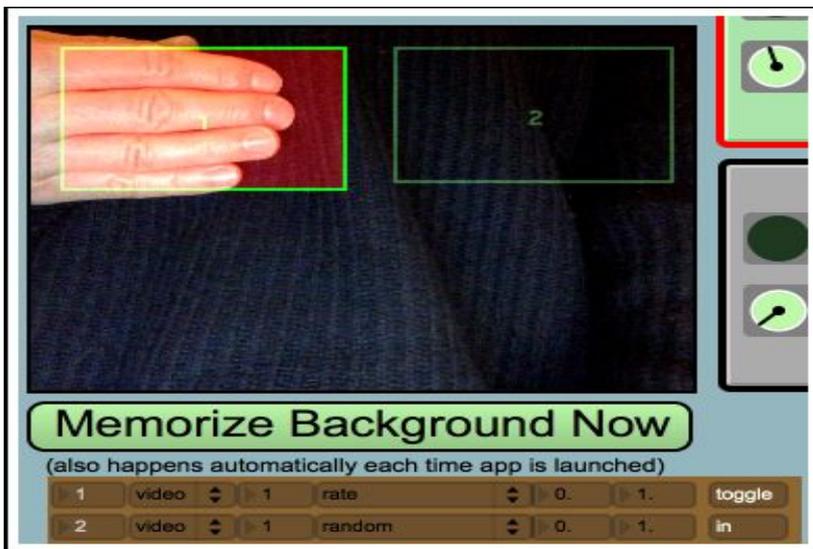
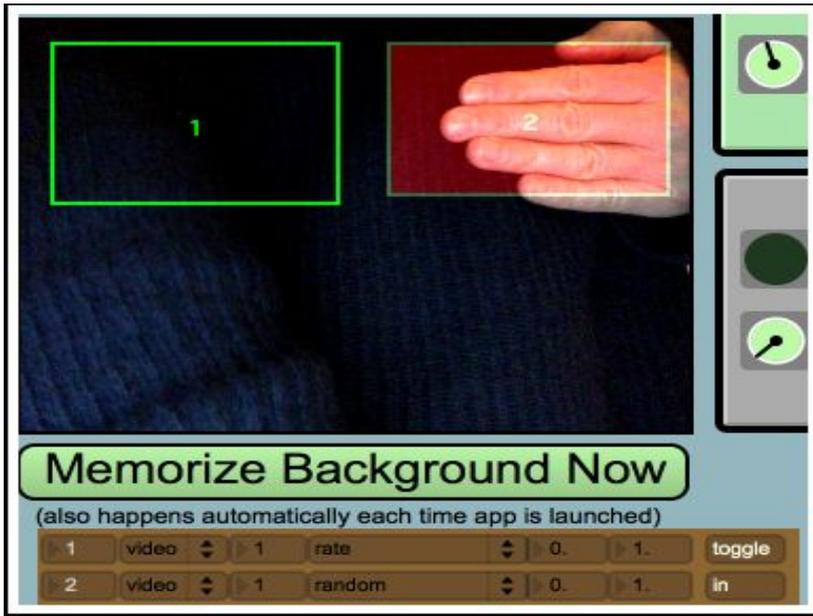
たとえば、もし、あるゾーンにビデオソースのランダム・クリップをトリガーしたい場合、"toggle"モードを使用すると、二重トリガになってしまうでしょう。したがって、ゾーンがトリガ活性化されたとき、およびゾーンが非活性化するときも、randomtriggerの状態になるでしょう。つまり、この場合は設定に "in" を使用すべき状況です。

ひとつのゾーンから完全に異なるイベントをトリガしたい場合は、ゾーンの活性化 (イン) を取り扱う1つのルータ行、およびゾーンの非活性化 (アウト) を取り扱う別のルータ行を持つことができます。

VPTと別のコンピュータの場合はIPアドレスの設定を忘れないでください。VPT受信ポート番号は6666です。

N. B. ! 「Nota bene !」 注意せよ！ ビデオトリガとVPTの両方で同じカメラを使用出来ません。



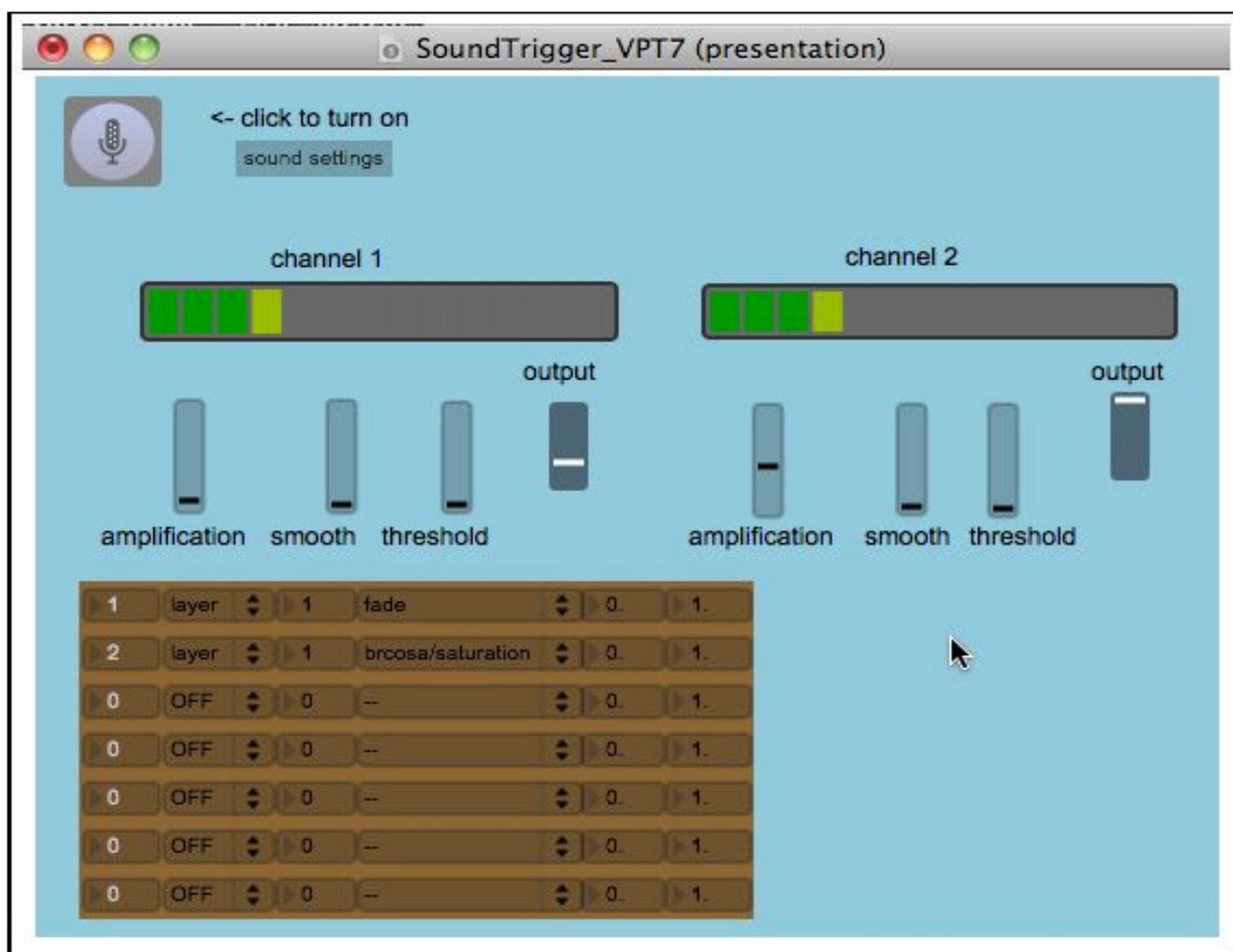


付録 D :SoundTrigger_VPT7

同じ様に、Soundtrigger VPTのアプリでは、VPTのパラメータを制御するために音入力を使用することができます。2つのコントローラ番号は、ステレオ入力の左右チャンネルです。

マイク記号をクリックして、サウンドをオンします。サウンド設定ボタンをクリックして、入力を設定します。

各チャンネルをモディファイするには、amplification (増幅)、smooth (滑らか)、threshold (しきい値) が使えることに注目してください。



Mavericksのトラブルシューティング

新しいMacのOS(mavericks)上では、VPTを開く際に問題があるかもしれません、なぜならアップルは独立の未登録の開発者（私のような）を信用していないためです。VPT上でCtrl+clickしてオープンを選択するだけで、この迷惑をバイパスできます。（初めて実行する時だけ必要です）

mavericksでは、複数モニタで作業を行う場合、デフォルトでは、各モニタにメニューバーを持つことになりました。メニューバーは現れて欲しくないため、VPTフルスクリーンモードでは問題です。システム環境設定に行き：ミッションコントロールで、”ディスプレイが個別のスペースを持つ”を無効にしてください。