



FINAL FANTASY XV -EPISODE DUSCAE-の エフェクトはこうして作られた

～Luminous VFX Editorの紹介～

株式会社スクウェア・エニックス
長谷川 勇 野副 竜太 小野 哲平

本セッションのゴール

- 伝えたいこと
 - 次世代機でのエフェクトデザインの一例
 - ノードベースエフェクトツールのメリット・効果的な運用
 - 実現時の課題とその解決方法
- 対象者
 - エフェクト開発に関わる人
(現在関わっていないが興味がある人を含む)

アジェンダ

- 次世代機でのエフェクトデザインの一例 ← アーティスト向け
 - FFXV-EPISODE DUSCAE-におけるエフェクトデザイン
 - Luminous VFX Editorの紹介
- ノードベースエフェクトツールのメリット・効果的運用
 - ノードベースのメリット
 - Luminous VFX Editorのツール設計 ← アーティスト・TA・プログラマ向け
- 実現時の課題とその解決方法
 - 実装時の技術的課題
 - Luminous VFX Editorにおける解決方法 ← プログラマ向け

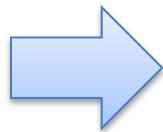
アジェンダ

- 次世代機でのエフェクトデザインの一例 ←アーティスト向け
 - FFXV-EPISODE DUSCAE-におけるエフェクトデザイン
 - Luminous VFX Editorの紹介
- ノードベースエフェクトツールのメリット・効果的運用
 - ノードベースのメリット
 - Luminous VFX Editorのツール設計
- 実現時の課題とその解決方法
 - 実装時の技術的課題
 - Luminous VFX Editorにおける解決方法

FINAL FANTASY XV-EPIISODE DUSCAE- におけるエフェクトデザイン

ゲームエフェクトとは

- ・ ゲームエフェクトは
ゲーム世界の中に存在するもの



**ゲームの世界観はエフェクトの
デザイン上で重要な情報**

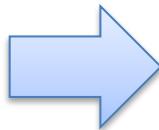
FINAL FANTASY XVの世界観

- ・ フォトリアルに描くファンタジー世界



世界観を踏まえた上で…

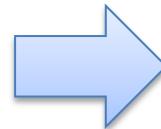
- FINAL FANTASY XVで目指すべき
エフェクトデザインは



フォトリアルに描く
ファンタジーエフェクト

フォトリアルに描くファンタジーとは

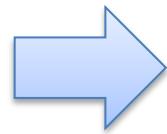
- FINAL FANTASY XVエフェクトの解釈
 - ファンタジックな表現においても
自然現象や物理現象をモデルとして描く



**自然な表現を目指すことで
フォトリアルとする**

フォトリアルの問題点

- ・ フォトリアルだけだと不足する演出
 - 過剰に演出したい場面
 - 記号的にエフェクトを表示したい場面



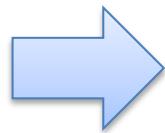
適度なフェイクを取り入れる

FINAL FANTASY XVエフェクトの作例



フォトトリアルに対する新たなアプローチ

- さらなるフォトトリアルを
目指すために実践したこと



**環境の影響により
動的に変化するエフェクト**

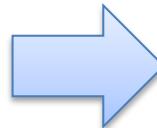
ゲーム内で変化する環境

- FINAL FANTASY XVにおける環境変化
 - 時間変化
 - 朝夕の切り替わり
 - 天候変化
 - 風の変化
 - 濃霧

環境変化を受けるエフェクトの作例

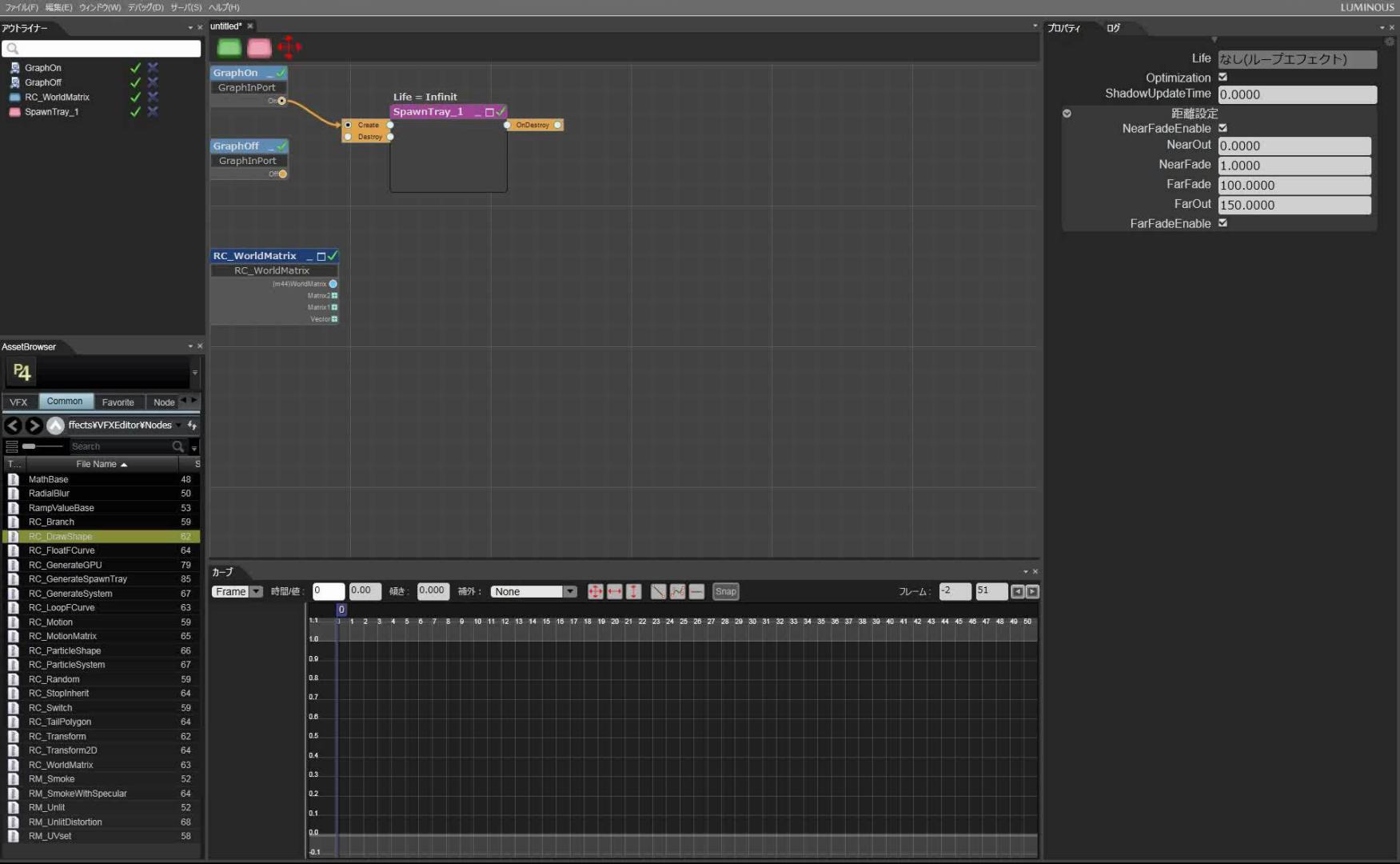


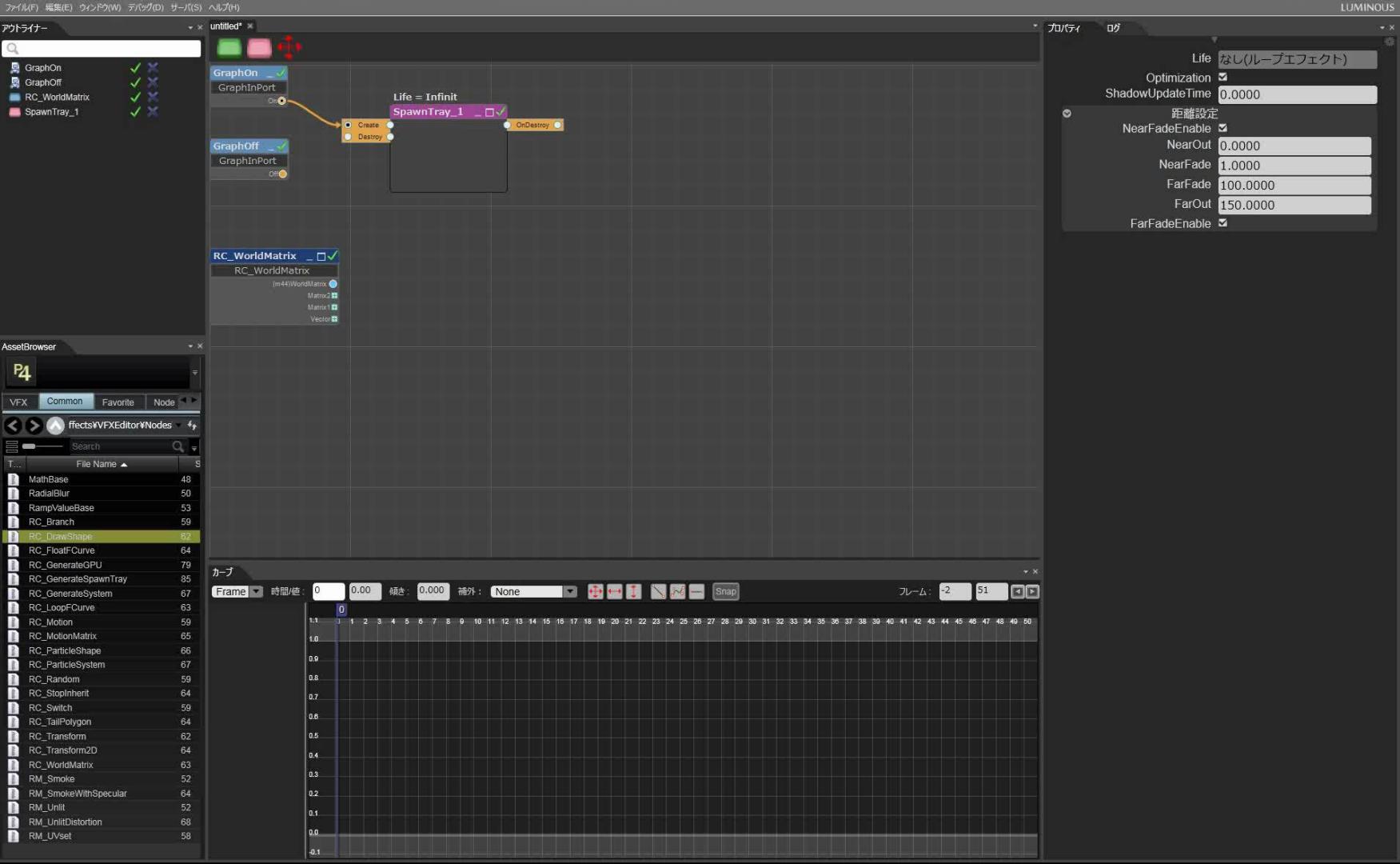
- FINAL FANTASY XVでは
 - ファンタジー表現を自然、物理現象の延長として捉える
 - フォトリアルが破綻しない程度のフェイク
 - 環境の影響で変化するエフェクト表現



**フォトリアルに描く
ファンタジーエフェクト**

エフェクト開発環境 LUMINOUS VFX EDITORの紹介





アトラーナー

- GraphOn ✓ ✗
- GraphOff ✓ ✗
- RC_WorldMatrix ✓ ✗
- SpawnTray_1 ✓ ✗

Untitled*

GraphOn ✓ GraphInPort

GraphOff ✓ GraphInPort

RC_WorldMatrix ✓ RC_WorldMatrix

SpawnTray_1

Life = Infinite

Create

Destroy

OnDestroy

AssetBrowser

P4

VFX Common Favorite Node

Effects\VFxEdition\Nodes

Search

File Name	Frame
MathBase	48
RadialBlur	50
RampValueBase	53
RC_Branch	59
RC_DrawShape	62
RC_FloatFCurve	64
RC_GenerateGPU	79
RC_GenerateSpawnTray	85
RC_GenerateSystem	67
RC_LoopFCurve	63
RC_Motion	59
RC_MotionMatrix	65
RC_ParticleShape	66
RC_ParticleSystem	67
RC_Random	59
RC_StopInherit	64
RC_Switch	59
RC_TailPolygon	64
RC_Transform	62
RC_Transform2D	64
RC_WorldMatrix	63
RM_Smoke	52
RM_SmokeWithSpecular	64
RM_Unit	52
RM_UnitDistortion	68
RM_UVSet	58

カーブ

Frame 時間/値: 0 0.00 傾き: 0.000 補外: None Snap フレーム: -2 51

0.00

0.000

None

0

1.1

1.0

0.9

0.8

0.7

0.6

0.5

0.4

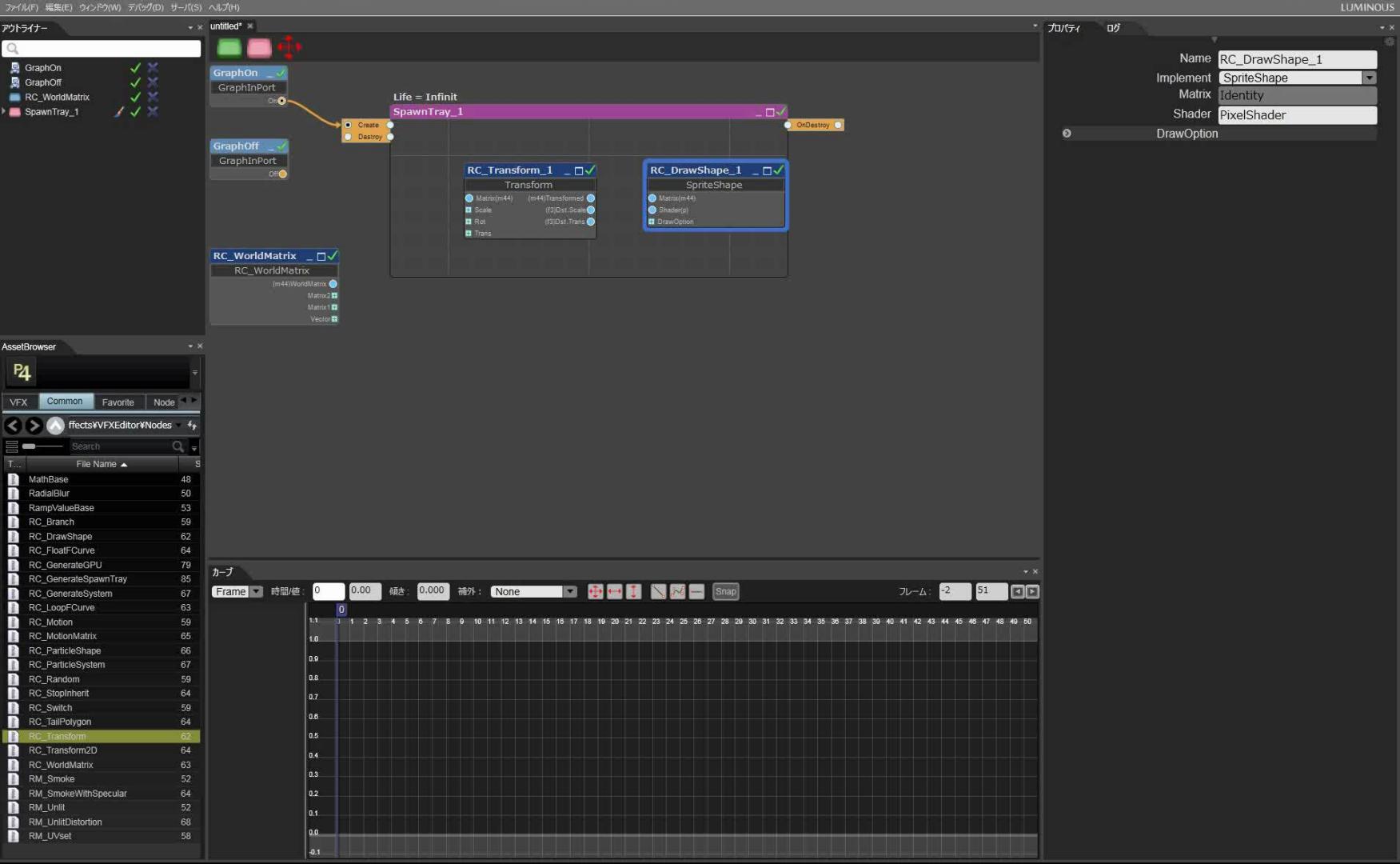
0.3

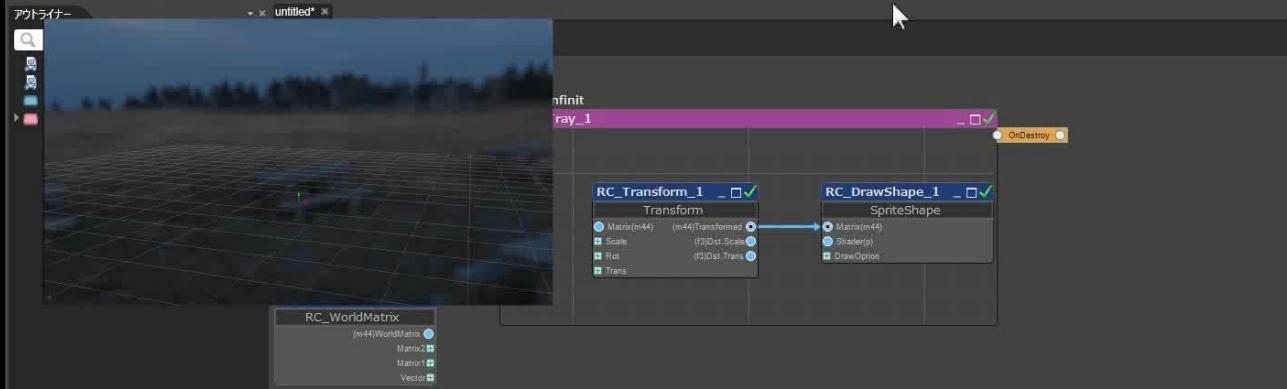
0.2

0.1

0.0

-0.1





P4

VFX Common Favorite Node

Search

File Name	Frame
MathBase	48
RadialBlur	50
RampValueBase	53
RC_Branch	59
RC_DrawShape	62
RC_FloatFCurve	64
RC_GenerateGPU	79
RC_GenerateSpawnTray	85
RC_GenerateSystem	67
RC_LoopFCurve	63
RC_Motion	59
RC_Matrix	65
RC_ParticleShape	66
RC_ParticleSystem	67
RC_Random	59
RC_StopInherit	64
RC_Switch	59
RC_TайлPolygon	64
RC_Transform	62
RC_Transform2D	64
RC_WorldMatrix	63
RM_Smoke	52
RM_SmokeWithSpecular	64
RM_Unit	52
RM_UnitDistortion	68
RM_UVset	58

カーブ

Frame 時間/値: 0 0.00 傾き: 0.000 補外: None Snap フレーム: -2 51

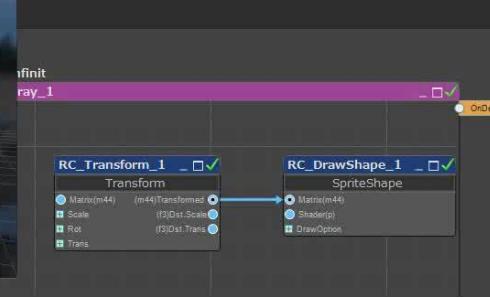
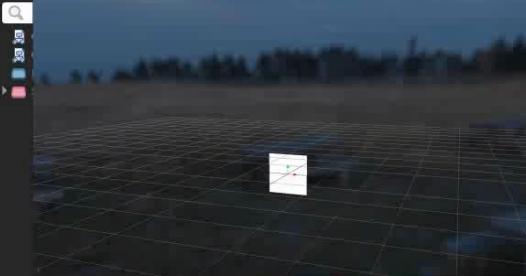
アトライナー

untitled*

プロパティ

ログ

...



AssetBrowser

P4

VFX Common Favorite Node

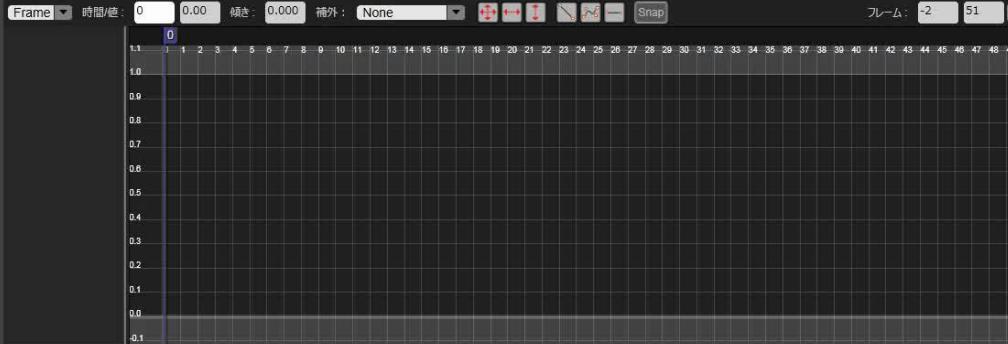
< > effects\VFXEditor\Nodes

Search

T... File Name ▲

MathBase	48
RadialBlur	50
RampValueBase	53
RC_Branch	59
RC_DrawShape	62
RC_FloatFCurve	64
RC_GenerateGPU	79
RC_GenerateSpawnTray	85
RC_GenerateSystem	67
RC_LoopFCurve	63
RC_Motion	59
RC_MotionMatrix	65
RC_ParticleShape	66
RC_ParticleSystem	67
RC_Random	59
RC_StopInherit	64
RC_Switch	59
RC_TайлPolygon	64
RC_Transform	62
RC_Transform2D	64
RC_WorldMatrix	63
RM_Smoke	52
RM_SmokeWithSpecular	64
RM_Unit	52
RM_UnitDistortion	68
RM_UVset	58

カーブ



アトラバー

Untitled*

プロパティ

ログ

ray_1

RC_Transform_1

Implement Transform

BillboardType None

SRTType SR

Matrix Identity

Scale

Scale X 3.0000

Y 1.0000

Z 1.0000

ScaleAll 1.0000

Rot

Trans

RC_Transform_1

Transform

Matrix(m44) < m44>Transformed

Scale

Rot

Trans

RC_DrawShape_1

SpriteShape

Matrix(m44)

Shader(s)

DrawOffset

RC_WorldMatrix

[m44]WorldMatrix

Matrix2

Matrix1

Vector

P4

VFX Common Favorite Node

effects\VFXEditor\Nodes

Search

File Name

MathBase 48

RadialBlur 50

RampValueBase 53

RC_Branch 59

RC_DrawShape 62

RC_FloatFCurve 64

RC_GenerateGPU 79

RC_GenerateSpawnTray 85

RC_GenerateSystem 67

RC_LoopFCurve 63

RC_Motion 59

RC_MotionMatrix 65

RC_ParticleShape 66

RC_ParticleSystem 67

RC_Random 59

RC_StopInherit 64

RC_Switch 59

RC_TailPolygon 64

RC_Transform 62

RC_Transform2D 64

RC_WorldMatrix 63

RM_Smoke 52

RM_SmokeWithSpecular 64

RM_Unit 52

RM_UnitDistortion 68

RM_UVSet 58

カーブ

Frame 時間/値: 0 0.00 傾き: 0.000 補外: None Snap フレーム: -2 51

1.1
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
-0.1

アトラバー

Untitled*

プロパティ

ログ

ray_1

RC_Transform_1

Implement Transform

BillboardType None

SRTType SR

Matrix Identity

Scale

Scale X 3.0000

Y 1.0000

Z 1.0000

ScaleAll 1.0000

Rot

Trans

RC_Transform_1

Transform

Matrix(m44) < m44>Transformed

Scale

Rot

Trans

RC_DrawShape_1

SpriteShape

Matrix(m44)

Shader(s)

DrawOffset

RC_WorldMatrix

[m44]WorldMatrix

Matrix2

Matrix1

Vector

AssetBrowser

P4

VFX Common Favorite Node

effects\VFXEditor\Nodes

Search

	File Name	Frame
MathBase	48	0
RadialBlur	50	0
RampValueBase	53	0
RC_Branch	59	0
RC_DrawShape	62	0
RC_FloatFCurve	64	0
RC_GenerateGPU	79	0
RC_GenerateSpawnTray	85	0
RC_GenerateSystem	67	0
RC_LoopFCurve	63	0
RC_Motion	59	0
RC_MotionMatrix	65	0
RC_ParticleShape	66	0
RC_ParticleSystem	67	0
RC_Random	59	0
RC_StopInherit	64	0
RC_Switch	59	0
RC_TailPolygon	64	0
RC_Transform	62	0
RC_Transform2D	64	0
RC_WorldMatrix	63	0
RM_Smoke	52	0
RM_SmokeWithSpecular	64	0
RM_Unit	52	0
RM_UnitDistortion	68	0
RM_UVSet	58	0

カーブ

Frame 時間/値: 0 0.00 傾き: 0.000 補外: None Snap フレーム: -2 51

アトラバー

Untitled*

プロパティ

ログ

ray_1

RC_Transform_1

Implement Transform

BillboardType None

SRTType SR

Matrix Identity

Scale

Scale X 3.0000

Y 1.0000

Z 1.0000

ScaleAll 1.0000

Rot

Trans

Trans

Loop Const

Thumbnail

Offset X 0.0000

Y 0.0000

Z 0.0000

AssetBrowser

P4

VFX Common Favorite Node

effects\VFXEditor\Nodes

Search

File Name

MathBase 48

RadialBlur 50

RampValueBase 53

RC_Branch 59

RC_DrawShape 62

RC_FloatFCurve 64

RC_GenerateGPU 79

RC_GenerateSpawnTray 85

RC_GenerateSystem 67

RC_LoopFCurve 63

RC_Motion 59

RC_MotionMatrix 65

RC_ParticleShape 66

RC_ParticleSystem 67

RC_Random 59

RC_StopInherit 64

RC_Switch 59

RC_TailPolygon 64

RC_Transform 62

RC_Transform2D 64

RC_WorldMatrix 63

RM_Smoke 52

RM_SmokeWithSpecular 64

RM_Unit 52

RM_UnitDistortion 68

RM_UVSet 58

カーブ

Frame 時間/値 0 0.00 傾き 0.000 補外: Const Snap フレーム: -2 51

RC_Transform_1

RC_Transform_1

Trans.Trans

0	1.0
1	0.9
2	0.8
3	0.7
4	0.6
5	0.5
6	0.4
7	0.3
8	0.2
9	0.1
10	0.0
11	-0.1
12	-0.2
13	-0.3
14	-0.4
15	-0.5
16	-0.6
17	-0.7
18	-0.8
19	-0.9
20	-1.0
21	-0.9
22	-0.8
23	-0.7
24	-0.6
25	-0.5
26	-0.4
27	-0.3
28	-0.2
29	-0.1
30	0.0
31	0.1
32	0.2
33	0.3
34	0.4
35	0.5
36	0.6
37	0.7
38	0.8
39	0.9
40	1.0
41	0.9
42	0.8
43	0.7
44	0.6
45	0.5
46	0.4
47	0.3
48	0.2
49	0.1
50	0.0

アトラバー

Untitled*

プロパティ

ログ

Name: RC_Transform_1
Implement: Transform
BillboardType: None
SRTType: SR
Matrix: Identity

Scale:
Scale X: 3.0000
Y: 1.0000
Z: 1.0000
ScaleAll: 1.0000

Rot:
Trans:
Trans:
Loop: Repeat
Thumbnail:
Offset X: 0.0000
Y: 0.0000
Z: 0.0000

AssetBrowser

P4

VFX Common Favorite Node

effects\VFXEditor\Nodes

Search

	File Name	Frame
MathBase	48	0
RadialBlur	50	0
RampValueBase	53	0
RC_Branch	59	0
RC_DrawShape	62	0
RC_FloatFCurve	64	0
RC_GenerateGPU	79	0
RC_GenerateSpawnTray	85	0
RC_GenerateSystem	67	0
RC_LoopFCurve	63	0
RC_Motion	59	0
RC_MotionMatrix	65	0
RC_ParticleShape	66	0
RC_ParticleSystem	67	0
RC_Random	59	0
RC_StopInherit	64	0
RC_Switch	59	0
RC_TailPolygon	64	0
RC_Transform	62	0
RC_Transform2D	64	0
RC_WorldMatrix	63	0
RM_Smoke	52	0
RM_SmokeWithSpecular	64	0
RM_Unit	52	0
RM_UnitDistortion	68	0
RM_UVSet	58	0

カーブ

Frame: 時間/値: 12.343 1.00 傾き: 0.000 補外: Repeat フレーム: -2 51

RC_Transform_1

RC_Transform

X: 1.0000
Y: 0.9000
Z: 0.8000

Y: 1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
-0.1

Frame: 時間/値: 12.343 1.00 傾き: 0.000 補外: Repeat フレーム: -2 51

RC_Transform_1

RC_Transform

X: 1.0000
Y: 0.9000
Z: 0.8000

Y: 1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
-0.1

アジェンダ

- 次世代機でのエフェクトデザインの一例
 - FFXV-EPISODE DUSCAE-におけるエフェクトデザイン
 - Luminous VFX Editorの紹介
- ノードベースエフェクトツールのメリット・効果的運用
 - ノードベースのメリット
 - Luminous VFX Editorのツール設計 ← アーティスト・TA・プログラマ向け
- 実現時の課題とその解決方法
 - 実装時の技術的課題
 - Luminous VFX Editorにおける解決方法

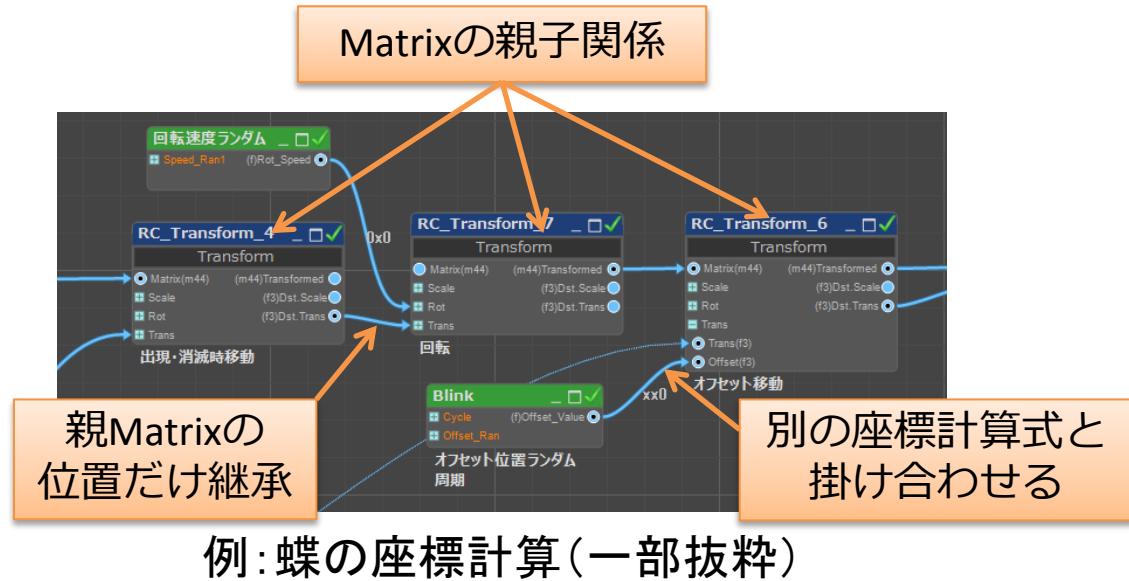
ノードベースのメリット

ノードベースのメリット

1. 柔軟性・自由度が高い
2. 外部とのやり取りが簡単
3. プログラマとアーティストの
作業分担がしやすい

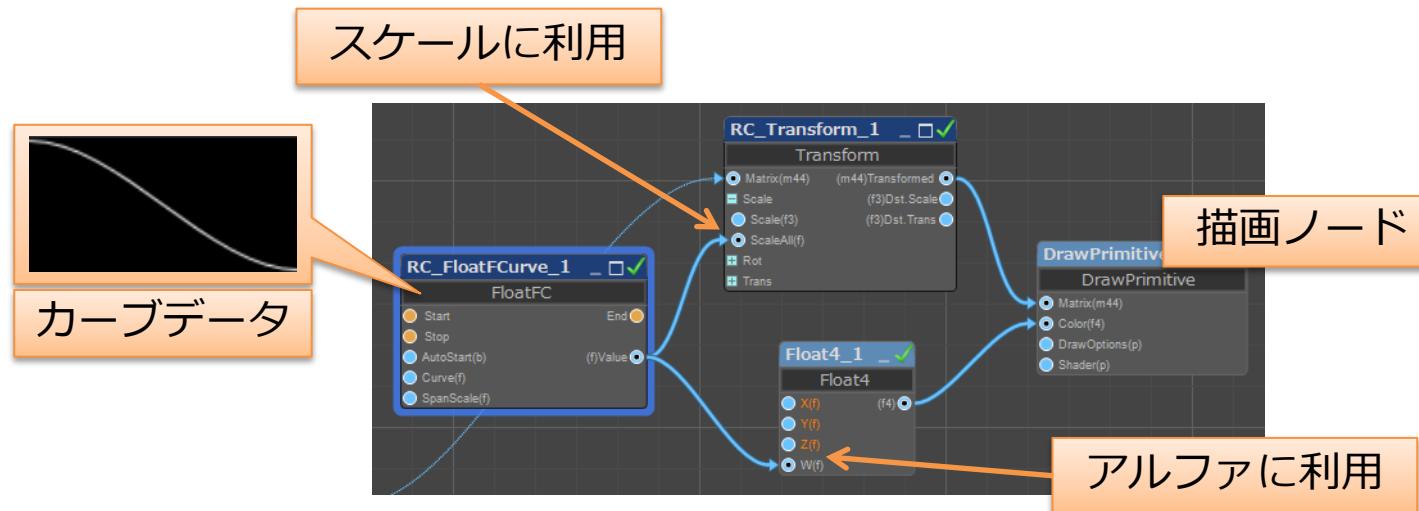
メリット1：柔軟性・自由度が高い

- つなぎ方次第でどんな計算もできる



メリット1：柔軟性・自由度が高い

- 例) Fcurveノード



例:スケールとアルファを同じカーブでフェードさせる

メリット2：外部とのやり取りが簡単

- 外部パラメータの例
 - 入力：時間、風の強さ、車のスピード、 ...
 - 出力：エフェクトの位置、描画領域、 ...
- 外部シグナルの例
 - 入力：フェードアウト開始タイミング、 ...
 - 出力：サウンドを鳴らすタイミング、 ...

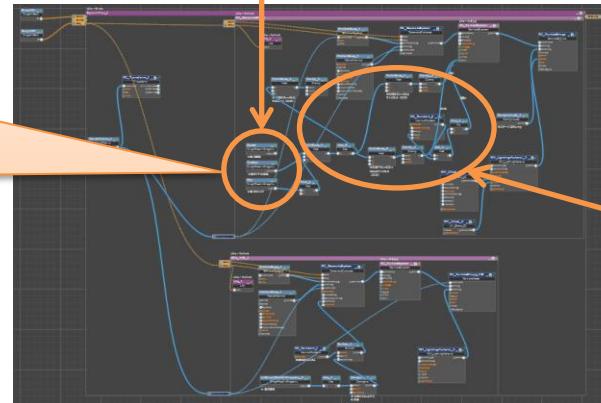
メリット2：外部とのやり取りが簡単

- 外部パラメータ/シグナル入出力ノード

外部から値を
もらうためのノード



グラフ上の必要に
なった箇所に置く

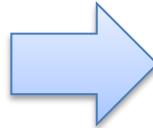


外部入力値を計算で
自由に加工できる

例：車の砂煙エフェクト

メリット3：作業分担がしやすい

- ・ プログラム更新なしで新機能を作れる
 - アーティストがプログラマに頼らず、新しいエフェクトを作れる
 - 試作しやすく、プログラマにとっても便利

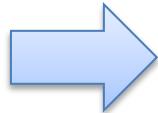


無駄なイテレーションが発生しない

メリット3：作業分担がしやすい

- 例) 揚陸艇が起こす風煙

- アーティスト要望
「煙を地面の傾き方向に
流したいけど、
起伏が激しいところでは
いい感じに無視したい」



プログラム更新なしにツール上で実装

ノードベースのメリット（まとめ）

1. 柔軟性・自由度が高い
2. 外部とのやり取りが簡単
3. プログラマとアーティストの
作業分担がしやすい

LUMINOUS VFX EDITORの ツール設計

ツール開発で目指したこと

ノードベースのメリットを活かした
柔軟性の高いツール

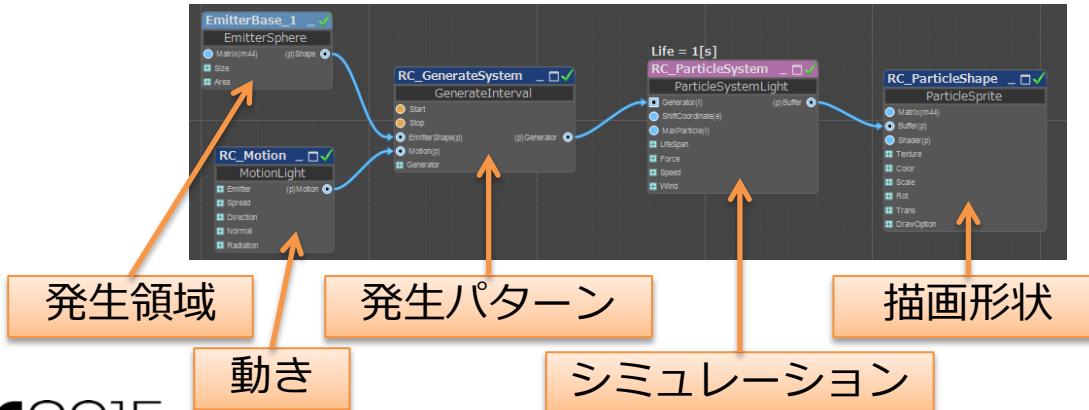
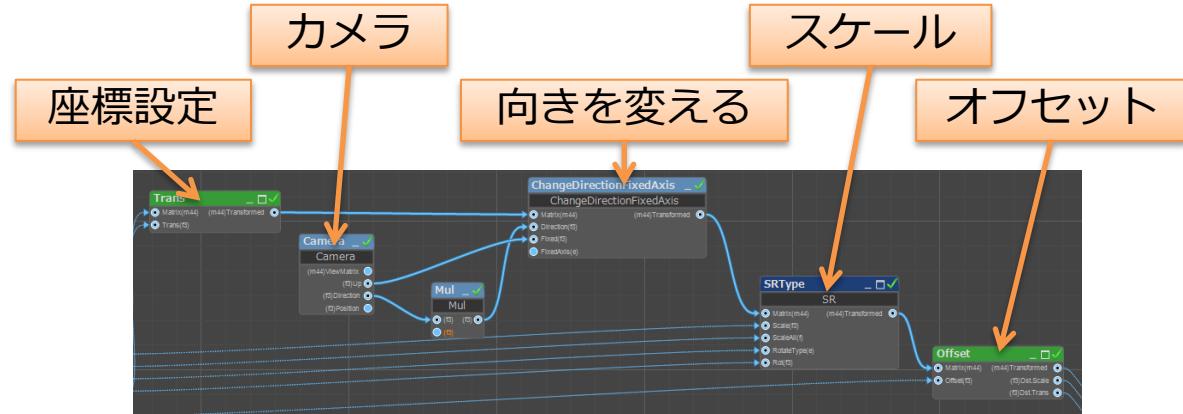
アーティストが学習しやすく
使いやすいツール

ノードベースのメリットを活かすためには

- ・個々のノードができるだけ小さく設計
 - いろんな場所で使いまわせるように
- ・大きな機能はノードを組み合わせて作る
 - 巨大なノード設計は大抵失敗する...
 - ・あるノードに入れた機能が他でも欲しくなる

ノード分離の例

Billboard →



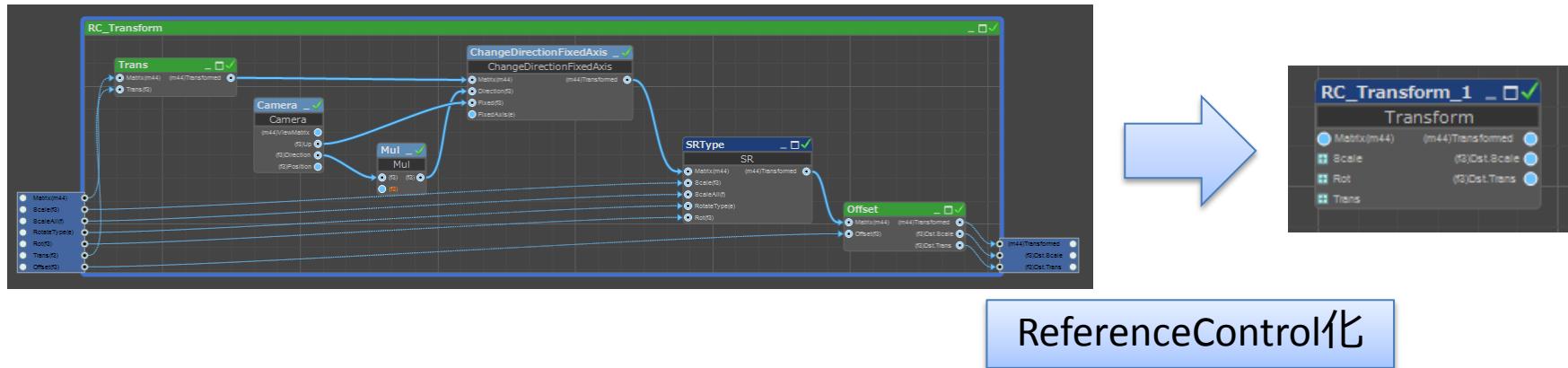
← Particle

課題

- ・個々のノードを細かくすると...
 - データを作る際の手間が増えてしまったり
 - ツールの学習が難しくなったり
 - たくさんのノードを覚えないといけない
 - ノード単体では動かない

課題解決のアプローチ

- 「Reference Control」を導入
 - 複数のノードを1つのノードとして扱う機能



Reference Controlのメリット

- ・自由度を選べる
 - 1つを大きくして座標計算から描画まで行う
 - 1つを細かくして自由度を高くする
 - アーティストが目的に合わせて作れる

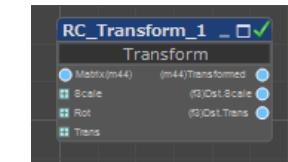
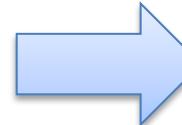
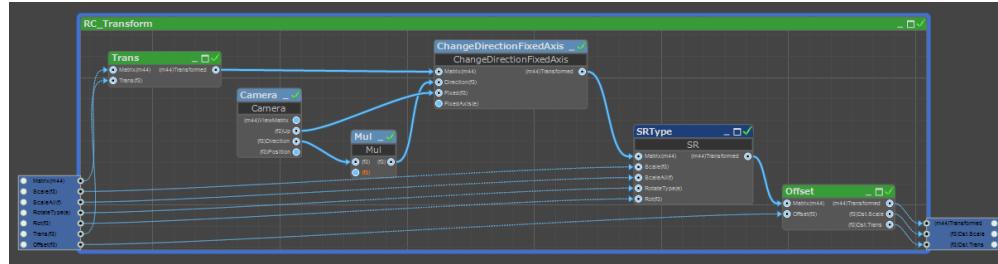


Reference Controlの機能を充実させていくことで
ツールの利便性が格段に上がる

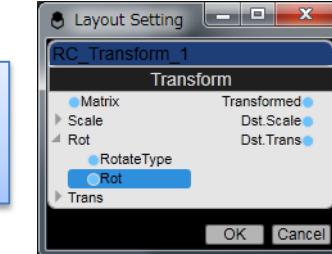
Reference Controlの機能拡充（その1）

- ReferenceControlの編集機能を追加

ツールのグラフ上で作成できる



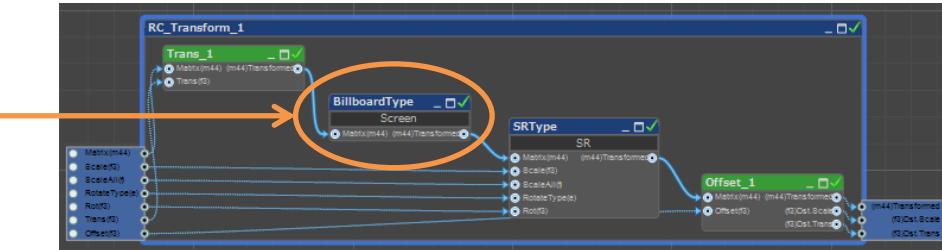
ポート編集機能
並び順, グループ化, 非表示



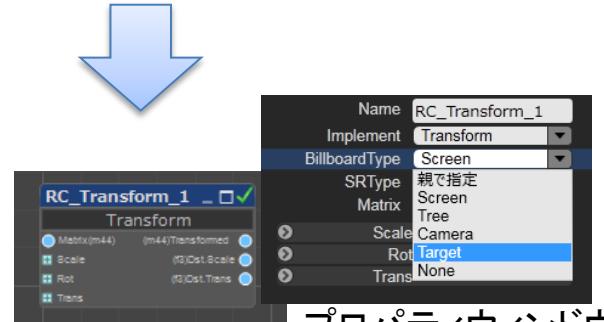
Reference Controlの機能拡充（その2）

- 「ベースノード」の概念を追加

向きを変える部分を
ベースノード化



中身のノードを選択して
動作を変えることができる



プロパティウィンドウ

結果

- ・ プログラマは
ノードを汎用性の高い設計にできる
- ・ アーティストは
ノードを扱いやすいように設計できる



ノードベースの利点を活かしつつ、
使いやすいツールになった！

アジェンダ

- 次世代機でのエフェクトデザインの一例
 - FFXV-EPISODE DUSCAE-におけるエフェクトデザイン
 - Luminous VFX Editorの紹介
- ノードベースエフェクトツールのメリット・効果的運用
 - ノードベースのメリット
 - Luminous VFX Editorのツール設計
- 実現時の課題とその解決方法
 - 実装時の技術的課題
 - Luminous VFX Editorにおける解決方法

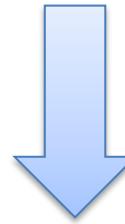
← プログラマ向け

実装時の技術的課題

ノードベースとリアルタイムエフェクト

- ノードベースのツール 자체は前例あり
⇒ 実装は問題なくできるはず

「リアルタイム」「エフェクト」を扱うノードベースツールはまだ少数



- Luminous VFX Editor開発時、リアルタイムエフェクト特有の課題に遭遇

実装時の課題概要

- ・ ノードベースのエフェクトツールの課題
 - エフェクト特有の寿命の概念と相性が悪い①
- ・ ランタイムを実装する上での課題
 - 重い（メモリコスト）②
 - 遅い（処理コスト）③

実装時の課題概要

- ・ ノードベースのエフェクトツールの課題
 - エフェクト特有の寿命の概念と相性が悪い
- ・ ランタイムを実装する上での課題
 - 重い（メモリコスト）
 - 遅い（処理コスト）

エフェクト特有の「寿命」の問題

- エフェクトの要素の特徴

- 短寿命
- 大量

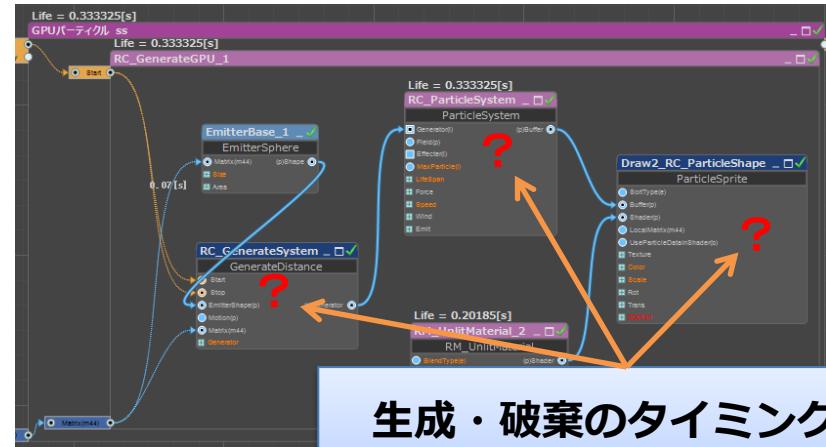
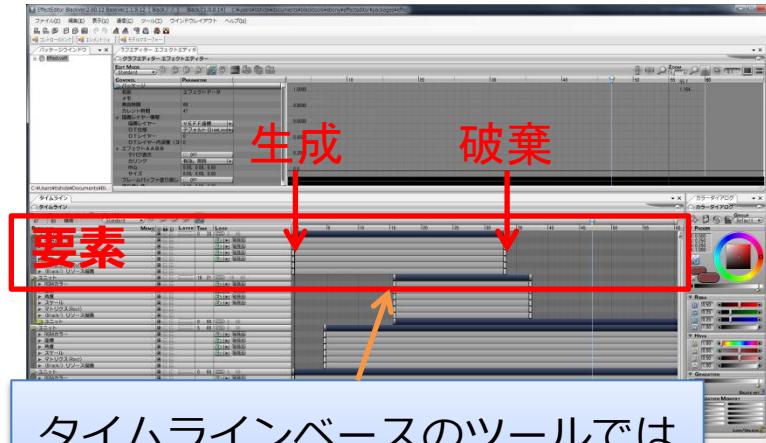
ゲームアセットの
中では特徴的

個々の火花は数フレー
ムで消えるが、大量



ノードベースは寿命の概念と相性が悪い

- エフェクトツールにおける寿命の表示は



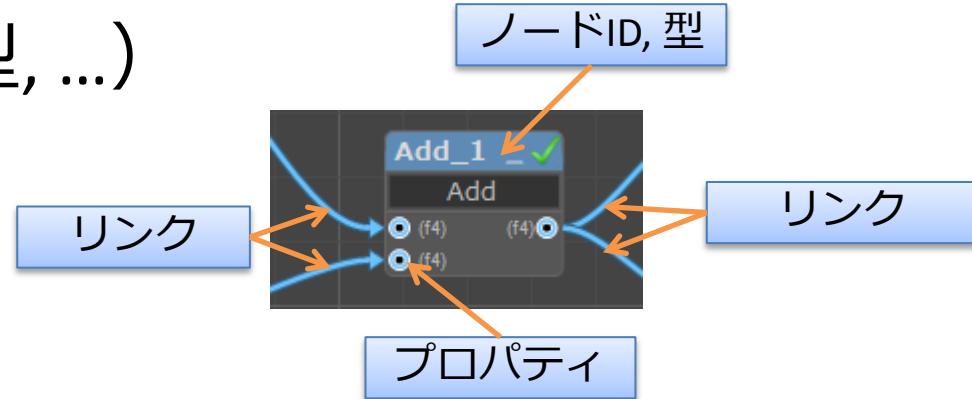
ノードベースでエフェクトは扱いにくい

実装時の課題概要

- ・ ノードベースのエフェクトツールの課題
 - エフェクト特有の寿命の概念と相性が悪い
- ・ ランタイムを実装する上での課題
 - 重い（メモリコスト）
 - 遅い（処理コスト）

ノードの消費メモリ

- ・ ノード1個に必要なメモリ量
 - ノード情報 (ID, 型, ...)
 - プロパティ
 - ポート
 - リンク情報
 - ...



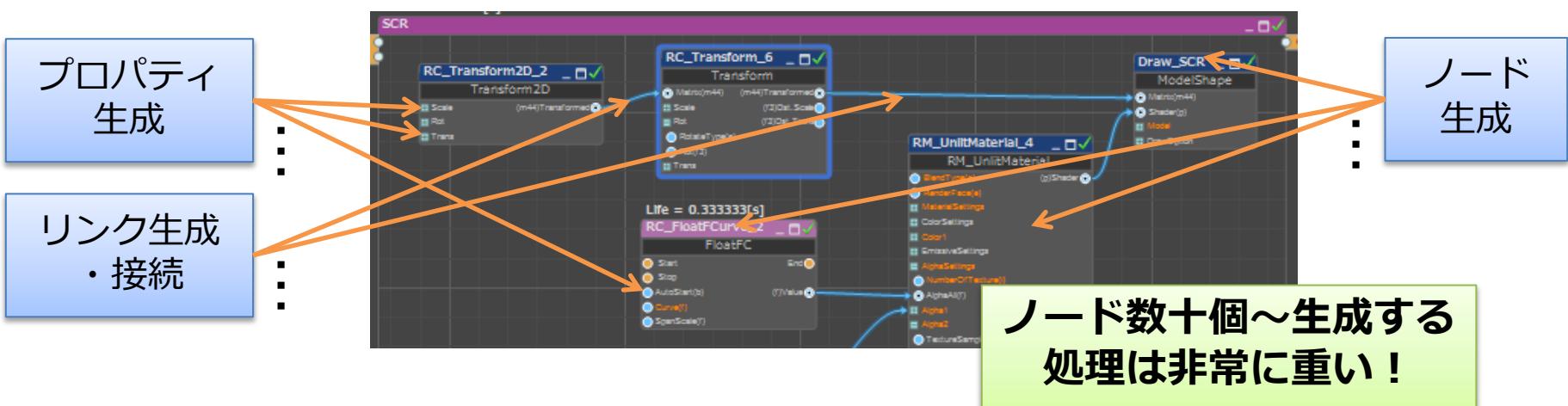
→ 高々足し算一個の情報には大きすぎる！

実装時の課題概要

- ・ ノードベースのエフェクトツールの課題
 - エフェクト特有の寿命の概念と相性が悪い
- ・ ランタイムを実装する上での課題
 - 重い（メモリコスト）
 - 遅い（処理コスト）

ノード生成の処理コスト

- ノード生成処理をナイーブに実装すると...



→ 生成はロード時に行いたいが...

リアルタイムエフェクトと動的生成



エフェクト内の要素は
その数もランダムに
決まるものがある



ノードベースのエフェクトツールでは
重いノード動的生成処理は避けられない

LUMINOUS VFX EDITOR における 解決方法

Luminous における解決策概要

課題

- ノードと寿命
- メモリコスト
- 処理コスト

解決策

- ノードのグループ化
 - 寿命管理
 - 生成破棄コスト削減
- データ構造の階層化
 - メモリコスト削減
- ノード群の式化
 - コスト削減

Luminous における解決策概要

課題

- ・ ノードと寿命
- ・ メモリコスト
- ・ 処理コスト

解決策

- ・ ノードのグループ化
 - 寿命管理
 - 生成破棄コスト削減
- データ構造の階層化
 - メモリコスト削減
- ノード群の式化
 - コスト削減

ノードのグループ化によるコスト減

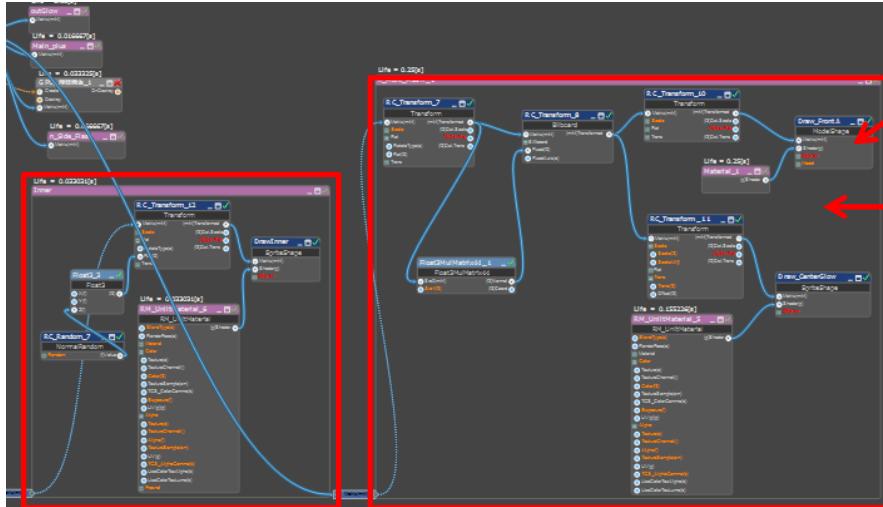
- ノードを個別に生成・削除は高コスト
⇒ ノードのグループ単位で生成・削除



- ノード個数分のメモリ確保・解放 ⇒ 1回に
- 解放時のリンク解除が不要に

ノードグループ化のトレードオフ

- ・ グループ単位で生成・削除ということは...



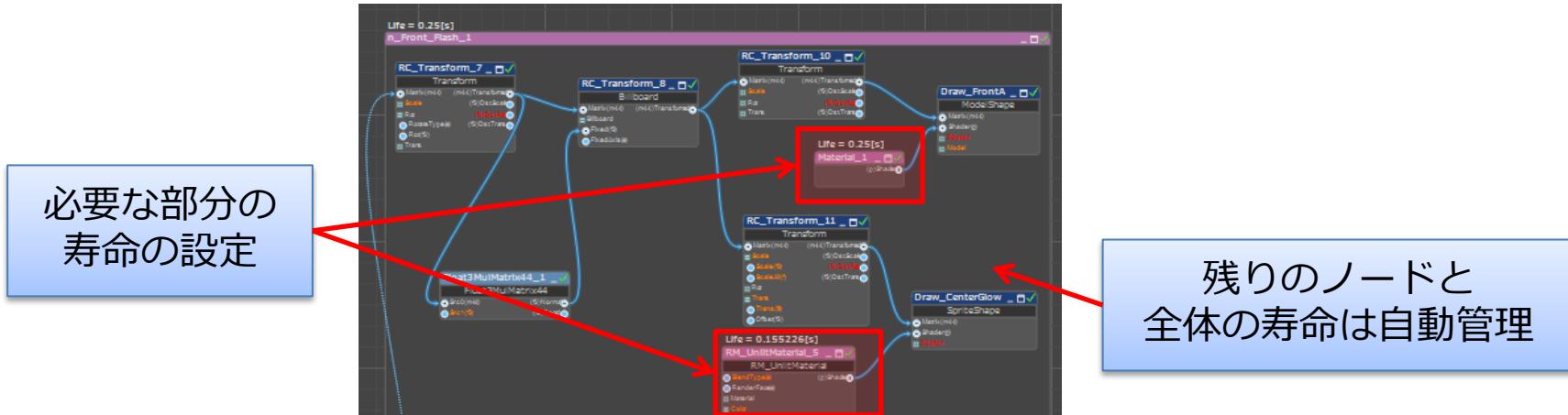
ノードグループ内で、個別に
ノードの生成・削除は不可

動的にノード種別を決定する
ような、動的グラフも不可

柔軟性低下だが、
トレードオフとして許容

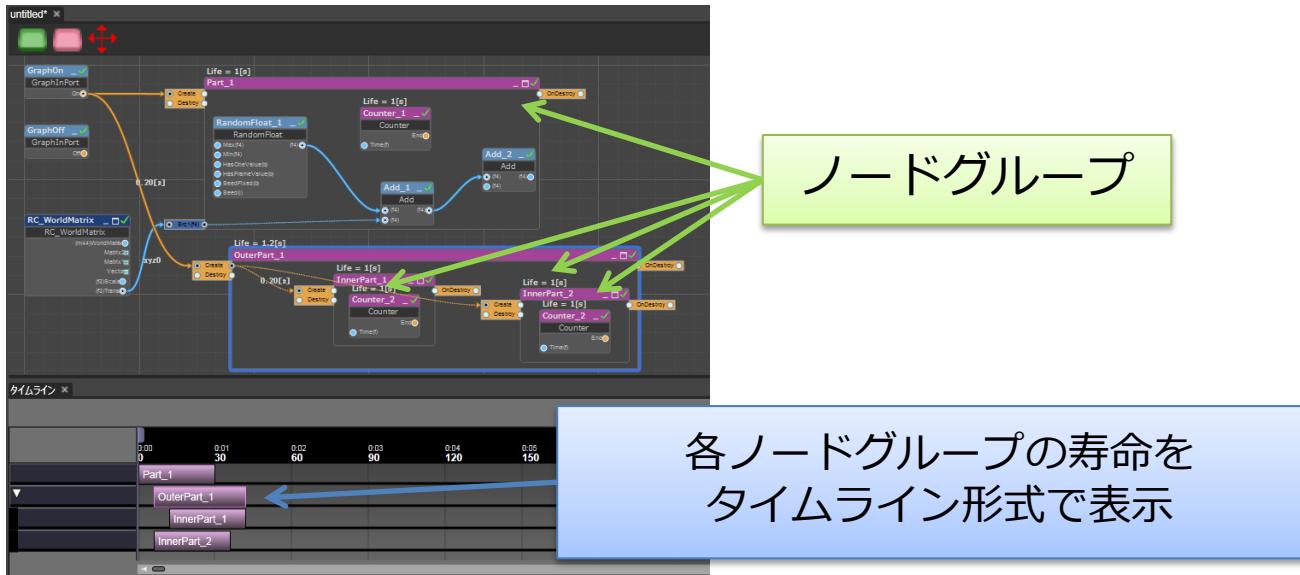
寿命管理のアプローチ

- 従来の寿命管理 ⇒ 個々の要素に設定
- Luminous VFX Editorでのアプローチは...



タイムラインビューによる寿命の可視化

- 寿命の分かりにくさへの対応
 - タイムラインビューで寿命を可視化



Luminous における解決策概要

課題

- ・ ノードと寿命
- ・ メモリコスト
- ・ 処理コスト

解決策

- ノードのグループ化
 - 寿命管理
 - 生成破棄コスト削減
- データ構造の階層化
 - メモリコスト削減
- ノード群の式化
 - コスト削減

VFX Editorのデータレイヤー

ノード

編集時グラフ

実行時グラフ



個数 1 :

N

:

N × M

⋮

⋮

データ階層化によるコスト減

ノード：編集時ノード：実行時ノード

⇒ 個数は 1 : N : N × M

- 実行時ノードに必要なデータは一部
 - 例: プロパティの初期値は編集時ノードでOK
- データを適切な階層に持たせ、コスト減
- 編集時グラフはロード時に構築できる



Luminous における解決策概要

課題

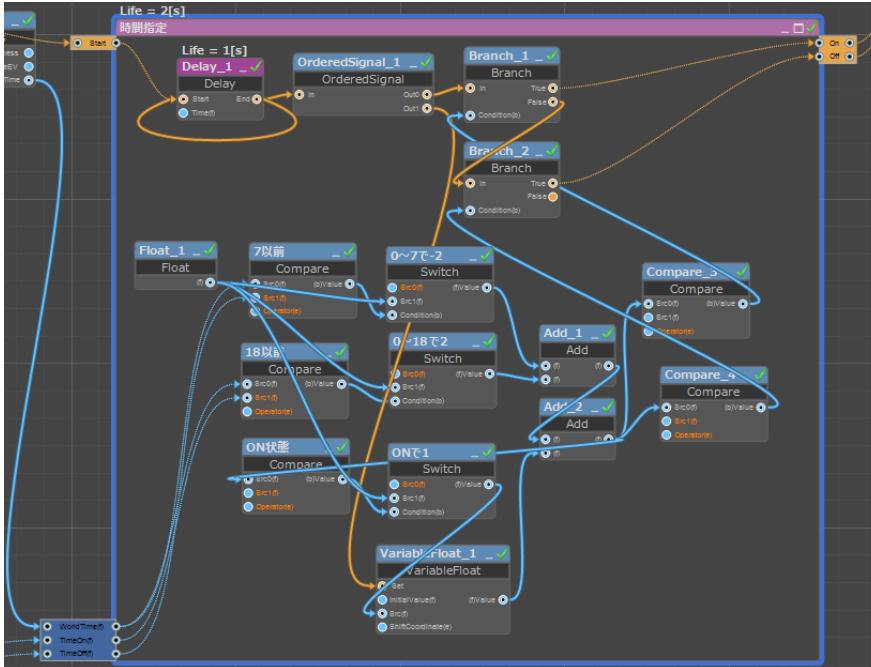
- ・ ノードと寿命
- ・ メモリコスト
- ・ 処理コスト

解決策

- ノードのグループ化
 - 寿命管理
 - 生成破棄コスト削減
- データ構造の階層化
 - メモリコスト削減
- ノード群の式化
 - コスト削減

ノードグループの式化

- 式を書くと大量のノードができてしまう

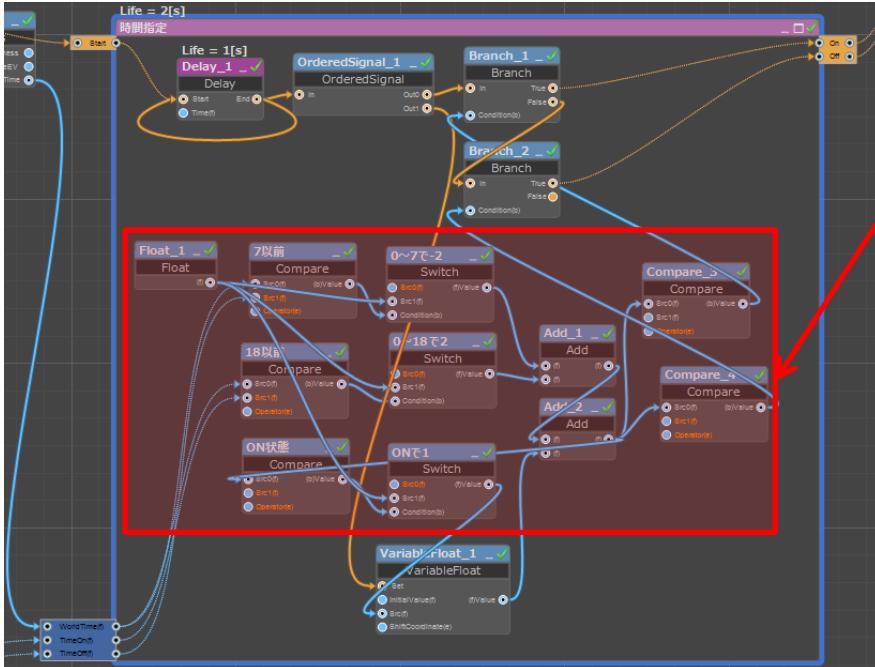


ノードの数が増えると…
• 消費メモリ増
• 処理時間増

例:蝶の出現ロジック(一部抜粋)

ノードグループの式化

- ツールの最適化機能により、計算ノード群を変換



仮想的な計算ノードとして扱う



実行時は1ノードになり
• 消費メモリ減
• 処理時間減

本セッションのまとめ

- 次世代機でのエフェクトデザインの一例
 - ⇒ フォトリアルに描くファンタジーエフェクト
 - ⇒ エフェクトが環境の影響を受ける別軸からのリアリティ表現
- ノードベースエフェクトツールのメリット・効果的運用
 - ⇒ 「柔軟性・自由度」「外部とのやり取り」「容易な作業分担」
 - ⇒ アーティストが学習しやすく使いやすいツール（ReferenceControl）
- 実現時の課題とその解決方法
 - 課題: 「寿命」「重い」「遅い」
 - 解決方法: 「グループ化」「データ構造階層化」「式化」

Q&A