2021/02/19 第4回ビジュアリゼーションワークショップ



#### Unity用可視化フレームワークVisAssetを用いた 可視化アプリケーション開発

#### 〇川原慎太郎(海洋研究開発機構) 宮地英生(東京都市大学)



- 本日御用意頂くもの
- ・ <u>VisAssetsの概要</u>
- <u>VisAssetsのダウンロード</u>
- <u>新規プロジェクトの作成</u>
- <u>Unityエディタの画面構成</u>
- <u>VisAssetsのインポート</u>
- ・ 実行に必要なアセットの追加
- <u>可視化モジュールの場所</u>
- <u>データを読み込む</u>
- <u>ここまでの状態の確認</u>
- <u>カラースライスを表示する</u>
- <u>任意の要素のカラースライスを表示する</u>
- 複数のカラースライスを表示する
- <u>等値面を表示する</u>

- <u>ユーザインタフェースの追加</u>
- <u>ビルド前の準備</u>
- Windows用実行ファイル作成
- [参考] PC-HMDでの立体視表示
- [参考] Android用ビルド
- [参考] Cardboard対応ビルド
- ベクトル場の可視化
- <u>新たなモジュールの開発</u>
- ・ <u>まとめ</u>
- <u>謝辞</u>
- [付録] VFIVE用データの読み込み
- [付録] GrADS用データの読み込み

# 本日御用意頂くもの

※本資料はWSのページからダウンロードできます

- UnityをインストールしたWindowsPC (Windows7 SP1+以上, 64bit)
   ※ Macでも動くかとは思いますが、十分な動作確認をしていません
- Unityのバージョンは2019.4 LTS を使用します
   ※ Unity Hubからのインストールをおすすめします
- ハンズオン後半で使用するサンプルデータ(下記URLから取得してください) <u>https://www.jamstec.go.jp/ceist/aeird/avcrg/vfive.ja.html</u>
   ダウンロードの項にあるサンプルデータABC Flowのビッグエンディアン版(sample\_big2.tar.gz)を使用します
- Android 用ビルドをする場合 (時間が足りない場合は紹介のみとなります)
   Android用ビルド環境 (Unityのインストール時に選択可)
   Android機 (スマートフォン、タブレット等)
   上記とPCを接続するデータ通信可能なUSBケーブル (充電専用のものはNG)

#### VisAssetsの概要(1)

- ・Unity用可視化フレームワーク
- ・ 宮地英生教授(東京都市大学)と川原との共同開発
- AVS/Expressのような可視化ネットワークの構築をUnity上でも簡単に、
   をコンセプトに開発中



AVS/Expressのビジュアルエディタ



Unityのシーンヒエラルキー上での VisAssetsによる可視化ネットワーク構築

#### VisAssetsの概要(2)

- 可視化フローを構成する小要素を機能毎にモジュールアイコン化
- Unityのシーンツリーへのモジュールアイコンのドラックアンドドロップによる プログラミングレスでの可視化アプリケーション開発を実現



Modularized visualization elements

Visualization network on Unity

#### VisAssetsの概要(3)

・ サンプルモジュールによるスカラー場、ベクトル場の可視化例



#### VisAssetsの概要(4)

・ ビルド後のアプリケーションでのパラメータ変更用GUIを提供



#### VisAssetsの概要(5)

 マルチプラットフォーム対応であるUnityを基盤とすることにより、PC、スマートフォン等 多くのデバイスで実行可能なアプリケーションの開発ができます





Androidスマートフォンでの実行

HMDへの表示

# VisAssetsのダウンロード(1)

<u>https://github.com/kawaharas/VisAssets</u> にアクセスします



# VisAssetsのダウンロード(2)

下にスクロールし、Releasesの「1 tags」の部分をクリックします

	C	ᅌ https://	github.com/kawaharas/Vis	Assets		аљ	Co 🗣 C≞	Ē	2	
۴r	main 👻	<b>양 1</b> branch	🟷 <b>1</b> tag	Go to file	Add file 🔻	⊻ Code -	About		ŝ	
	kawahara	s update LICENS	E	7d615dc	15 hours ago	🕑 44 commits	Visualization Asset	ts for Ur	nity	
	Assets		delete streamlines			19 hours ago	শ্রু View license			
	Packages		fixed ReadV5 and ReadM	oduleTemplate		8 days ago				
	ProjectSet	tings	update .gitignore			20 hours ago	Releases		5	
ß	.gitattribu	tes	Initial commit			2 months ago	🛇 1 tags			
ß	.gitignore		Update .gitignore			19 hours ago	Create a new release		1	
Ľ	LICENSE		update LICENSE			15 hours ago			-	
Ľ	README-	J.md	update README-J.md			15 hours ago	Packages			
Ľ	README.r	nd	update README.md			15 hours ago	No packages published Publish your first packag	je		
REA	DME.md					Ø	Languages			
VisAssets						<ul><li>C# 97.5%</li><li>ShaderLab 2.5%</li></ul>				
v It th	enable u ne hierarc	s a visualization s to construct hy window of	visualization application by Unity.	connecting vis	sualization mo	odules on				
L	low to	11502								

# VisAssetsのダウンロード(3)

・「Assets」をクリックして展開し、unitypackage をダウンロードします



# 新規プロジェクトの作成(1)

 Unity Hubを起動し、新規作成ボタンを押します (複数のバージョンのUnityをインストールしている場合は、 新規作成ボタン右端の下矢印ボタンを押して2019.4.XXを選択します)

Unity Hub 2.4.2				_	
🚭 unity				x	з ѕк
プロジェクト	プロジェクト		リスト	に追加新規作品	t –
◆ 使い方を学ぶ	プロジェクト名	Unity パージョン	ターゲット	最終更新 个	۹
👪 コミュニティ					- 1
■ インストール					

# 新規プロジェクトの作成(2)

 テンプレートで「3D」を選択、プロジェクト名と保存先を設定し「作成」を押します (ここではプロジェクト名をデフォルトの「New Unity Project」から「Visualization Workshop」に変更しています)

Unity Hub 2.4.2						×
Unity 2019.4.18f1 で新しい	プロジェクトを作成			_		$\times$
テンプレート			設定 ブロジェクト名*			
(i)	(i)	(i)	Visualization Workshop			
⊞	•	\$	保存先。"			
2D	3D	3D With Extras	C:\Users\kawahara\Documents\Ur	nity Projects	\$ •••	
(i)	(i)	<u>+</u>				
=	•	<u></u>				
High Definition RP	Universal Render	2D Platformer				
	Fibeine	Microganie				
			キャンセル	1	作成	

# Unityエディタの起動

• 作成した新規プロジェクトでUnityエディタが起動します

Visualization Workshop - SampleScene - PC, Mac & File Edit Assets GameObject Component Window	& Linux Standalone - Unity 201 w Help	9.4.18f1 Personal <dx11></dx11>				- (	5 >	×
O II II ⊕ X ØCenter	r 😭Local 📇		► II ►		Collab 🔻 📤 Account 🔻 L	avers 🔻 L	ayout	-
E Hierarchy – – – – – – – – – – – – – – – – – – –	# Scene 👁 Game 🔮	Asset Store			i 🕒 Inspector		a	3 :
E Hierarchy d: +	# Scene ∞Game Display 1 ▼ Free Aspect	Asset Store	1x	Maximize On Play Mute Audio Stats Gizr	e Inspector		-	5 :
Project 🖶 Console					- F			
Clear Collapse Clear on Play Clear on Build Err	ror Pause Editor 🔻				0 🙂 0			
					Aut	o Generate Ligh	nting Off	

## Unityエディタの画面構成(1)



# Unityエディタの画面構成(2)

例えば、ヒエラルキーウィンドウで「Main Camera」を選択すると、インスペクタウィンドウに「Main Camera」を構成するコンポーネントとそのパラメータが表示されます

※前のスライドの画面から、ゲームビュー⇒シーンビュー、コンソールウィンドウ⇒プロジェクトウィンドウへと 表示を切り替えています

Visualization Workshop - Sam	pleScene - PC, Mad	& Linux Stan	dalone - Unity	2019.4.18f1 Personal	<dx11></dx11>							- 0	×
ile Edit Assets GameObject	Component Wind	ow Help			and the second								
o 💠 🗘 🗹 🖽 🍭	🖇 🔀 Cent								S Collai	o 🕶 📤 Account		▼ Layo	
		# Scene	😎 Game	🗎 Asset Store						Inspector			
▼ Q: All		Shaded	▼ 2D	• • ≉ ∞(	0 崁 👻		 🛠 💵 🕶 Gizmos 👻 🤇	er All		🕥 🗸 Main Cam	era		
Main Camera										Tag MainCam	era 👻 La	yer Default	
💮 Directional Light		-		_	-	_		_	~~~	7 🧏 Transform		(	0 7 i
					-				x	Position	X O	Y 1 Z	-10
										Rotation	x o	Y O Z	0
										Scale			
		1								🔻 🗖 🖌 Camera			0 ⊉ :
										Clear Flags	Skybox		
										Background			8
										Culling Mask	Everything		
										Projection	Perspectiv		
										FOV Axis	Vertical		
										Field of View			
										Physical Camera			
										Clipping Planes	Near 0.3		
											Far 100		
		1								Viewport Rect			
			_				 	_	: E	X O	Y (	)	
									♣ ♥ ★ Ø8	WI	н		
Favorites	Assets									Depth	-1		
Q All Materials										Rendering Path	Use Graph	ics Settings	•
Q, All Prefabs										Occlusion Culling	None (Ren	der Texture)	e
										HDR	Use Graph	ics Settings	
Scenes										MSAA	Use Graph	ics Settings	
Packages													
										Target Display	Display 1		
										Target Eye	Both		
										n 🗸 Audio Liste	ner	(	071
											dd Componer		
											Auto Gene	rate Lighting	g Off

# VisAssetsのインポート(1)

 メニューの[Assets]、またはプロジェクトウィンドウ内のAssetsフォルダの右クリック メニューから、[Import Package]-[Custom Package]を選択します







プロジェクトウィンドウからのインポート

# VisAssetsのインポート(2)

ダウンロードしたVisAssetsのUnityPackageを選択し、「開く」ボタンを押します

Import package			×
← → ∽ ↑ 📕 > F	PC > デスクトップ > VisAssets	م <b>ن</b> ب	VisAssetsの検索
整理▼ 新しいフォルダ・	-		::: • 🔳 ?
OneDrive	* 名前	更新日時	種類 サイズ
S PC	visassets_20120215	2021/02/15 9:21	Unity package file 201
<ul> <li>              ダウンロード                  デスクトップ                 デスクトップ                 ドキュメント                 ビクチャ                 ビブオ                 ビデオ                 ミュージック</li></ul>	<		>
ファイル	焰( <u>N</u> ): visassets_20120215		itypackage ~ 開く( <u>O</u> ) キャンセル

# VisAssetsのインポート(3)

・ 開いた「Import Unity Package」ウィンドウの「Import」ボタンを押します



# VisAssetsのインポート(4)

・ パッケージインポート前後のプロジェクトウィンドウの表示の変化

Project 🖻 Console						
+-						٩
<ul> <li>Favorites         <ul> <li>All Materials</li> <li>All Models</li> <li>All Prefabs</li> </ul> </li> <li>Fassets         <ul> <li>Scenes</li> <li>Scenes</li> </ul> </li> </ul>	Assets					
Project E Console			Ŷ	-		
+•						٩
▼ ★ Favorites Q. All Materials Q. All Models Q. All Prefabs ▼ ➤ Assets	Assets					
<ul> <li>■ Editor</li> <li>▶ ■ Resources</li> <li>■ Scenes</li> <li>■ StreamingAssets</li> <li>▶ ■ VisAssets</li> <li>▶ ■ Packages</li> </ul>	Editor	Resources	Scenes	Streaming	VisAssets	

# 実行に必要なアセットの追加(1)

- 読み込み対象となるファイルを選択する際のファイルダイアログについて、VisAssets ではアセットストアで無償公開されている「Runtime File Browser」を使っています
- ・ このような便利なアセットを利用できるのもUnityの特徴です(有償のものもあります)



## 実行に必要なアセットの追加(2)

アセットストアの検索窓で「Runtime File Browser」を検索します
 ※ タブに「Asset Store」が無い場合はメニューの[Window]ー[Asset Store]を選択

Visualization Workshop - SampleScene - PC, Mac & Linux Standalone - Unity 2019.4.18f1 Per	rsonal <dx11></dx11>	〕タブでアセットス	トアに切り替え	- o ×
le Eait Assets GameObject Component Window Help	Bell Asset - Feedback FAQ  My Assets	Open in Brows	Collab V 📥 Accou : 🔁 Inspector ier English V USD -	int ▼ Layers ▼ Layout ▼ 금 :
♀ Main Camera ♀ Directional Light	Save up to <b>96% on Lunar New Yea</b>	r Mega Bundles!		
Q Runtime File B	9 Browser	■ ○ ▲ : 2 検索窓で「Rur ※ 大文	# & Alignment ntime File Brows 、字小文字に注意	er」を検索
Runtime File Brow ×		Hide Purchased Asse	ts	
		All Categories	~	
FILE BROM	ISER	Pricing	+	
Project Console		3 Runtime File B	Browser を選択	
I Editor E Assurces I Scenes ■ StreamingAssets ■ StreamingAssets ■ Contemport ■	actSca ExtractVec Isosurface Outline	ReadField ReadGrADS ReadV5	Silcer StreamLin	
Assets(VisAssets(Scripts)Readers)ReadField.cs(8,7): error CS0246. The type or namespace	e name 'SimpleFileBrowser' could not be found (a	re you missing a using directive or an asse	mbly reference?)	Auto Generate Lighting Off

# 実行に必要なアセットの追加(3)

• 「Runtime File Browser」をプロジェクトにインポートします



# 実行に必要なアセットの追加(4)

• 「Import Unity Package」ウィンドウの「Import」ボタンを押す



# 実行に必要なアセットの追加(補足)

 「Runtime File Browser」がアセットストアからうまく取得できない場合は githubからダウンロードしたUnityPackageを使っても構いません

https://github.com/yasirkula/UnitySimpleFileBrowser



## 可視化モジュールの場所

• 可視化モジュールは VisAssets/Prefabs 以下にインポートされます

Project E Console			a:
	Assets		
-		_	
Project	Assets > VisAssets	٩	3: ₽ ★ Ø8
Q. All Materials Q. All Models Q. All Prefabs ■ Editor ■ Editor ■ Resources ■ StreamingAssets > M VisAssets > M Packages	Editor Images Materials Models Prefabs Scene	s Scripts Shaders	
_	Assets/VisAssets	_	
Project ⊡ Console	Assets > VisAssets > Prefabs		∃: ★ ★ ∞8
Q. All Prefabs ■ Elitor > Belitor > Besources Besources > StearningAssets > WestearningAssets > Westearnin	Animator Arrows Bounds Downdos ExtractSca ExtractVe	rc Itosurface Outline ReadField ReadGrADS Rea	dV5 Slicer StreamLin
Scenes	Assets/VISASSETS/P(e)abs		

#### データを読み込む(1)

- VisAssetsでは I×J×K(×時刻)の構造格子データを扱うことができます
- サンプルモジュールとして、3つのデータ読み込み用モジュールを用意しており、
   ここではテキストファイルからデータを読み込む「ReadField」モジュールを使ってみます



モジュール名	読み込み可能なデータ
ReadField	テキストファイル
ReadV5	<u>CAVE型VR装置用可視化ソフトウエア「VFIVE」</u> 用データ (テキスト+バイナリ)
ReadGrADS	<u>大気海洋データ用可視化ソフトウエア「GrADS」</u> 用データ (テキスト+バイナリ)

#### データを読み込む(2)

• ReadFieldモジュールで読み込み可能なテキストデータの例

11, 11, 8 格子数 (I × J × K) -5, -5, -3.5, 7.889866919, -5, -5 -4, -5, -3.5, 7.297259760, -4, -5 -3, -5, -3.5, 6.800735254, -3, -5 -2, -5, -3.5, 6.422616289, -2, -5 -1, -5, -3.5, 6.184658438, -1, -5 0, -5, -3.5, 6.103277808, 0, -5 1, -5, -3.5, 6.184658438, 1, -5 2, -5, -3.5, 6.422616289, 2, -5 3, -5, -3.5, 6.800735254, 3, -5 4, -5, -3.5, 7.297259760, 4, -5 5, -5, -3.5, 7.889866919, 5, -5 ~(後略)~ 格子点上の値 格子点座標 (3成分)

#### データを読み込む(3)

- アセットストアの表示のままであればシーンビューに戻しておきます
- プロジェクトウィンドウから「ReadField」モジュールを、ヒエラルキーウィンドウの 「Main Camera」、「Directional Light」と同じ階層にドラッグアンドドロップします



#### データを読み込む(4)

 ・ ヒエラルキーウィンドウで「ReadField」モジュールを選択し、インスペクタウィンドウの TransformのPositionが原点(X=0, Y=0, Z=0)でない場合はリセットしておきます

: (• All	<ul> <li>Inspector</li> <li>ReadField</li> <li>Tag VisModul</li> <li>Prefab Open</li> </ul>	l e <del>▼</del> La Select	ayer Defa Overrie	A : Static ▼ oult ▼ des ▼	
x Z	▼ <mark>↓</mark> Transform Position Rotation	X O X O	Y 0 Y 0	❷ ∓ : Z 0 Z 0	
	Scale <b>#</b> Activation ( For	X 1 (Script) rce Module U	Y 1 Ipdate	9 7 i	左クリックで表示される メニューから「Reset」を選択
	<ul> <li>Read Field</li> <li>Filename:</li> <li>Load At Startup</li> <li>Use Embedded</li> </ul>	<b>(Script)</b> Data		<b>9</b> ;	

#### データを読み込む(5)

- 「ReadField」モジュールでのデータ読み込み方法には以下の三種類があります
  - 1. モジュール内蔵のテスト用データを使う
  - 2. インスペクタで指定されたデータファイルを読み込む
  - 3. ビルド後のアプリケーション実行中に、ファイルダイアログ<sup>※</sup>からデータファイルを 指定して読み込む
- 本チュートリアルでは 1. のモジュール内蔵のテスト用データを使う方法、
   または 2. の指定したデータファイルをビルド後のアプリケーションに組み込む方法を
   紹介します

※アセットストアからインポートしたUnitySimpleFileBrowserはここで使っています

#### データを読み込む(6)

・「ReadField」モジュールのインスペクタ表示と設定項目は下記の通りです

		読み込み対象とするファイル名を入力
	Read Field (Scrip.) Filename:	0 ≓ :
指定したファイルを起動直後 に読み込む場合はチェック	Load At Startup	指定したファイルをビルド後のアプリケーションの 組み込みデータとして内蔵する場合はチェック
	If you use an embedded data, it must "Assets/StreamingAssets".	t be placed in
	Use Dummy Data Current Step: 0	ρ_+ :
	Tata Field (Script)	モジュール内蔵のダミーデータを使って 動作確認する場合はチェック

#### データを読み込む(7)

 ・ ヒエラルキーウィンドウで「ReadField」を選択し、インスペクタウィンドウ内の 「Read Field (Script)」のオプションを下記のいずれかに設定します





#### モジュール内蔵のテスト用データを使用する場合 11×11×11、要素数2

データファイルをアプリケーションに組み込む場合 11×11×8、要素数3 (Assets/StreamingAssetsフォルダに配置済み)

#### データを読み込む(8)

- データが正しく読み込まれたかどうかの確認のため、
   データ境界線を表示するBoundsモジュールを使ってみます
- プロジェクトウィンドウから「Bounds」モジュールをヒエラルキーウィンドウの 「ReadField」モジュールの子となるようドラッグアンドドロップします



#### ここまでの状態の確認(1)

 ツールバーの再生ボタンを押すとゲームビューに実行結果が表示されます (この段階ではデータ領域を示す線のみが表示されます)



#### ここまでの状態の確認(2)

 ゲームビューに切り替え、左ドラッグすることで回転、ホイールの前後で拡大縮小表示 ができます


#### ここまでの状態の確認(3)

- インスペクタの DataField の項で読み込まれたデータが確認できます
- Element IDのスライダーを左右に動かすと各要素の詳細が表示されます

🔻 錯 🛛 Data Field	(Script)	0 7 i	🔻 # 🛛 Data Field	(Script)	0 7 i	🔻 # 🛛 Data Field	(Script)	0 :
Data Type		•	Data Type		-	Data Type		•
Data Loaded	~		Data Loaded			Data Loaded		
Element ID:	•		Element ID:		— 1	Element ID:		2
🔻 Element 0		/	▼ Element 1			T Element 2		
▶ Dims			▶ Dims			▶ Dims		
Ndim			Ndim			Ndim		
Size			Size			Size		
Steps			Steps			Steps		
Scale			Scale			Scale		
Min			Min			Min		
Max			Max			Max		
Average			Average			Average		
Variance			Variance			Variance		
Var Name			Var Name			Var Name		
Use Undef			Use Undef			Use Undef		
Undef			Undef			Undef		
Field Type			Field Type			Field Type		*
Is Active			Is Active			Is Active		

#### ここまでの状態の確認(4)

再生ボタンをもう一度押すとゲームビューでの実行が終了します
 (実行中の回転・拡大縮小の操作は保存されず実行前の状態に戻ります)



### 接続を間違えた場合

- 下図ではBoundsがReadFieldの下に接続されていないため無効な接続となります
- 実行時に無効なモジュールは非アクティブな状態となり、アプリケーションの動作にも 影響を与えません



#### シーンの保存

- 再生を止めた状態で、メニューの「File」-「Save」で現在のシーンを保存します (ヒエラルキーウィンドウでCtrl-Sでも可)
- 以下、進捗確認後に適宜シーンを保存することをおすすめします



### カラースライスを表示する(1)

 プロジェクトウィンドウから、「Slicer」モジュールをヒエラルキーウィンドウの 「ReadField」モジュールの子となるようドラッグアンドドロップして接続します



### カラースライスを表示する(2)

 ツールバーの再生ボタンを押すとゲームビューにBoundsによるデータ領域に加えて カラースライスが表示されます



#### カラースライスを表示する(3)

 実行中にヒエラルキーウィンドウで「Slicer」を選択し、インスペクタウィンドウ内の オプションを変更することで、軸、スライスレベル(0~1の範囲)等を設定できます

Inspector	a :					
Slicer	🗌 Static 🔻					
Tag VisModule    Layer Defa	ult 👻					
🔻 🗰 Activation (Script)	0 ‡ i					
Force Module Update						
🔻 🛱 🖌 Slicer (Script)	0 ‡ i					
Filter Mode: TRILINEAR	-					
Axis:	- 0					
Slice:	- 0					
Color Shift:	- 0					
Unlit/SliceShader (Instance)	0 ‡ ¢					
Shader Unlit/SliceShader	•					
Add Component						



#### カラースライスを表示する(4)

・ 次の作業に移る前に、再生ボタンを押してゲームビューでの実行を終了します



#### 任意の要素のカラースライスを表示する(1)

- ReadFieldの直下に接続したSlicerへの入力データは、入力データに含まれる要素の内、最初の要素となります
- 他の要素をSlicerへの入力としたい場合には「Extract Scalar」モジュールを挿入し、
   任意の要素を選択できるようにする必要があります



#### 任意の要素のカラースライスを表示する(2)

実行すると「Extract Scalar」のインスペクタを使って、
 入力データからSlicerに渡す要素(Channel)を一つ選択可能となります

Inspector			2	:
💬 🗸 ExtractSca	lar			с 🕶
Tag VisModule				•
🔻 🗰 Activation (S	cript)	0		:
Forc	e Module Update			-
🔻 # 🔽 Extract Scala	ar (Script)	0		:
Channel varial	ble 0			•
v	variable 0			
🔻 🗰 🛛 Data Fi	ariable 1	0		:
Data Type	variable 2			¥
Element ID:				



Activat	ion (Scrint)	Ø 7 :			
M Activat		•• :			
	Force Module Update				
🛛 # 🔽 Extract	Scalar (Script)	0 7 i			
Channel variable 1					
# Data Fi	eld (Script)	07:			
Element ID:		0			

Tag Vis			Layer				
Activ	vation (So	cript)			0		
Force Module Update							
🖷 🗹 Extra	act Scala	r (Scrip	t)		0		
Channel variable 2							
📕 Data	Field (So	cript)			0		
Element ID:					0		

### 複数のカラースライスを表示する(1)

実行を停止し、「Extract Scalar」の子として「Slicer」を二つ追加します
 (同一階層の同名モジュールにはUnityエディタが識別子を自動で付与します)



### 複数のカラースライスを表示する(1)

 再生ボタンで実行し、各「Slicer」モジュールの設定をインスペクタから Slicer:Axis=0、Slicer (1):Axis=1、Slicer (2):Axis=2、いずれもSlice=0.5 とすると下図のような結果が得られます(下図では視点を変えてあります)



## 等値面を表示する(1)

- 実行を停止し、「Extract Scalar」の子として「Isosurface」を追加します
- 「Slicer」とは別の要素の等値面を表示したい場合は、「ReadField」の下に 「Extract Scalar」を追加し、その下に「Isosurface」を接続します



### 等値面を表示する(2)

 「Isosurface」モジュールのインスペクタから閾値(Threshold)を変更することで 等値面が生成されます

(Shading Modeが選択できますが、諸事情によりFLATのままとしてください)



### ユーザインタフェースの追加(1)

 実行を停止し、「UIManager」モジュールをヒエラルキーウィンドウに「Main Camera」
 と同じ階層にドラッグアンドドロップして追加することにより、ゲームビューの右端にUIが 追加されます



### ユーザインタフェースの追加(2)

 再生ボタンで実行すると、ゲームビュー内のUIでの操作対象となるモジュール選択と、 選択中のモジュールのパラメータ変更ができるようになります(インスペクタからのパラ メータ変更はUnityエディタでの実行中のみ可能で、ビルド後にはできません)

Modules Slicer#1
Visibility Filter Mode POINT V Axis: I V J K
Color Shift

# ビルド前の準備(1) ~組み込みシェーダーの登録~

・ メニューから[Edit]-[Project Settings]でProject Settingsウィンドウを表示し、 そこで「Graphics」を選択します

🜣 Project Settings						: = ×	
				٩			
Audio	Graphics					0 ⊉ ⊅	
Editor	Scriptable Render Pipeline Settings None (Render Pipeline Asset) Compare Settings						
Input Manager							
Package Manager							
Physics Physics 2D	Transparency Sort Mod	le	Default				
Player	Transparency Sort Axis		хo		YO	Z 1	
Preset Manager							
Script Execution Order	▼ Tier Settings					Open Editor	
Tags and Layers	Ţ				÷	9	
TextMesh Pro	Low (Tier1)					Use Defaults 🗸	
VFX							
XR Plugin Management							
						<b>•</b>	
	Medium (Tier 2)					Use Defaults 🖌	
	HDRIVIOUE				1110		

## ビルド前の準備(2) ~組み込みシェーダーの登録~

 「Graphics」内の[Built-in Shader Settings]-[Always Included Shaders]の Sizeを現在の値から2つ増やします(下図では9から11に変更しリターンキーで確定)

🗱 Project Settings				: □×			
Audio	Graphics			0 ‡ ¢			
Editor							
Graphics	Duilt in Charles Cattings						
Package Manager	Built-In Snader Settings	Puilt in chador					
Physics	Deferred Deflections	Built-in shader					
Physics 2D	Legeov Deferred	Built-in shader					
Player Preset Manager	Legacy Deletted	Built-in shader					
Quality	Depth Normals	Built-in shader					
Script Execution Order	Motion Vactors	Built-in shader					
Tags and Layers	Light Halo	Built-in shader					
Textmesh Pro		Built-in cheder					
VFX	Always Included Shade						
XR Plugin Management	Size	Size 0					
	Element 0	I enacy Shaders/Diffuse		•			
	Element 1	B Hidden/CubeBlur					
	Element 2	Hidden/CubeCopy					
	Element 3	S Hidden/CubeBlend		0			
	Element 4	Sprites/Default		0			
	Element 5	SUI/Default		0			
	Element 6	Hidden/VideoComposite		0			
	Element 7	S Hidden/VideoDecode		0			
	Element 8	S Hidden/Compositing		$\odot$			
	Chadae Christian						
	Lightman Modes						
	Eighthap Modes	Automatic					
	Instancing Variants	Strin Unused					
	instancing variants	outpoindood					
	Shader Loading						
	Log Shader Compilation						
	Preloaded Shaders	6					
	Size	0					
	Currently tracked: 15 sha	iders 20 total variants					
			Save	to asset Clear			
				Ť			

### ビルド前の準備(3) ~組み込みシェーダーの登録~

 シェーダーリストの末尾に追加された「Element 9」と「Element 10」に SimplePhongとSliceShaderを設定します(順不同)

🌣 Project Settings				: □×		
Audio Editor Graphics Input Manager	Graphics Rendering Path Realtime Global Illum			0 ⊉ ¢ ▼ ↑	Select Shader	
Packager Manager Physics 2D Physics 2D Player Preset Manager Quality Script Execution Order Tags and Layers TextMesh Pro Time VFX XR Plugin Management	Built-in Shader Settings Deferred Deferred Reflections Legacy Deferred Screen Space Shadows Depth Normals Motion Vectors Light Halo Lens Flare RAWays Included Shade Size Element 0 Element 1 Element 2 Element 3 Element 3 Element 4 Element 5 Element 5 Element 7 Element 8 Element 7 Element 8 Element 9 Element 9 Element 9 Element 10 Shader Stripping Lightmap Modes Fog Modes Instancing Variants Shader Loading Log Shader Compilation Preloaded Shaders Size	Built-in shader Built-in shade	ここを押して開いた ウィンドウから シェーダーを選択		Assets None SimplePho SliceShader Sur	S faceSh GU

### ビルド前の準備(4) ~組み込みシェーダーの登録~

• 組み込みシェーダーの登録後のGraphics設定画面

🗘 Project Settings				: 🗆 ×		
Audio Editor	Graphics			0 ⊉ ≎		
Graphics Input Manager						
Package Manager	Built-in Shader Settings					
Physics Physics 2D	Deferred	Built-in shader				
Player	Deferred Reflections	Built-in shader				
Preset Manager	Legacy Deferred	Built-in shader				
Quality	Screen Space Shadows	Built-in shader				
Script Execution Order	Depth Normals	Built-in shader				
TextMesh Pro	Motion Vectors	Built-in shader				
Time	Light Halo	Built-in shader				
VFX	Lens Flare	Built-in shader				
XR Pluqin Management	▼ Always Included Shaders					
		11				
	Element 0	Legacy Shaders/Diffuse				
	Element 1	Hidden/CubeBlur				
	Element 2	S Hidden/CubeCopy				
	Element 3	S Hidden/CubeBlend				
	Element 4	Sprites/Default				
	Element 5	SUI/Default		$\odot$		
	Element 6	B Hidden/VideoComposi	te	$\odot$		
	Element 7	B Hidden/VideoDecode		$\odot$		
		lidlen/Compositing				
	Element 9	Custom/SimplePhong		0		
	Element 10	Unlit/SliceShader		$\odot$		
	Chadar Stripping					
	Lightman Modes					
	Eighthap Modes	Automatic				
	Instancing Variants	Strin Unused				
		Sulp Shuseu				
	Shader Loading					
	Log Shader Compilation					
	▼ Preloaded Shaders					
		0				

### シェーダーを登録せずにビルドした場合

- Unityエディタでは正常に動いていたアプリケーションですが、この後のステップで ビルドが正常に完了しても、実行時にシェーダーが見つからずSlicerが図のような 表示になります
- モジュールの実行途中に内部エラーとなるため、UIのモジュール一覧にもSlicerが 登録されていません(Isosurfaceも同様です)



# Windows用実行ファイル作成(1)

 メニューから[File]-[Build Settings]でビルド設定を開き、Platformが「PC, Mac & Linux Standalone」になっていることを確認し、「Add Open Scenes」ボタンで 現在開いているシーンをビルド対象として追加します



## Windows用実行ファイル作成(2)

「Build」ボタンを押し、ビルドしたファイルの出力先を指定します。ビルドの際、複数のファイルが出力されるため、下の図ではプロジェクトディレクトリ内に「Build」という名前のフォルダを新規に作成し、出力先として指定しています

	and one			
→ ` ↑ 📙 « U	nity Projects > Visualization Workshop >	~ Ū	Visualization Work	shopの検索
里▼ 新しいフォルダー				· 0
OneDrive	~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~	更新日時	種類	サイズ
DC	.vs	2021/02/15 14:46	ファイル フォルダー	
	📕 Assets	2021/02/15 15:07	ファイル フォルダー	
	🧵 Build	2021/02/15 18:39	ファイル フォルダー	
	📙 Library	2021/02/15 18:35	ファイル フォルダー	
テスクトップ	📙 Logs	2021/02/15 9:45	ファイル フォルダー	
三 ドキュメント	📜 obj	2021/02/15 14:46	ファイル フォルダー	
■ ピクチャ	Packages	2021/02/15 14:48	ファイル フォルダー	
🚼 ビデオ	ProjectSettings	2021/02/15 9:46	ファイル フォル	エキャットマスズナカリックメニュ
🎝 ミュージック	🣜 Temp	2021/02/15 18:39	ファイル フォル	」もないとこうで石 クリックスニュー
Blade (C:)				新規ノオルタを作成できます
ネットワーク 🗸 🗸	<		_	>
フォルダ	-: Build			
		74	ルダーの選択キ	ャンセル

# Windows用実行ファイル作成(3)

- ・ ビルドが完了するとビルドフォルダ以下に実行ファイルが生成されます
- プロジェクトフォルダはUnityエディタのプロジェクトウィンドウでの右クリックメニューから「Show in Explorer」を選択することでも開くことができます
- ・ 実行ファイルは下の図でハイライトされている右端のファイルです



## Windows用実行ファイル作成(4)

- 生成された実行ファイルをダブルクリックすると、構築したアプリケーションが全画面 表示で実行されます
- GUIからモジュール選択やパラメータ設定ができることを確認してください。GUIはダブ ルクリックで表示/非表示を切り替えることができます
- ・ アプリケーションの終了は Alt + F4 キーです



## [参考] PC-HMDでの立体視表示

 メニューから[Edit]-[Project Settings]でProject Settingsウィンドウを表示し、 Playerを選択、最下部にある[XR Settings]の「Virtual Reality Supported」を有 効にすることで、Unityエディタでの実行、ビルド後の実行ファイル共にPC-HMDでの 表示が可能となります(Oculus Quest 2 + Oculus Link、StermVR(OpenVR)で 実行を確認しています)

C Project Settings			: 🗆 ×	
Project Settings  Audio Editor Graphics  Prackage Manager Physics Physics 2D Player Preset Manager	Player Company Name Product Name Version Default Icon	م DefaultCompany Visualization Workshop 0.1	E □ × <b>XR Settings</b> 360 Stereo Capture Built-in XR is deprecated instead. You can find set	and will be retired in a future version of Unity. Use the new Unity XR Plugin System tings for Unity XR Plugin System in Project Settings under XR Plugin Management.
Quality Script Execution Order Tags and Layers TextMesh Pro Time VFX	Default Cursor	xo	Deprecated Settings Virtual Reality Supported Virtual Reality SDKs	
	Settings for PC, Mac & Linux Standalone Icon Resolution and Presentation Splash Image Other Settings Up other Settings	.cc <b>∳</b> e	<ul> <li>■ ▼ Oculus</li> <li>Shared Depth Buffer</li> <li>Dash Support</li> <li>= OpenVR</li> </ul>	
	XR Settings     XR Settings     Xelevel Capture     Perrocated and will be:     Network of the settings     Virtual Reality SUpported     Virtual Reality SDKs     ■ ♥ Oculus     Shared Depth Buffer     Dash Support     Open/R	retired in a future version of Unity. Use the by "R may S term in the ot "et may and V	new Unity XR Plugin System	※ PCへのHMDの接続が必要です ※ XR Plugin Management には未対応です

#### [参考] Android用ビルドの準備

[開発者オプションの有効化]

 [設定]-[デバイス情報]にあるビルド番号を7回タップして開発者オプションを有効に する(PIN/パターン/パスワードの入力が必要になります)

[USBデバッグの有効化]

- [設定]-[システム]-[開発者オプション]でUSBデバッグを有効にする
- USBケーブルでPCに接続した際、「USBデバッグを許可しますか」のダイアログが 出たら「許可」を押す

[開発者オプションの無効化]

・ [設定]-[システム]-[開発者オプション]で最上部のスイッチをONからOFFにする

※機種によって設定項目の場所や表示されるメッセージが異なる可能性があります ※WS終了後、不要であれば開発者オプション、USBデバッグは共に無効にしてください。 特にUSBデバッグを有効のままにした場合、一部のアプリが動作しない場合があります

## [参考] Android用ビルド(1)

 メニューから[File]-[Build Settings]でビルド設定を開き、ターゲットプラットフォーム をPCからAndroidに変更した後「Switch Platform」ボタンを押して確定します

Build Se	ettings		
Scene	s In Build		
✓ Sce	nes/SampleScene		0
			Add Open Scenes
Platfo	rm	_	
$\Box$	PC, Mac & Linux Standalone 🛛 🔫	🖷 Android	
4	Universal Windows Platform	Texture Compression	Don't override 👻
iOS	iQS	ETC2 fallback	32-bit 👻
<b>1</b>	Android	Symlink Sources Build App Bundle (Google F	Play
9	WebGL		
	tvOS	Development Build	
₽J4	PS4		
	Xbox One		
		Compression Method	LZ4 •
			Learn about Unity Cloud Build
Playe	er Settings	Swite	ch Platform Build And Run

## [参考] Android用ビルド(2)

 [Project Settings]-[Player]-[Other Settings]で「Auto Graphics API」に チェックが入っていることを確認します(後述のCardboard用)

Scenes in Build			
Proiect Settings			: 0
		٩	
	Player		<b>0</b> ≓
ditor	V Other Settings		
Fraphics	Rendering		
ackage Manager	Color Space*		
	Auto Graphics API		
hysics 2D	Color Gamut*		
rayer reset Manager			
Quality			ī <del>t.</del> - [
cript Execution Order	Multithreaded Rendering*	7	
ags and Layers TextMesh Pro	Static Batching		
Time	Dynamic Batching		
	Compute Skinning*		
R Plugin Management	Graphics Jobs (Experimental)		
	Lightmap Encoding	Low Quality	
	Lightmap Streaming Enabled		
	Streaming Priority	0	
	Enable Frame Timing Stats		
	Mullion Continue		
	Vulkan Settings		
	Number of evanebain buffere#		
	Acquire swapchain image late as possible*		
	Acquire swapenair inage late as possible	•	
	Identification		
	Package Name	com.DefaultCompany.VisualizationWorkshop	
		0.1	
	Minimum ADL avial	Android 4.4 'KitKat' (ADLlovel 19)	
	Target ADL evel	Automatic (highest installed)	
		Automatic (ingriest installed)	
	Configuration		
	Scripting Backend	Mono	
	Api Compatibility Level*	.NET Standard 2.0	
		Release	
	Use incremental GC		
	Assembly Version Validation Android TV Compatibility		
		- Dalia Dalia I	

[参考] Android用ビルド(3)

 Android機をデータ通信可能なUSBケーブルでPCに接続し、 「Build And Run」ボタンを押します



## [参考] Android用ビルド(4)

 ビルドしたファイルの出力先を指定します。下図ではWindows用ビルドの際に 作成した「Build」フォルダを出力先、「visassets.apk」というファイル名で出力します

Build Settings				×
Scenes In Bui ✓ Scenes/Sar	ld npleScene			0
Build Android				$\times$
← → × ↑ 📕 «	Visualization Workshop $>$ Build $>$	ٽ ~	➢ Buildの検索	
整理 ▼ 新しいフォルタ	<u>-</u>		=== •==	• ?
.S PC	<b>^</b> 名前 ^	更新日時	種類	サイズ
<ul> <li>3D オブジェクト</li> <li>ダウンロード</li> <li>デスクトップ</li> <li>ドキュメント</li> <li>ビクチャ</li> <li>ビデオ</li> <li>ミュージック</li> <li>Blade (C.)</li> </ul>	<ul> <li>MonoBleedingEdge</li> <li>Visualization Workshop_Data</li> <li>visassets.apk</li> </ul>	2021/02/15 18:45 2021/02/15 20:00 2021/02/15 20:45	ファイル フォルダー ファイル フォルダー APK ファイル	17,554
🥩 ネットワーク	✓ <			>
ファイル名( <u>N</u> ): <mark>vis</mark> ファイルの種類( <u>I</u> ): apl	assets K			~
ヘ フォルダーの非表示		I	保存( <u>S</u> ) キ	ャンセル
Player Settin	gs	Learn - Build	about Unity Cloud Bui	

## [参考] Android用ビルド(5)

- ・ ビルドが成功するとAndroid機上でアプリケーションが実行されます
- ・ 実機にインストールされるため、USBケーブルを抜いた後も実行できます
- ・「Build」ディレクトリ内のAPKファイルも転送してインストールが可能です
- ・ GUIで可視化モジュールの選択やパラメータの変更ができることを確認してください



ドラッグ: 回転

ピンチイン・ピンチアウト: 拡大縮小表示

ダブルタップ: GUIの表示/非表示の切り替え

# [参考] Cardboard対応ビルド(1)

 [Project Settings]-[Player]-[XR Settings]の「Virtual Reality Supported」を 有効にし、「Virtual Reality SDKs」を「None」、「Cardboard」の順にすることで、 Cardboardの二眼表示が可能となります

(画面左上のアイコンをタップ/ダブルタップで通常表示に戻ります)

Project Settings					: = >
			٩		
Audio Editor	Player				0 2 0
Graphics			0.1		
Input Manager Package Manager Physics Physics 2D	Default Icon				None (Texture 2D) Select
Player Preset Manager Quality Script Execution Order	Default Cursor				None (Texture 2D) Select
Tags and Layers TextMesh Pro	Cursor Hotspot		x o	Y O	
VFX	Ţ			Ť.	9
XR Plugin Management	Settings for Android				
	⊳ Icon				
	► Resolution and Pres	entation			
	▶ Splash Image				
	► Other Settings				
	▶ Publishing Settings				
	▼ XR Settings				
	Built-in XR is depreinstead. You can fi	cated and will be ret nd settings for Unity	ired in a future version of U XR Plugin System in Project	nity. Use the new Unity XR P Settings under XR Plugin Ma	lugin System nagement.
	Deprecated Settings Virtual Reality Supporte				
	Virtual Reality SDKs				

# [参考] Cardboard対応ビルド(2)

 ビルド時に "Company Name" のエラーが出る場合は[Project Settings]-[Player]内の「Package Name]を設定してください (com.visws.visassetsなどでよいです)

Froject Settings			: □
Audio	Player		<b>0</b> ,‡
ditor Graphice			
nout Manager			
ackage Manager	Streaming Priority	0	
	Enable Frame Timing Stats		
hysics 2D	Vulkan Settings		
layer			
nality		3	
cript Execution Order	Acquire swapchain image late as pos	sible*	
ags and Layers			
extMesh Pro	Deckers Name	ann DefaultCompany VisualizationWedlahan	
Time (FX	Package Name	com.DefaultCompany.visualizationworkshop	
/FX (P Plugin Management			
an rugin wunugement	Bundle Version Code	1	
	Minimum API Level	Android 4.4 'KitKat' (API level 19)	
	Target API Level	Automatic (highest installed)	
	Configuration		
		.NET Standard 2.0	
	Assembly Version Validation		
	Mute Other Audio Sources*		
	Target Architectures		
		<b>v</b>	
	Split APKs by target architecture (Exp	erimen	
		Prefer External	
	Internet Access	Auto	
	Write Dormission	Internal	
	Filter Touches When Observed		
	Sustained Defermence Mede		
	Sustained Performance Mode		
	Low Accuracy Location	-	
	An elected TV On energy Albitta		

## ベクトル場の可視化(1)

- ReadField用データにはベクトル場のデータが含まれていませんので、 「ReadV5」モジュールを使ってCAVE型VR装置用可視化ソフトウェアVFIVEの サンプルデータ(ABC Flow)に含まれるベクトル場データを可視化してみましょう
- ダウンロードしたデータ(sample\_big2.tar.gz)を展開し、アーカイブに含まれる8つの ファイルを「sample\_big2」という名前のフォルダに入れます (フォルダの作成場所はどこでも構いません)
- 「sample\_big2」フォルダをUnityエディタのプロジェクトウィンドウの Assets/StreamingAssets にドラッグアンドドロップし、アセットとして登録します

Project	E Console		
+ ▼ ▼ 🔶 Favorit	es	Assets > Strot	sing Accets
Q, All M	aterials		IIgAssets
् All Mo ् All Pr	odels efabs		<u> </u>
V 🖬 Assets	r	sample_bi	Sample3D3
🕨 🖿 Plugi	ns		
🕨 🖿 Resoi	urces es		
🕨 🖿 Strea	mingAssets		
VisAs	ssets		

## ベクトル場の可視化(2)

 現在開いているシーンを保存し、プロジェクトウィンドウから Assets/VisAssets/Scenes にある ReadV5Sample.scene をダブルクリックします


## ベクトル場の可視化(3)

- ・ このサンプルシーンでは既に可視化モジュールが接続されています
- ・ 次のスライドでABC Flowのデータを読み込むための設定をします



## ベクトル場の可視化(4)

 サンプルシーンはVFIVEの別のサンプルデータ(ダイナモシミュレーション)を対象として 設定されていますので、ABC Flowのデータ用に設定を変更します (ファイル名の変更の他、エンディアン変換のためByteswapをONにしています)

Inspector		а	:
🝸 🗹 ReadV5	<b>_</b> s	Statio	: <b>•</b>
Tag VisModule - Layer Defa	ult		
Prefab Open Select Overric	les		
🖂 Transform	0	7 <b>t</b>	
🕨 # Activation (Script)	0		
🔻 ≢ 🗹 Read V5 (Script)	0	-1 <del>1</del> -	
Filename: sample_little/dynamo.v5			
Precision: DOUBLE			
Byteswap 🖌 Header			
<ul> <li>Load At Startup</li> </ul>			
✓ Use Embedded Data			
If you use an embedded data, it must be plac "Assets/StreamingAssets".	ed in		
✓ Centering			
✔ Auto Resize			
<b>亦</b> 百前			

	Stati	<b>c</b> 7
Tag VisModule 🝷 Layer De	efault	•
Prefab Open Select Ove	rrides	-
🕨 🙏 🛛 Transform	9 ≓	
🕨 # Activation (Script)	0 i	
🔻 # 🖌 Read V5 (Script)	0 i	
Filename:		
sample_big2/abc_flow.v5		
Precision: DOUBLE ✓ Byteswap ✓ Header ✓ Load At Startup		~
<ul> <li>Use Embedded Data</li> </ul>		
<ul> <li>Use Embedded Data</li> <li>If you use an embedded data, it must be p "Assets/StreamingAssets".</li> </ul>	laced in	
<ul> <li>Use Embedded Data</li> <li>If you use an embedded data, it must be p "Assets/StreamingAssets".</li> <li>Centering</li> </ul>	laced in	

### ベクトル場の可視化(5)

#### • 再生ボタンを押し、読み込みが完了すると下記のような表示になります



# ベクトル場の可視化(6)

 ・ ヒエラルキーウィンドウで「Extract Vector」モジュールを選択し、インスペクタから ベクトル場の各軸成分(Channel 0~2)を下図のように設定します

各Channelをチェックボックスで有効にすることで、各軸成分が選択可能になります

Inspector			а	
💬 🗹 Extract	Vector			
Tag VisMod	ule 🔻 Layer Defa	ult		
🕨 🗰 🛛 Activatio	n (Script)	0		
🔻 # 🗹 Extract V	/ector (Script)	0		
Channel 0				
Channel 1				
Channel 2				
All input eleme grids.	nts must have the same i	numb	er o	f
🔻 # 🛛 Data Fiel	d (Script)	0		
Element ID:	•	— 0		

Inspector		А	
Extract	Vector		
Tag VisMoo	lule 🔻 Layer De	fault <sup>,</sup>	•
🕨 # 🛛 Activatio	on (Script)	0 7±	
🔻 🗰 🔽 Extract \	/ector (Script)	0 ∔	
Channel 0	Velocity Field (U)		•
🖌 Channel 1	Velocity Field (V)		•
✓ Channel 2	Velocity Field (W)		
🔻 # 🛛 Data Fie	ld (Script)	Ø∴⊧	
Element ID:	•	— o	
▼ Element 0 ▶ Dims			
	赤市後		

## ベクトル場の可視化(7)

・ シーン上に表示されたUIManagerのGUIからの設定も可能です

各Channelをチェックボックスで有効にすることで、各軸成分が選択可能になります

	Tag VisM
Modules	🕨 🗰 🛛 Activat
ExtractVector	🔻 🛱 🖌 Extract
Visibility	Channel 0
	Channel 1
Variable #0	
Velocity Field (U)	Channel 2
Variable #1	
Velocity Field (V)	
Variable #2	
Velocity Field (W)	🔻 🗰 Data Fi
	Data Type
	Data Loaded
	Element ID:
	▼ Element 0
	▶ Dims
	Ndim
	Size

### ベクトル場の可視化(8)

- ・ 設定が完了すると下記のような表示になります
- 「Extract Vector」モジュールで設定したベクトル場が、子として接続された 「Arrows」モジュールによりゲームビューの左側に多数の矢印として描画されます



## ベクトル場の可視化(9)

- 「Arrows」モジュールは、ベクトル場データを入力とし、指定した断面上のベクトル場 を矢印で表現するためのモジュールです
- インスペクタから、またはシーン上のGUIから軸と断面位置の設定ができます



Axis=0, Slice=0.5



Axis=1, Slice=0.5



Axis=2, Slice=0.5

### ベクトル場の可視化(10)

・ 等値面についても併せて設定したものが下図となります



### 新たなモジュールの開発(1)

#### VisAssetsにおける可視化モジュール

• 必要な機能を実装したC#スクリプトなどをアタッチしたゲームオブジェクト

Inspector	a :	Inspector	a :	:		
✓ ReadField         S           Tag         Untagged         ✓	Static <del>▼</del>	✓ ReadField Tag VisModule    Layer Defa	Static <del>-</del> ault -	• •		
🔻 🙏 Transform 🛛 🚱		Prefab Open Select Overri				
Position X 0 Y 0 Z 0	0	🔻 🙏 🛛 Transform	0 :  :			
Rotation X 0 Y 0 Z 0	0	Position X 0 Y 0	Z 0			
Scale X 1 Y 1 Z 1	1	Rotation X 0 Y 0	ZO			
Add Component		Scale X 1 Y 1	Z 1			
		T # Activation (Script)	07:			
		Force Module Update				
		🔻 # 🖌 Read Field (Script)	07:			
		Filename:				
		✓ Load At Startup				
		Use Embedded Data				
		If you use an embedded data, it must be plac "Assets/StreamingAssets".	ced in			
		<ul> <li>Use Dummy Data</li> </ul>				
		Current Step: 0				
		🔻 # 🛛 Data Field (Script)	07‡ :			
		🕨 # 🗸 Ctrl OBJ (Script)	0 :  :			l
		Add Component				

空のゲームオブジェクト

ReadFieldモジュール

### 新たなモジュールの開発(2)

 モジュール間の接続チェックや、パラメータ変更の他のモジュールへの伝達については パッケージに含まれるテンプレートクラスが処理するため、開発者はテンプレートクラス
 を継承した新たなクラスのオーバーライド関数に必要な機能を実装すればよい



ReadFieldモジュール

### 新たなモジュールの開発(3)

• C#スクリプトの例(1): データ読み込みモジュール用

```
// 読み込みモジュール用テンプレートクラスを継承したクラス
public class ReadField : ReadModuleTemplate
 public override void InitModule()
   // モジュールの初期化処理を記述
 public override int BodyFunc()
   // Unityにおけるフレーム毎のイベントループで親モジュールや自身のパラメータを監視し、
   // それらに変更があった場合に実行される処理を記述する
   // このスクリプトの場合はデータ読み込み用なので、
   // ファイルダイアログから指定されたファイルや組み込みデータの読み込み処理を記述する
```

### 新たなモジュールの開発(4)

#### • C#スクリプトの例(2): フィルターモジュール用

```
// フィルターモジュール用テンプレートクラスを継承したクラス
public class ExtractScalar : FilterModuleTemplate
 // 下の二つのメンバ変数はテンプレートクラスで宣言しているため
 // 継承したクラスではそれぞれ pdf, df の変数名でアクセス可能
 // public DataField pdf; // 親モジュールのDataField
 // public DataField df; // 自身のDataField
 public int selected channel;
 public override void InitModule()
   // モジュールの初期化処理を記述
   selected_channel = 0;
 public override int BodyFunc()
   // Unityにおけるフレーム毎のイベントループで親モジュールや自身のパラメータを監視し、
   // それらに変更があった場合に実行される処理を記述する
   df.elements[0] = pdf.elements[selected_channel].Clone();
```

### まとめ

- 本チュートリアルでは、Unity用可視化フレームワーク「VisAssets」をハンズオン形式 で体験して頂きました
- 本フレームワークは現在も開発中のものであり、同梱した各モジュールもサンプル実装のため、高速化など改良の余地は多々あります
- 新たなモジュール開発もしやすい設計となっていますので、よろしければ是非皆さんも 開発にご参加ください

### 謝辞

本研究の一部はJSPS科研費JP19K11995の助成を受けたものです。

# [付録] VFIVE用データの読み込み

CAVE型VR装置用可視化ソフトウェア VFIVE

<u>https://www.jamstec.go.jp/ceist/aeird/avcrg/vfive.ja.html</u>

ダイナモシミュレーションデータ (sample\_little.tar.gz)

- アプリ組み込みデータとする際は Assets/StreamingAssets/sample\_little の下にデータを配置し、下記の ようにReadV5モジュールの設定をする
- ReadV5の設定: Filename = sample\_little/dynamo.v5
   Load At Startup = ON, Use Embedded Data = ON
   Precision=DOUBLE, byteswap = OFF, header = ON

ABC Flow (sample\_big2.tar.gz)

- アプリ組み込みデータとする際は Assets/StreamingAssets/sample\_big2 の下にデータを配置し、下記の ようにReadV5モジュールの設定をする
- ReadV5の設定: Filename = sample\_big2/abc\_flow.v5
   Load At Startup = ON, Use Embedded Data = ON
   Precision=DOUBLE, byteswap = ON, header = ON

# [付録] GrADS用データの読み込み

GrADS (The Grid Analysis and Display System)

- http://cola.gmu.edu/grads/
- ・ テキストファイル(読込用)+バイナリファイル(データ本体)で構成
- サンプルモジュール(ReadGrADS)では非圧縮のバイナリデータのみに対応 (本家GrADSはNetCDFやGRIBにも対応)

サンプルデータ

• ダウンロードページにある example.tar.gz