

可視化事例の紹介と作り方 ～ AVS/Express チュートリアル ～

ADS第2 事業部
ビジュアリゼーション部
黒木
i-kuroki@cybernet.co.jp

サイバネットシステム株式会社



分野別の可視化事例ページ

<http://www.cybernet.co.jp/avs/example/category/>



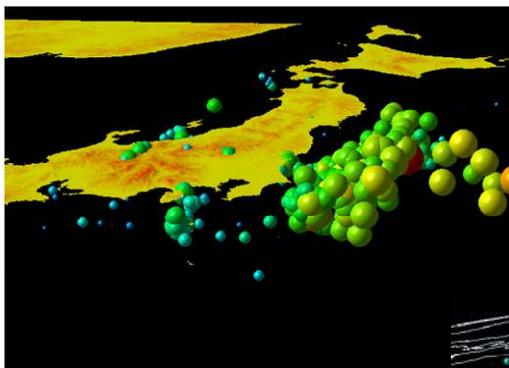
CYBERNET

本日の内容

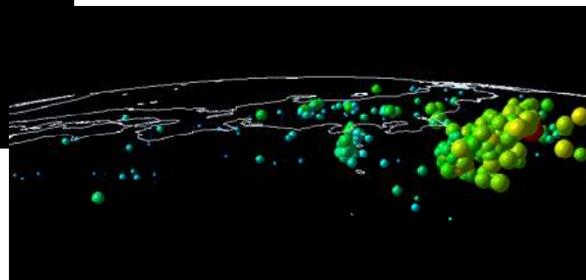
- 震源データの可視化
(緯度経度変換、離散点表示)
- 気象レーダーデータの可視化
(標高地図、複数等値面、ポリリュームレンダリング、…)
- 台風データの可視化
(ポリリュームレンダリング、流線 …)
- 津波データの可視化
(高さによる色づけ、高さによる半透明表示、…)
- カーボンナノチューブの可視化
(ボール&スティック、領域カット、フライスルー)

CYBERNET

震源データの可視化



- 離散点の球表示
- 日本地図の表示
(マップ変換)



データ出典：

防災科学技術研究所 Hi-net 高感度地震観測網

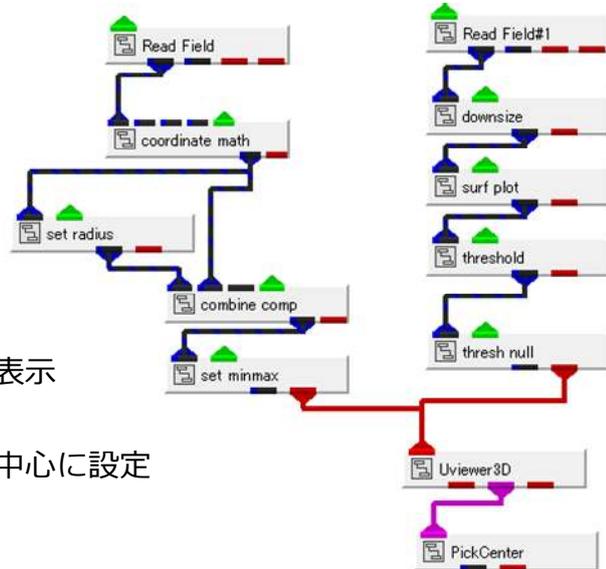
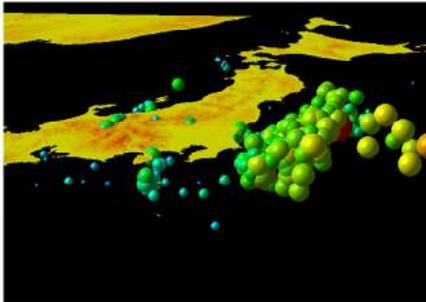
(http://www.hinet.bosai.go.jp/about_data/?LANG=ja)

可視化データ提供：

株式会社インターリスク総研 リスク計量評価チーム 川久保様

CYBERNET

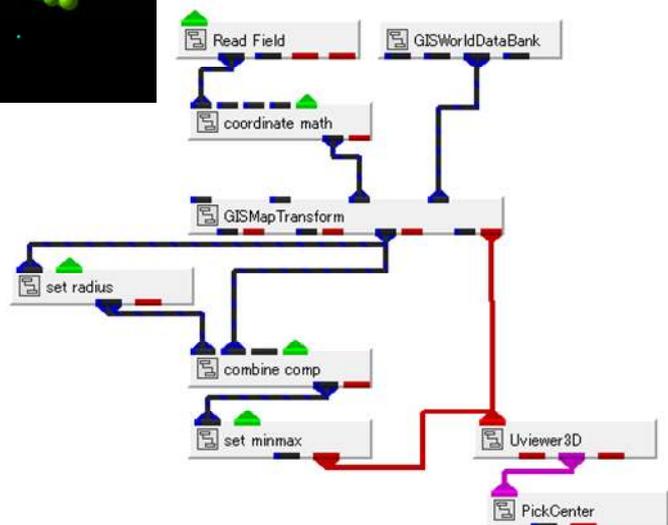
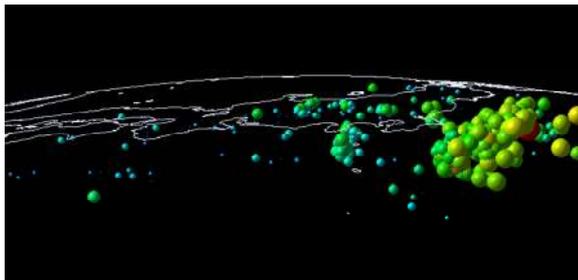
震源データの可視化（1）



- 離散点を球で表示
- 標高メッシュデータの凹凸表示
- 海域部分は非表示に設定
- ピックした位置を幾何変換中心に設定

CYBERNET

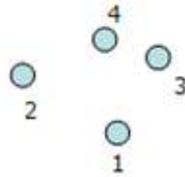
震源データの可視化（2）



- WorldDataBank II データ
- 地図座標変換

CYBERNET

補足)



AVS/Express で離散点（点のみ）を表現するには、Field データの以下の形式を利用することができます。

Field 1d 3-space irregular 型データ

例えば、左図の例では、4点のデータがあります。簡単なデータ例を以下に示します。

```

XYZ DATA1 DATA2  → ヘッダー行があることを想定
0.0 0.0 0.0 1.0 10.0  → このデータでは XYZ DATA1 DATA2 を想定
1.0 0.0 0.0 2.0 20.0
2.0 0.0 0.0 3.0 30.0
1.0 1.0 0.0 4.0 40.0
    
```

```

# AVS          → 1行目の #半角スペース AVS は必須
#              → 2行目以降の # はコメント
ndim = 1       → 1次元配列で表現できるデータ
dim1 = 4       → 配列の大きさ (点が4個)
nspace = 3     → 3次元空間 (xyz) に分布
veclen = 2     → データ成分は2個
data = float   → データのタイプは float 型 (double 型もあり)
field = irregular → 離散点の場合は任意の並びなので、irregular を指定
label = DATA1 DATA2 → データ成分2個の名前 (日本語不可、スペース区切り)
variable 1 file=/atom.dat filetype=ascii skip=1 offset=3 stride=5 → DATA1
variable 2 file=/atom.dat filetype=ascii skip=1 offset=4 stride=5 → DATA2
coord 1 file=/atom.dat filetype=ascii skip=1 offset=0 stride=5 → X座標
coord 2 file=/atom.dat filetype=ascii skip=1 offset=1 stride=5 → Y座標
coord 3 file=/atom.dat filetype=ascii skip=1 offset=2 stride=5 → Z座標
    
```

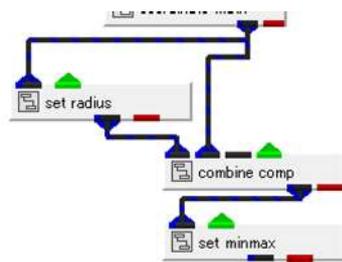
注)
ここに示す offset や stride を使った方法は、読み込みがもっとも“遅い”方法となります。
データ数が多くなる場合、成分毎のファイルに分割する、バイナリ化を行うなど、他の方法を検討してください。

CYBERNET

離散点データの可視化

・球で表示

set_radius モジュール

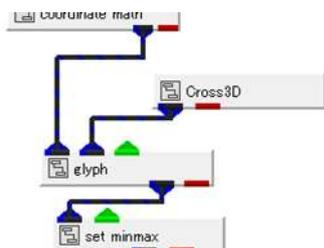


重要：マイナスで指定した場合、半径が同じ球を作成する。
注意：巨大な球が描かれないように、極小値から入力してください。

レンダラをソフトウェアに。(黄色をオフ)

・グリフ形状で表示

glyph + 形状モジュール (Sphere、Cross など)



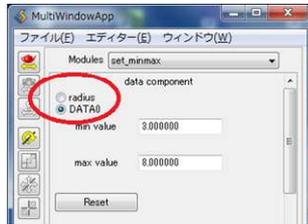
サイズの調整は、glyph モジュールのパラメーターで行います。(Normalize 一定の大きさ)

CYBERNET

離散点データの可視化（続き）

・ 数値と色の対応

set_minmax モジュールのパラメーターで、青(min)の値、赤(max)の値を指定します。



set_radius の後方につないだ場合



glyph の後方につないだ場合

注意：正しい対象を選択してください。

CYBERNET

日本地図の表示

・ 標高メッシュデータ

→ 以降の事例でも出てきます。
 今日では標高地図の作り方については触れません。
 以下のドキュメントをぜひダウンロードしてください。

地図データその他、気象を例に、各種データの可視化、
 動画の作成などについて記述しています。

http://www.cybernet.co.jp/avs/download/express_tech.html

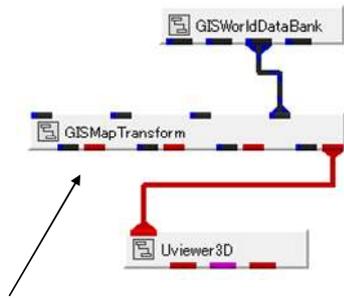
検索サイトで "AVS 気象 ハンドブック" で検索してみてください。



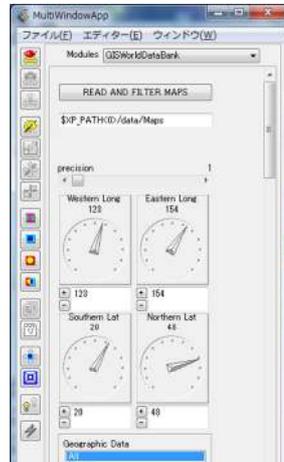
CYBERNET

- ・ CIA World Bank II データ (ライン)

AVS/Express に標準で含まれている外形線データです。
緯度経度で範囲を抽出します。



読み込んで緯度経度データを
画面に表示するために、
xyz 座標系に地図変換しています。

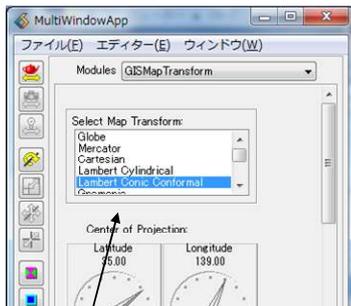


緯度経度を指定し、
READ AND FILTER MAPS ボタンをクリックします。

CYBERNET

緯度経度変換

GISMapTransform モジュール



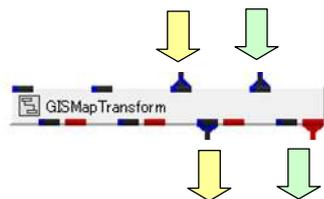
地図変換法を選択



Mercator



Globe



1 モジュールに 4 つまで、
データを入力できます。

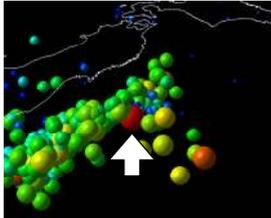
入力と同じ位置にある出力ポートから
変換後のデータが出力されます。
(どのポートを使っても同じです)

CYBERNET

ピックによる幾何変換中心の設定

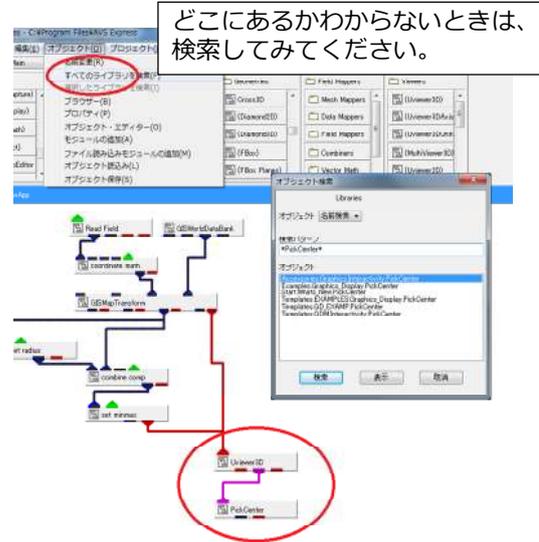
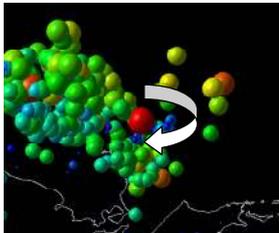
表示オブジェクトの中心ではなく、事象の中心など、マウス操作の中心を設定したい

→ PickCenter モジュールが利用できます。



Ctrl キーを押しながら、マウス左ボタンで、センターに設定したい部分をクリック

→ その位置を操作の中心に設定できます。



Accessories.Graphics.Interactivity.PickCenter

CYBERNET

以下のセッションにも、ぜひ、ご参加ください。

3 – 3 「確率論的津波リスク計量モデルの開発」

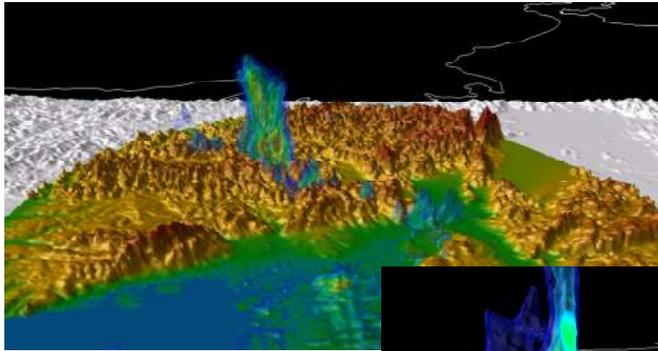
15:30 ~ 16:10

株式会社インターリスク総研

総合企画部 リスク計量評価チーム 川久保 様

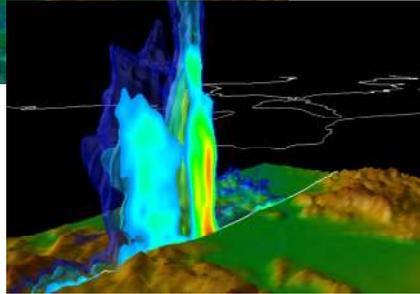
CYBERNET

気象レーダーデータの可視化



動画 1

- 標高地図データ
- 複数枚等値面
- 断面カットコンター
- アニメーター



動画 2

データ提供：

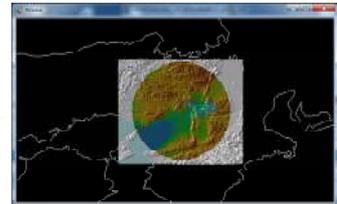
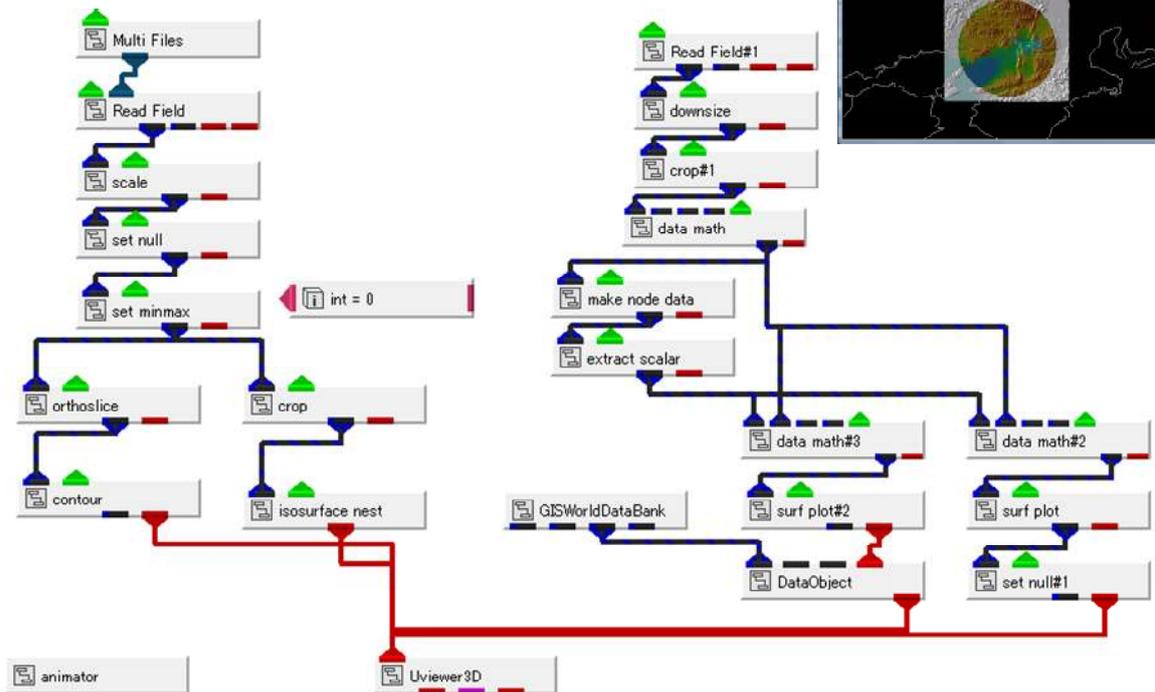
情報通信研究機構 電磁波計測研究所 佐藤様

フェーズドアレイ気象レーダーについて

<http://www.nict.go.jp/publication/NICT-News/1301/02.html>

CYBERNET

気象レーダーデータの可視化

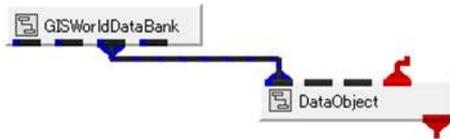


CYBERNET

日本地図の表示

・日本外形線の表示

前の例と同じ World Data Bank II のデータを利用しています。
 この例では、地図座標の変換は行っていません。
 緯度経度をそのまま、xy 座標としてプロットしています。



DataObject は数値データを形状データに変換するモジュールです。

読み込んだ外形線データから、ラインを作成しています。

CYBERNET

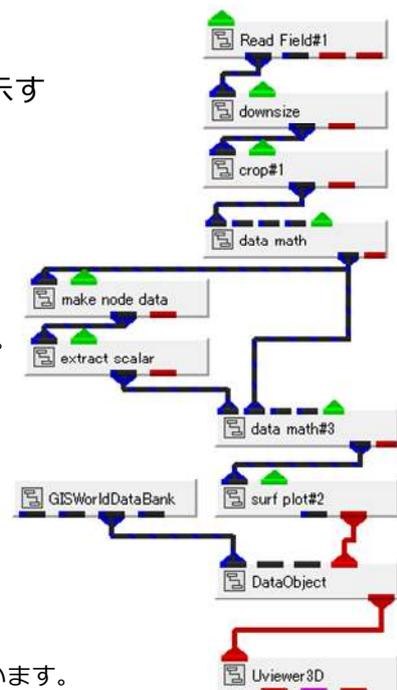
日本地図の表示

・標高メッシュデータ

先の例と同じですが、レーダーの範囲を示す仮想円の作成などを行っています。

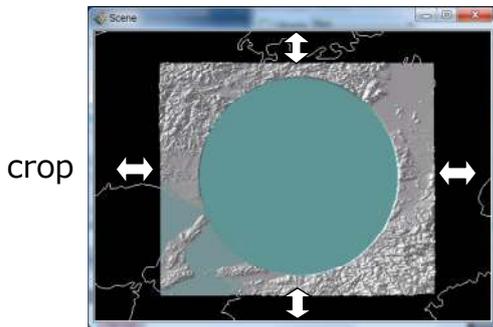
各モジュールの役割

- ・ Read_Field#1 モジュールで、標高データ（地図については、先の通り、別ドキュメントを参照）を読み込みます。
- ・ downsize は標高データの間引き（粗く）を行っています。
- ・ crop#1 モジュールは、領域抽出を行うモジュールです。標高データの範囲を抜き出すために利用しています。
- ・ data_math モジュールは、データ成分の計算を行うモジュールです。この例では、標高データが short 型のため、float 型に変換するために利用しています。
- ・ make_node_data は、座標値をノードのデータ成分に変換できるモジュールです。今回の例では、ある中心から同心円のデータを生成しています。
- ・ さらに、data_math#3 モジュールを使って、ある半径内にあるデータとその外側にあるデータとを区別しています。

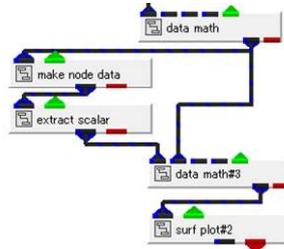


CYBERNET

日本地図の表示（続き）



前ページのネットワークで作成される地図



円の外側の標高データの抽出

surf_plot#2 は凹凸をつけるモジュールです。

この計算はデータ特有です。
-999 といった欠損値があるため、
データが -10 より大きい
部分のみを残しています。

レーダー中心部から、
距離データを作ります。

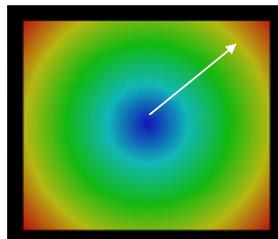
距離が 0.55 より大きい部分
を残し、その他の値は -10 に
しています。

CYBERNET

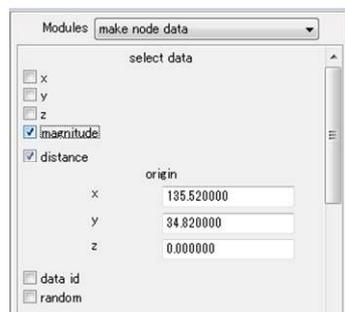
補足)

- make_node_data モジュール

座標情報からノードデータを生成します。



前ページの例で作成したデータ値。
ある中心からの距離をノードデータにしています。

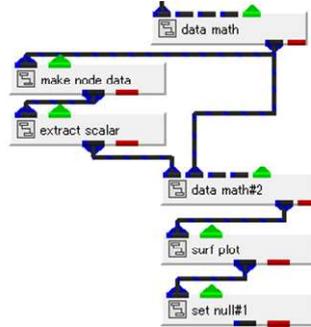
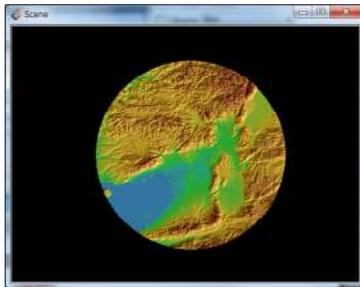


その他、x, y, z の各座標値をノードデータにすることもできます。

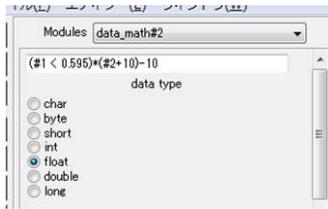
後半の津波データの可視化では、z 値をデータに変換し、その高さで色づけなどを行っています（後述）。

CYBERNET

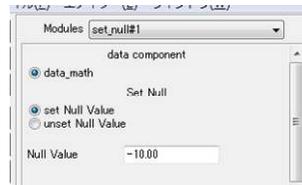
日本地図の表示（続き）



円の内側の標高データの抽出



距離が 0.595 より小さい部分を残し、その他の値は -10 にしています。



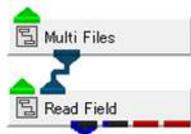
-10 の円の外側を非表示

CYBERNET

データの読み込み

- ・複数ファイルの連続読み込み

Multi_Files モジュール



あるフォルダの中のファイル名のリストを作成します。その名前をひとつずつ、リーダーモジュールに受け渡すことができます。



指定した拡張子のファイルリストが表示されるので、選んで追加します。

リストを選ぶと、その文字列が出力ポートにセットされます。

自動実行もできます。

ポートをつなぐには、中を開いて、現在の設定を消去します。ポートを出します。

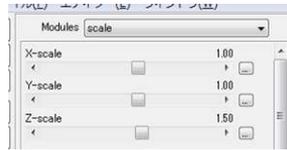
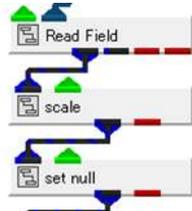
注) Linux/Mac 版では、現在、ファイルが順番に並ばないという不具合があります。

CYBERNET

データの読み込み

- 座標スケールの変更

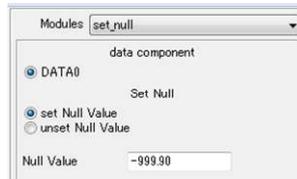
scale モジュール



scale モジュールは、高さ方向をデフォルメするために利用しています。

- null 値処理

set_null モジュール



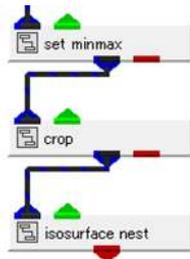
未観測値などを無効化（表示対象外）にしています。このデータでは、-999 の値となっています。

CYBERNET

可視化

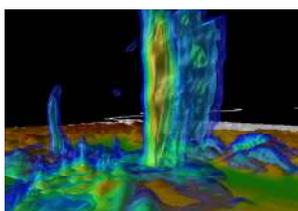
- 等値面表示

isosurface_nest モジュール

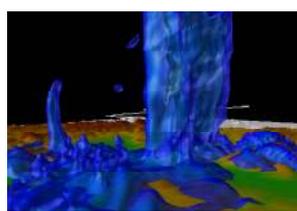


複数枚の等値面を作成します。最小、最大値間に指定した枚数の等値面を作成します。

補：set_minmax はカラーの調整に利用しています。crop については、後述します。



min = 60
max = 30



min = 30
max = 60

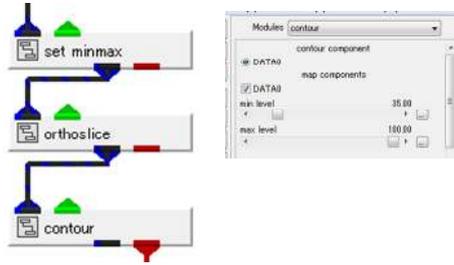
この例では、5枚の等値面を作成しています。

また、min に大きな値を、max に小さな値を設定しています。

CYBERNET

・断面コンター表示

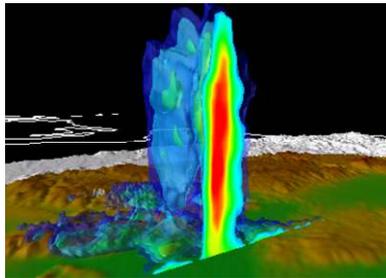
contour モジュール



orthoslice モジュールで指定した断面に、コンター図を作成します。

contour モジュールでは、表示の最小最大の範囲を指定できます。

このネットワークでは set_minmax モジュールで指定した値が引き継がれ、その範囲のみを表示しています。

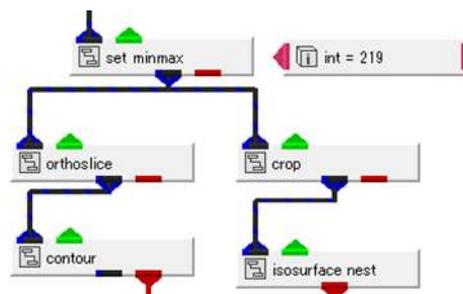


この例では、crop モジュールで等値面をカットしています。そのカットした場所にコンター図を作成しています。(次ページ参照)

CYBERNET

・等値面のカットと断面コンター表示

crop モジュールと orthoslice モジュールの連携



この例のように、あるパラメーターと別のパラメーターを連携させることができます。



① int オブジェクト

まず、int 型オブジェクトをインスタンス、ダブルクリックして、以下を入力します。

int => orthoslice.OrthoSliceParam.plane

→ 断面の位置を移動すると、この int の値が変わります。

② crop モジュールを開き（右上図）、min のパラメーターに以下を入力します。

min[3] => {int + 2, 0, 0}

→ int (足す2) の位置でクリップ（抽出）処理を行います。

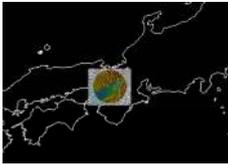
CYBERNET

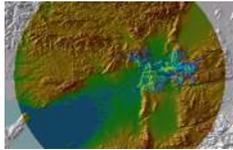
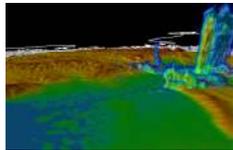
キーフレームアニメーション

- ・アニメーター

animator モジュールは、入出力ポートのない、単に、インスタンスだけして使うモジュールです。以下のように利用します。

animator モジュール


視点をセットしたら、+ キーをクリック

次の状態で、+ キーをクリック

次の状態で、+ キーをクリック

完成したら、再生ボタンをクリック

CYBERNET

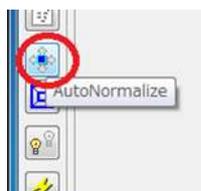
補足：アニメーション時の注意

- ・自動正規化（オートノーマライズ）機能

標準では、表示オブジェクトに変化が発生したら、表示画面内に収まるように、自動的に正規化される機能がオンになっています。

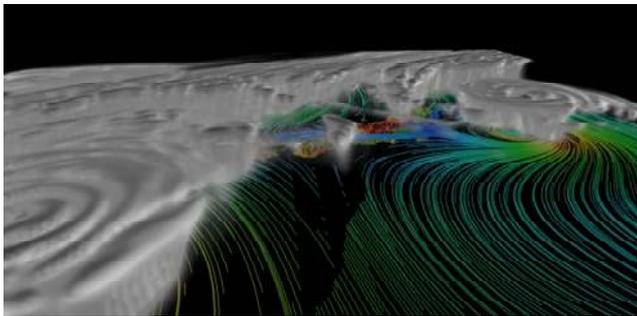
例えば、時系列データの読み込みで、高さや横方向の大きさなどが変わると、その幅を画面内に収めようとするため、カクカクとしたアニメーションになってしまいます。

この機能は以下のアイコンでオフにできます。



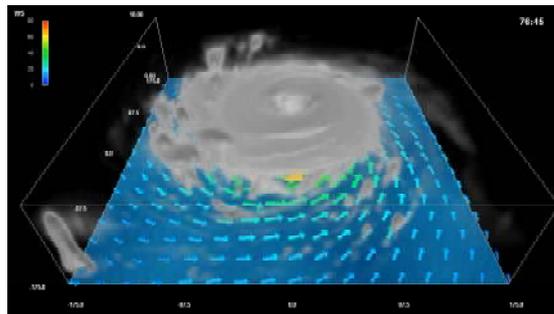
黄色の状態がオンです。
一度クリックすると、オフになります。
(再度クリックするとオンに戻ります)

台風データの可視化



動画 1

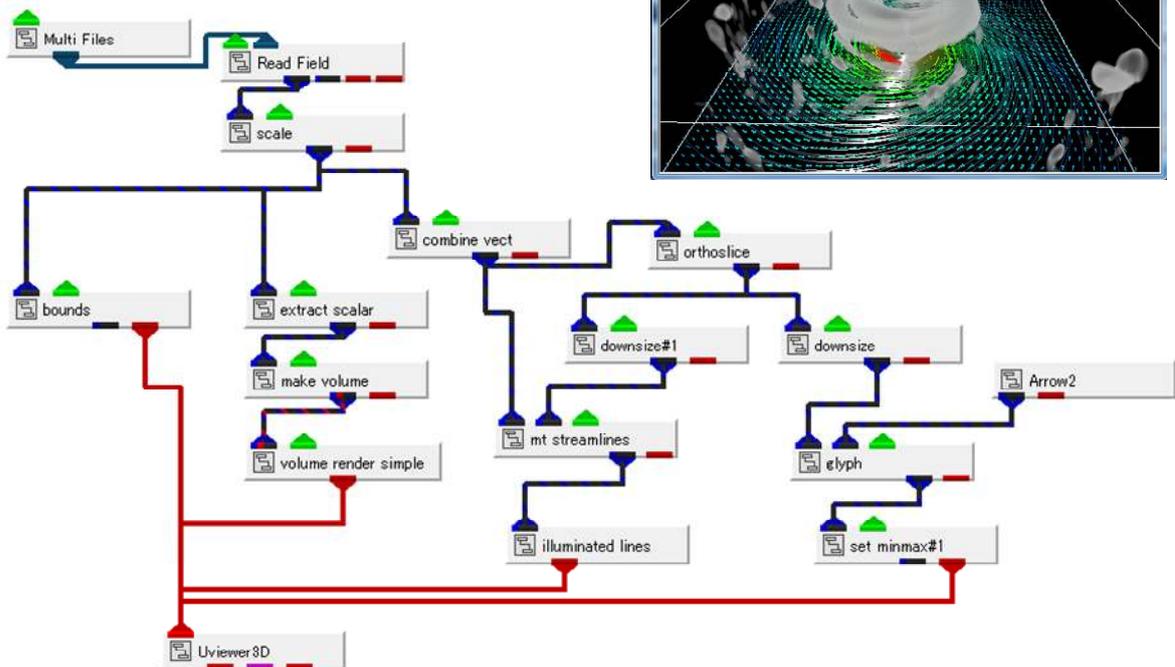
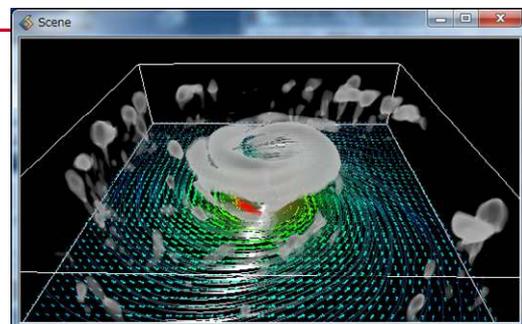
- ボリュームレンダリング
- 流線、流跡線
- ベクトル図
- フライスルー
- ローカル回転
- テクスチャ



動画 2

データ提供：
横浜国立大学 教育人間科学部 筆保様

台風データの可視化

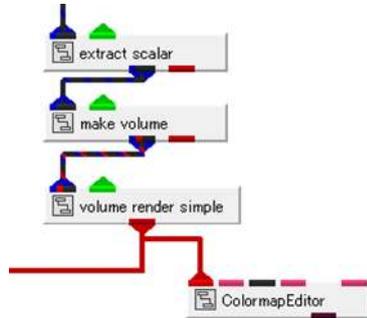


CYBERNET

可視化

- ・ボリュームレンダリング表示

volume_render_simple モジュール

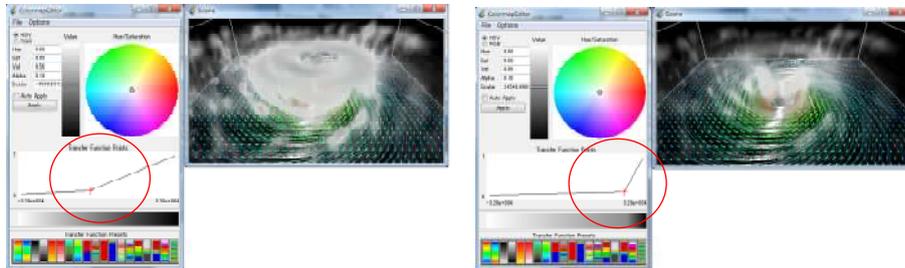


注) ボリュームレンダリングを行うモジュールには、volume_render と volume_render_simple の2つがあります。

この2つはどちらも同じです。色や透明度に関するパラメーターがあるかどうかの違いです。

volume_render_simple モジュールは、図のように、ColormapEditor モジュールと使います。

アルファ値をどのように定義するかを決めます。詳細は、先の気象のハンドブックや、AVS/Express のマニュアル等をご参照ください。

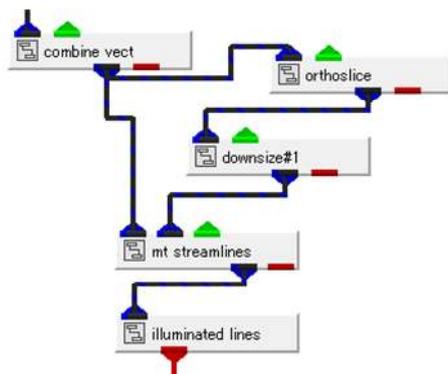


CYBERNET

- ・流線の表示とイルミネーションライン

streamlines (mt_streamlines) モジュール

illuminated_lines モジュール



流線やベクトル図を作成するには、3成分（2成分）のデータ・コンポーネントを作る必要があります。combine_vect モジュールで、ベクトル成分を選択しています。

streamlines モジュールは、流線を作成するモジュールです。どこから流線計算をスタートするか初期位置が必要です。この例では、ある断面で、且つ、ダウンサイズして間引いた格子点を初期位置としています。

illuminated_lines モジュールは、その流線に対して、装飾できるモジュールです。（OpenGL レンダラのみで有効）

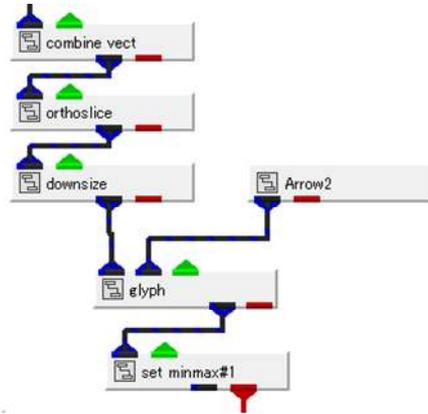
補足) mt_streamlines モジュールは、streamlines モジュールのマルチスレッド版です。流線の計算は、その初期位置1つずつ（1本ずつ）、計算が行われます。マルチスレッド版を使えば、一度に複数本（コア数分）の計算を行うことができます。

CYBERNET

・ベクトル図表示

glyph モジュール
Arrow2 モジュール

前ページ同様、まず、ベクトル・コンポーネントを選択しています。



ベクトル図は、先の事例で球やクロスを作成したモジュールと同じ glyph を利用します。

glyph モジュールは、ベクトル成分の場合、その方向と大きさを表すことができます。

この例では、Arrow2 モジュールを利用しています。その他の矢印形状 (Arrow1 ~ Arrow5) もあります。

表示位置は、格子断面で、且つ、その格子を間引いた位置としています。

CYBERNET

補足)

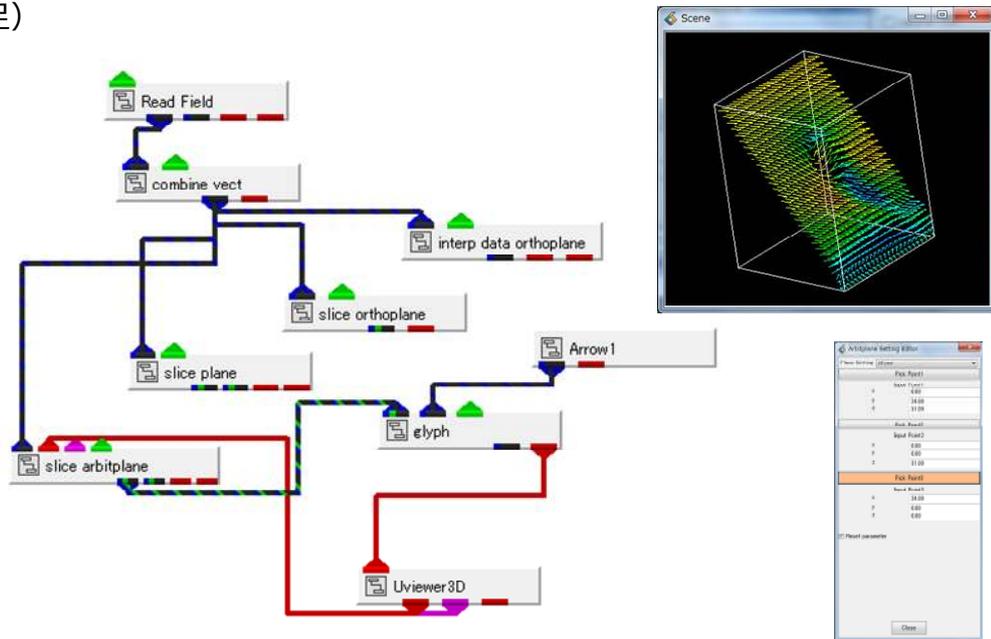
・断面ベクトル図

断面ベクトル図を描くには、その他、以下のモジュールを利用できます。

- ・ slice_orthoplane モジュール
x/y/z 断面を作成します。
- ・ interp_data_orthoplane モジュール
x/y/z 断面を作成します。
且つ、その断面を自分で指定したサイズに分割できます。
- ・ slice_plane モジュール
任意断面を作成します。
- ・ slice_arbitplane モジュール
任意断面を作成します。
3点ピックなど、対話的に位置決めできます。

CYBERNET

補足)



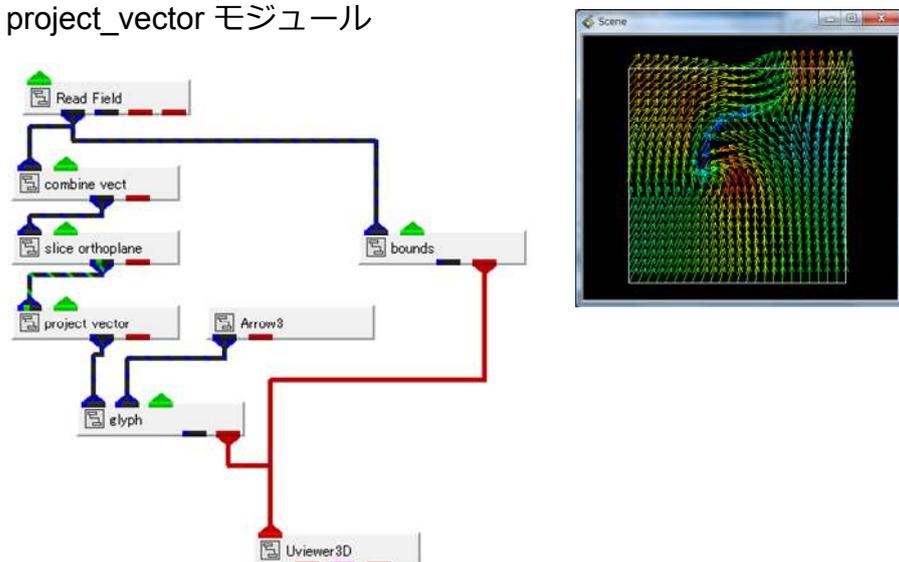
CYBERNET

補足)

- x/y/z 断面成分の抽出

以下のモジュールを利用すると、ベクトルデータに対して、x/y/z 断面に対する垂直、水平成分を抽出できます。

- project_vector モジュール

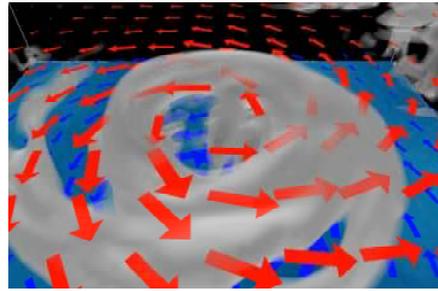


CYBERNET

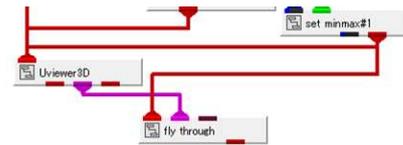
アニメーション

- ・ フライスルーアニメーション
fly_through モジュール

このモジュールの使い方は、以下のドキュメントで紹介しています。こちらをぜひダウンロードしてみてください。



動画



http://www.cybernet.co.jp/avs/download/express_tech.html

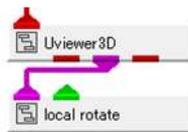


機能紹介にある「フライスルーモジュール」からPDF をダウンロードできます。

CYBERNET

- ・ ローカル座標回転アニメーション
local_rotate モジュール

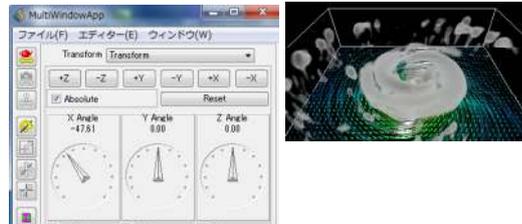
local_rotate モジュールは、そのオブジェクトの傾きを保ったまま、回転アニメーションを作ることができるモジュールです。



表示をリセット、ノーマライズした状態を作ります。



正しい軸を選び、一度、Save Current Position をクリックします。
その後 Run にチェックします。



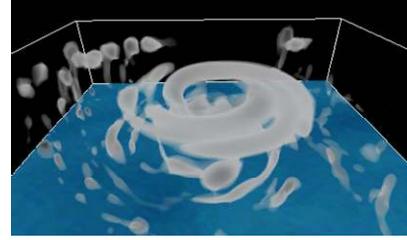
エディターメニューの Transform を使って、適当な角度に設定します。

CYBERNET

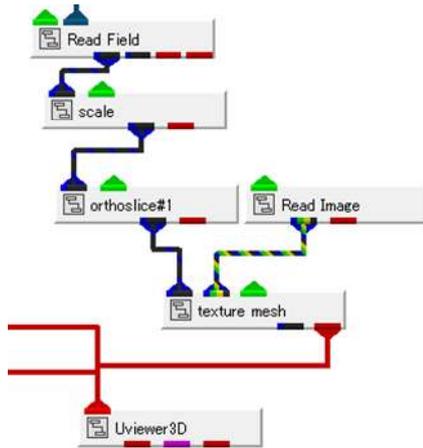
可視化の装飾

- ・テクスチャ画像の貼り付け

texture_mesh モジュール
Read_Image モジュール



この事例では、海をイメージした画像を結果の最下層に貼り付けています。
このように画像を貼り付けるには、texture_mesh モジュールを利用できます。

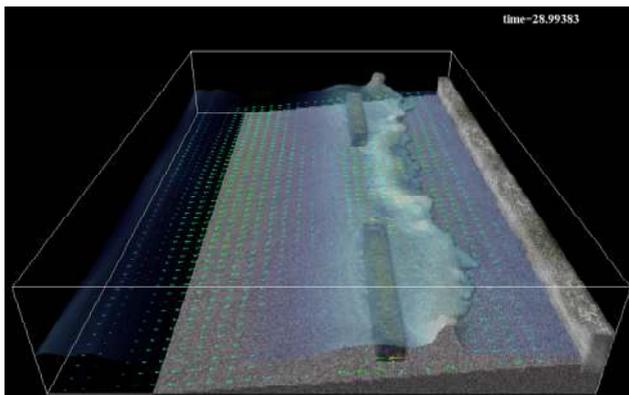


orthoslice#1 モジュールで、K 断面（高さ方向）の最下層を抜き出した面を作ります。

その断面に Read_Image モジュールで海の画像を読み込み、その板に貼り付けています。

CYBERNET

津波データの可視化



- 高さによる色づけ
- 高さによる半透明
- カラーマップ
- テクスチャ
- データ変換（HDF形式）

[動画 1](#)

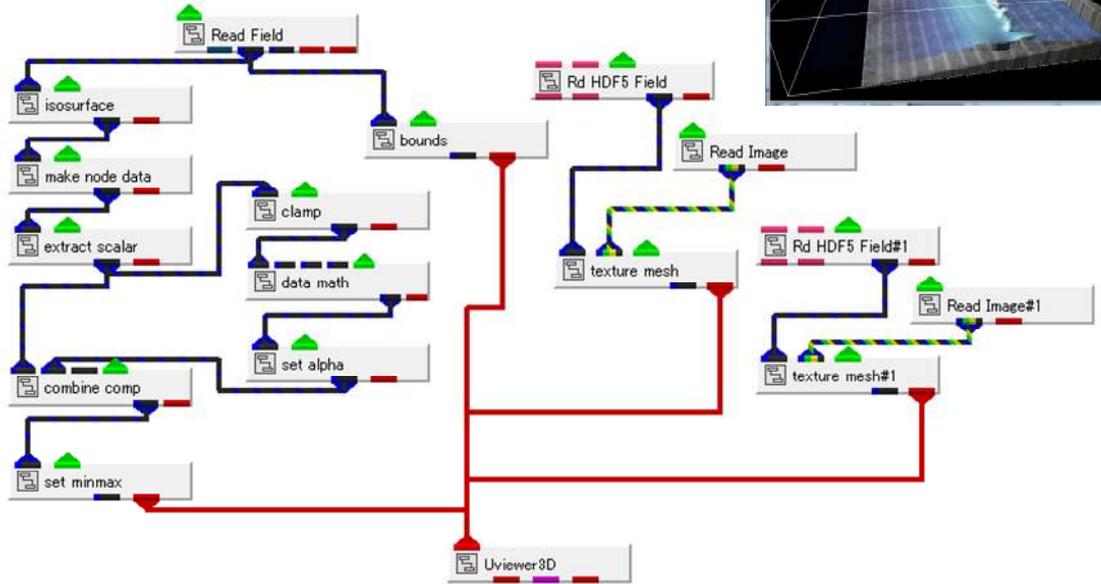
[動画 2](#)

データ提供：
富山大学 総合情報処理センター 奥村様

関連論文：
奥村 弘, 有川 太郎:
GPUによるCADMAS-SURF/3DのCUDA並列化と検討,
土木学会論文集B3（海洋開発）, Vol. 69 (2013) No. 2 p. I_754-I_759
http://dx.doi.org/10.2208/jscejoe.69.I_754

CYBERNET

津波データの可視化

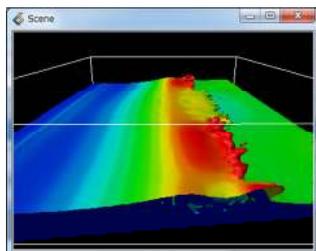
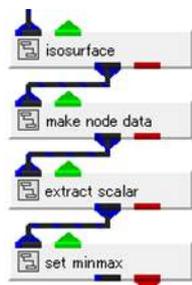


CYBERNET

可視化

- ・ 等値面と高さによる色づけ

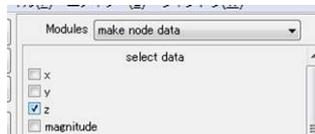
isurface モジュール
make_node_data モジュール



自由表面 (VOF) 流れシミュレーターによる計算で、その海面を、VOF の結果の等値面で表現しています。

make_node_data モジュールを使って、この水面の高さの情報をそのノードデータに持たせることができます。

※ make_node_data については、先のレーダーの可視化の事例でも利用しています。



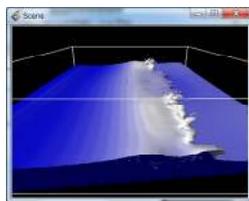
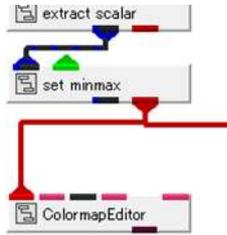
左図は、高さで色づけした例です。set_minmax モジュールで、適当な高さとの色の対応を設定しています。

CYBERNET

補足)

・ カラーマップ

ColormapEditor モジュール



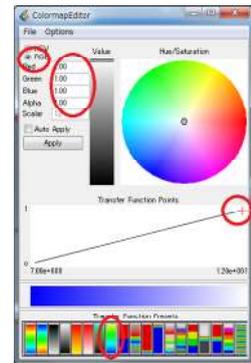
前ページと同じ表示を青から白のカラーマップに変更した例です。

AVS/Express のデフォルトカラーマップは、青から赤のグラデーションです。

このカラーマップを他の色分布に変更することもできます。前の事例のポリウムレンダリングで紹介した ColormapEditor モジュールを利用できます。

ColormapEditor モジュールを赤ポートに接続します。例えば、下図のように青から白に変更するには、

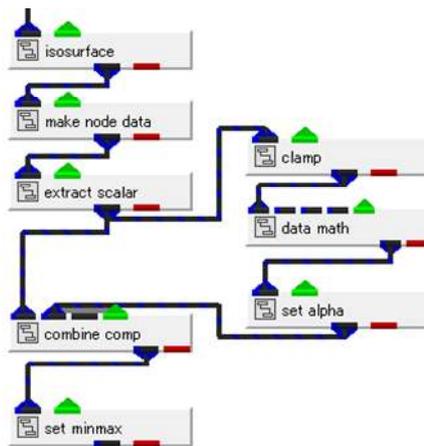
- 1) RGB を選択します。
- 2) 画面下のパレットから、一番近い、青から黄色のパレットを選びます。
- 3) カーブの一番右をマウス左でクリックし、選択します。
- 4) この点の色を RGB = (1,1,1) の白に設定します。



CYBERNET

・ 高さによる半透明表示

clamp モジュール
set_alpha モジュール

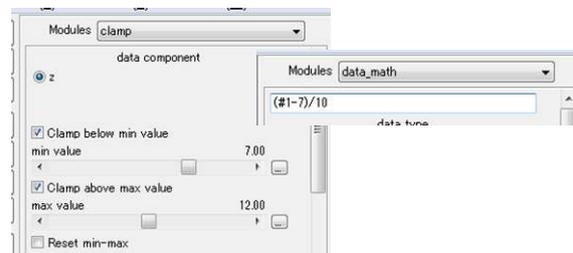


さらに、高さによって、透明度を変更しています。高さが高い部分は不透明で、高さが低い部分は透明で表現しています。

clamp モジュールは、指定した範囲外の値を、すべて、その指定した値に変換するモジュールです。

この例では、7 より小さい値はすべて 7 に、12 より大きい値はすべて 12 に変換し、7 ~ 12 の分布にしています。

さらに、data_math モジュールを使って、0 から 0.5 の範囲の値に変換しています。



set_alpha モジュールは、ノードデータの値をアルファ値（透明度）に変換するモジュールです。

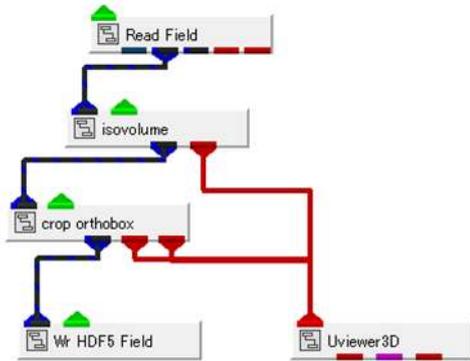
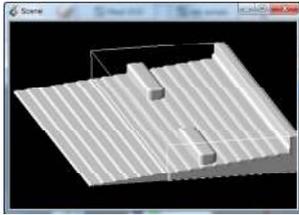
0 ~ 1 に正規化された値に対して、1 を不透明、0 を透明でレンダリングできます。（この例では、0 ~ 0.5 の値を作っています）

アルファ値だけでは色がつかないため、もともとのデータと combine_comp モジュールで、合成しています。

CYBERNET

・形状の抽出と HDF5 形式での保存

isovolume モジュール
 crop_orthobox モジュール
 Wr_HDF5_Field モジュール



このデータには、境界条件の形状を表すことができるデータが含まれています。

例えば、その境界を値から抜き出せる場合、isovolume モジュールなどを使って、形状を表現できます。

この事例では、さらに、crop_orthobox モジュールを使って、領域を抽出することで、防波堤を表現しています。

このように元データから何らか抜きだした静的なデータは、HDF5 形式で別ファイルに保存しておくことが便利です。

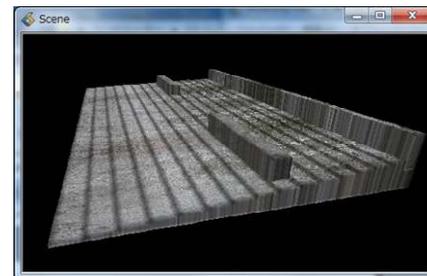
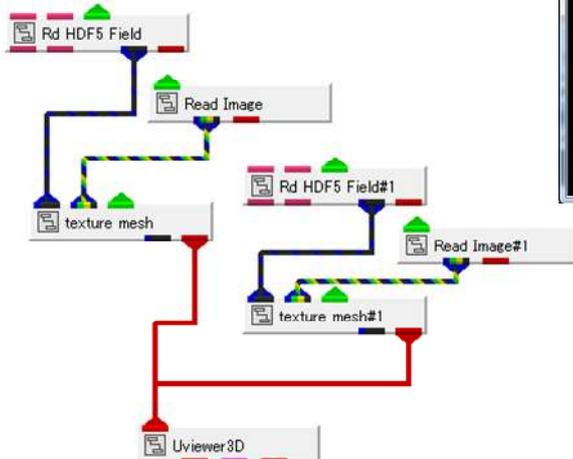
Wr_HDF5_Field モジュールで、ファイルとして保存、次ページのネットワークの例のように Rd_HDF5_Field モジュールで読み込むことができます。

※ その他、データによっては、速度成分の大きさがゼロの等値面（や等値ボリューム）で、形状を抜き出すことができる場合などもあります。

CYBERNET

・テクスチャ

Read_Image モジュール
 texture_mesh モジュール

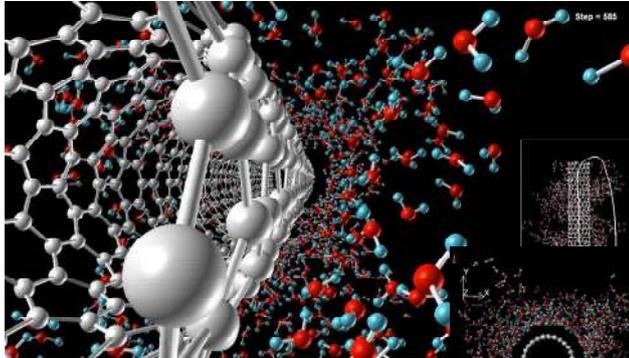


さらに、この例では、抜きだした形状に適切な画像を貼り付けて表現しています。

texture_mesh モジュールは、画像を貼り付けることができるモジュールです。

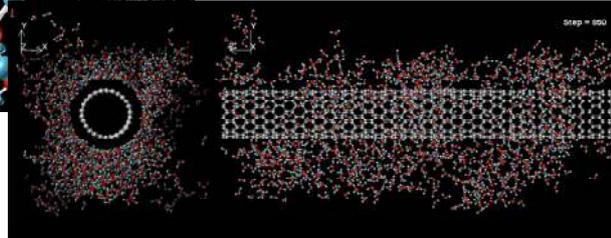
注) 複雑な貼り付けはできません。片方向から画像を投影しています。

カーボンナノチューブの可視化



- ボール&スティック
- 領域抽出
- フライスルー

動画 1

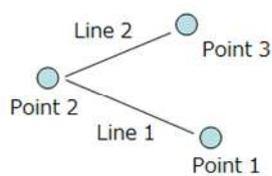


動画 2

データ提供：
東京理科大学 工学部 物性理論研究室 山本様

論文：
Y.Homma et.al. Phys.Rev.Lett. **110**, 157402 (2013)

ボール&スティック

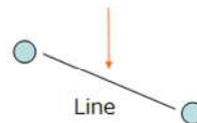
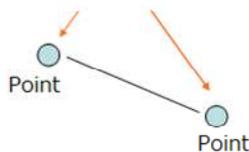


AVS/Express でボール&スティックを表現するには、UCD データを利用することができます。

例えば、左図のように、Point 要素 (Pt) と Line 要素とを組み合わせた UCD データを作成します。

ノードデータ：
粒子の ID や Type (種類)
粒子の直径
粒子の質量
粒子上で計算されたベクトル など

セルデータ
ボンドの ID や Type (種類)
ボンドの直径
ボンドの長さ
ボンド上で計算された値 など



この Point の定義点であるノードにノードデータを持たせたり、Line の要素中心 (Line そのもの) にセルデータを持たせることができます。

ご自身のボール&スティックデータがある場合、UCD データフォーマットへのデータ変換を検討してみてください。

CYBERNET

補足)

・材料系データの読み込みモジュール

その他、以下のデータの読み込みモジュールがあります。

- ・ PDB
KGT.Data_IO.Read_PDB モジュール
- ・ Gaussian
CCMS.Gaussian.Read_Gaussian_Log_Ang
CCMS.Common.Read_XYZ
CCMS.Gaussian.Gaussian_Vibration
CCMS.Gaussian.Read_Gaussian_Cube_Ang
KGT.Data_IO.Read_Gaussian_Log
KGT.Data_IO.Read_Gaussian_Cube
- ・ VASP
CCMS.VASP.Read_VASP
CCMS.VASP.Read_VASP_MD
- ・ XYZ
CCMS.Common.Read_XYZ
- ・ ATK netCDF
CCMS.TOMBO.TOMBO_Cube

CYBERNET

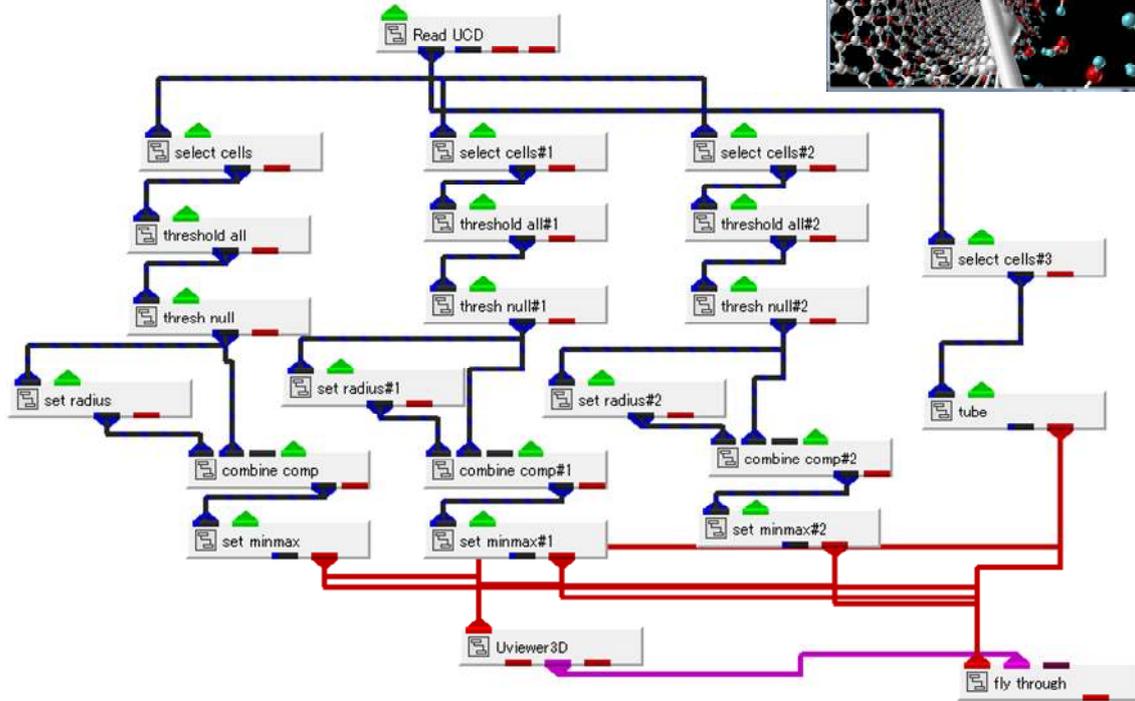
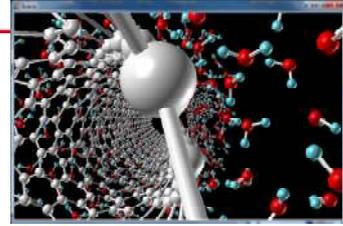
・材料系データの読み込みモジュール（続き）

- ・ TOMBO
CCMS.Common.Read_XYZ
CCMS.TOMBO.TOMBO_Cube
CCMS.TOMBO.TOMBO_MD
- ・ Octa/COGNAC *1)
- ・ Lammps *1)

※ *1) については、お問い合わせください。

CYBERNET

カーボンナノチューブの可視化



CYBERNET

可視化

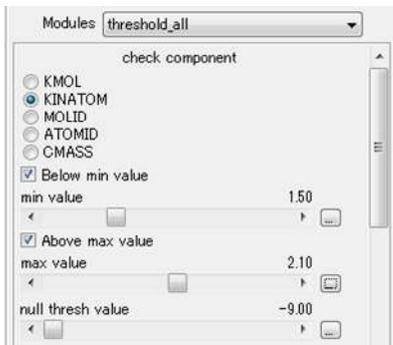
- ・セルの選択と値によるデータ抽出

select_cells モジュール
 threshold_all モジュール
 thresh_null モジュール

select_cells モジュールは、UCD の要素タイプ毎にデータを抽出できるモジュールです。

この例では、Point/Line が選択でき、ボールを描くためには Point を、チューブを描くためには、Line を選択しています。

(前ページの例では、ボールが3種、チューブが1種の表示のネットワークがあります。)



threshold_all モジュールは、ある設定した範囲外の値を null 値に変換するモジュールです。

この例では、KINATOM という変数に ATOM の種別を示す ID がデータ値に含まれています。

この ID が 1.5 より大きく、2 より大きい値に Null 値を、すなわち、ID = 2 のみを抽出しています。

同様に、前ページのネットワークでは、3種のボールを表示するモジュールのラインがありますが、ID = 1/2/3 の3種をそれぞれ抽出、表示しています。

thresh_null モジュールは、Null 値の要素を非表示ではなく、データとして削除してしまうモジュールです。

CYBERNET

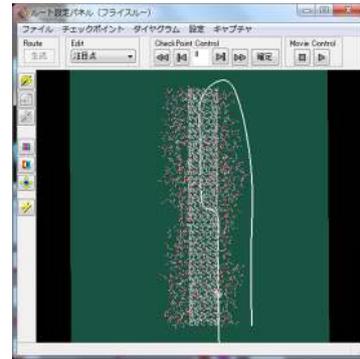
アニメーション

- ・ フライスルーアニメーション

fly_through モジュール

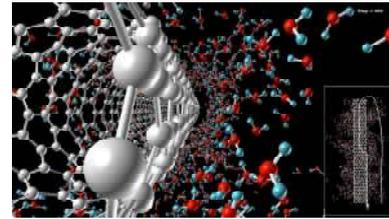
台風の事例で紹介したフライスルーのドキュメントをご参照ください。

軌跡を作成し、それに沿ったカメラ・アニメーションを作成できます。



なお、この事例では、右図のように、カメラ視点と、軌跡の位置表示の2つの画像を表示しています。

このような動画を一度に、AVS/Express 上で作ることはできません。フライスルーのパネルから、その軌跡画面をキャプチャして、動画に出力することができます。その軌跡動画とフライスルーのアニメーション動画をそれぞれ作成し、連番画像として出力します。その後、右図のように配置した後、新しい動画を作成しています。



CYBERNET

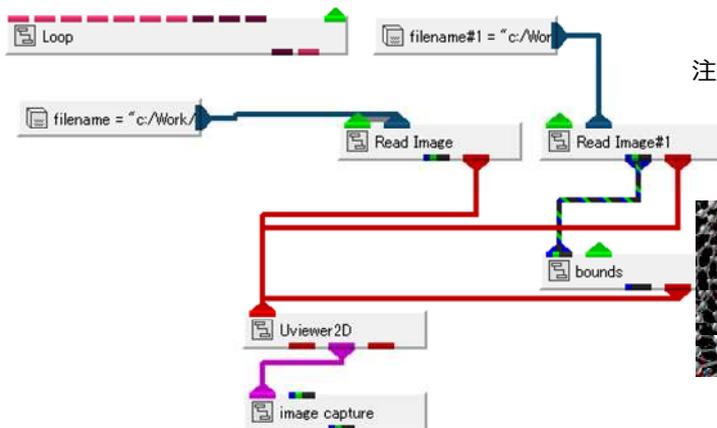
補足)

- ・ Read_Image と動画作成

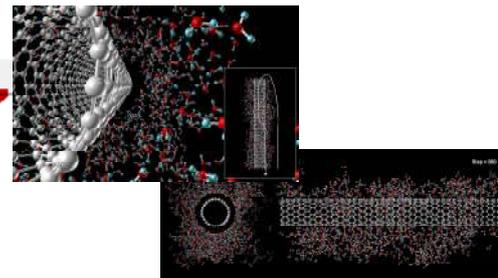
Read_Image モジュール
Uviewer2D モジュール

前ページのカメラ視点と軌跡を並べた動画、また、最初の事例の正面、側面の2つを並べた動画は、AVS/Express 上で作成することもできます。

それぞれの動画素材を連番画像で作成します。それらを Read_Image モジュールで読み込み、Uviewer2D モジュールでレイアウトします。そのレイアウトした連番画像の読み替えによるアニメーションを再度動画として保存します。



注) Uviewer2D のビューサイズを作成している連番画像にあわせて、画像を読み込んだらリセットし、同じ解像度で表示されるようにします。



連番ファイルを Loop で読み込む設定例：

```
filename => str_format("c:/Work/img/img01_%05d.bmp",Loop.count)
```

CYBERNET

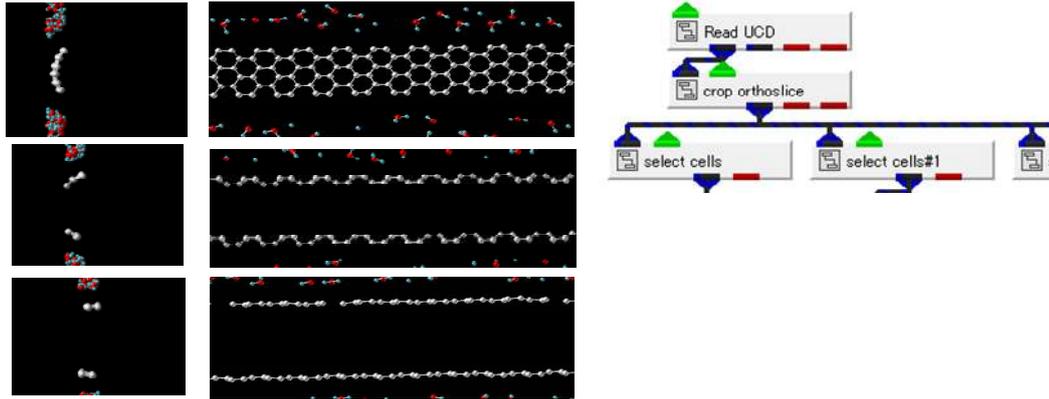
補足)

・ 空間抽出

- crop_sphere モジュール
- crop_cylinder モジュール
- crop_orthoslice モジュール
- crop_orthobox モジュール

ボール&スティック表示では、データ数が多くなると、その現象がわかりにくい場合があります。

左記のモジュールを使うと、空間的なデータ抽出を行うことができます。



crop_orthoslice モジュールでスライスした例

CYBERNET

ご利用上の注意：

本書中の解説、及び、図、表は文書による許可なしに、その全体または一部を無断で使用、複製することはできません。

サイバネットシステムでは記載内容に関して正確であることに努めていますが、本書の利用に関して生じた損害については法律上のいかなる責任も負いません。

AVS/Express は米国Advanced Visual Systems 社の商標です。
上記以外の製品名も一般に開発各社の商標、あるいは登録商標です。

なお、本書は、各事例をご提供いただいた皆様のご厚意により、利用の許可をいただき、サイバネットシステムにて、Viz カンファレンスのチュートリアル用書き起こしたものです。本書内の記載内容について、各事例をご提供いただいた皆様には、一切関係ありません。お問い合わせ等については、サイバネットシステムにお願いいたします。

また、各事例をご提供いただいた皆様に感謝いたします。

サイバネットシステム株式会社
ADS第2事業部
ビジュアライゼーション部