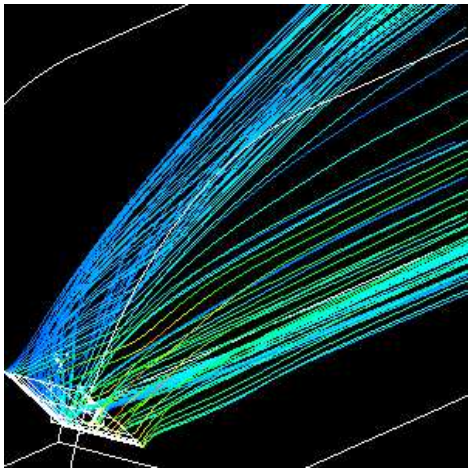
流線表示

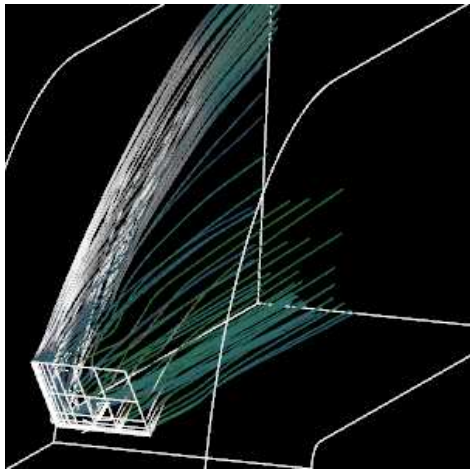
流線を生成・装飾するモジュールを紹介します。

流線モジュール

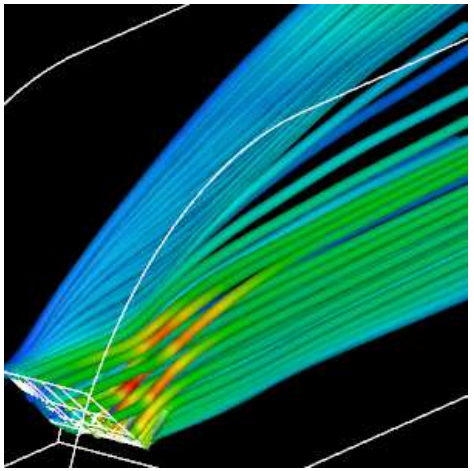
streamline	流線生成
Streamline	流線生成簡易版 （内部でstreamlineを利用）
mt_streamline	流線生成（マルチスレッド版）
FPlane	流線の発生点を指定
tube	チューブ表示
stream_color	流線の色付け
illminated line	イルミネーション表示
make_node_data	座標値やある基点からの距離、ID 、ランダム値を生成する



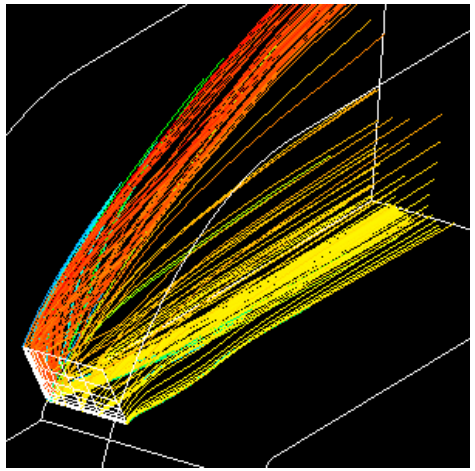
標準



イルミネーション表示



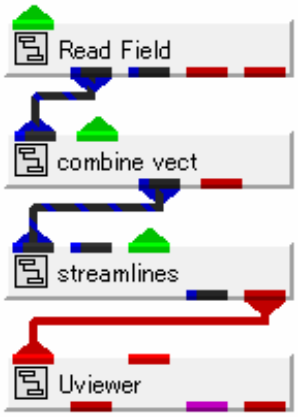
チューブ表示



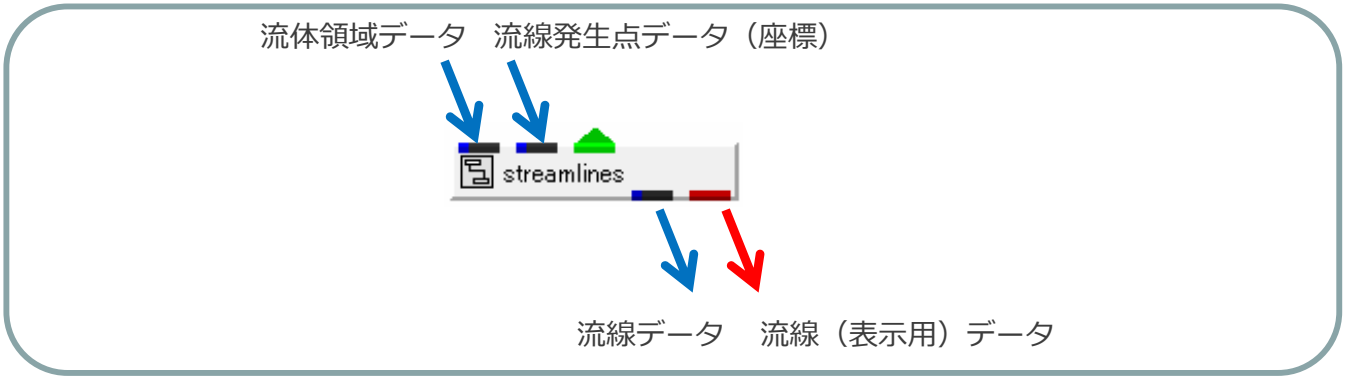
開始位置による色付け

※各モジュールの詳細はモジュールリファレンスを参照下さい。
※strema_colorは、Ver.8.1以降でご利用いただけます。

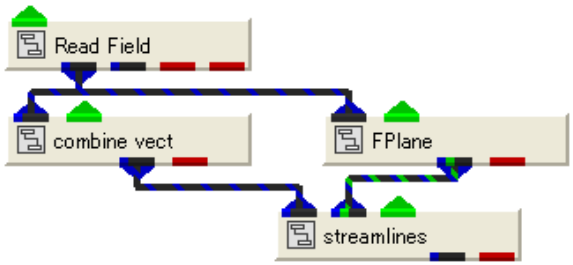
基本的な接続方法

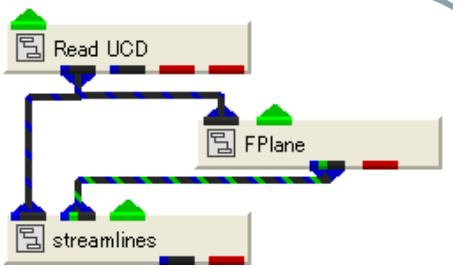


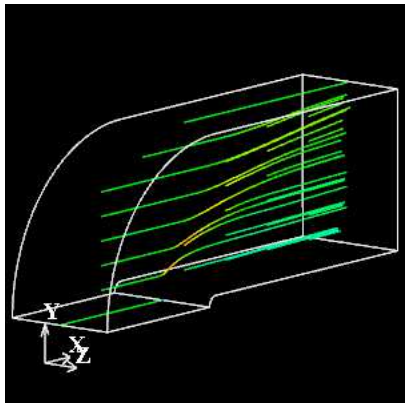
ここでは streamline モジュールを例として紹介します。（マルチスレッドモジュールのmt_streamlineも同様の接続で利用できます）入出力ポートの種類は以下の通りです。

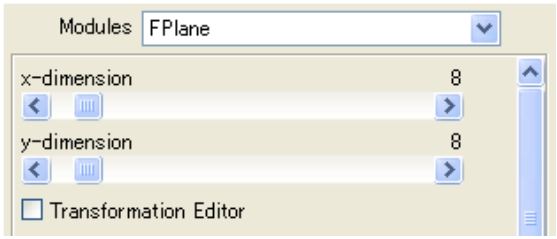


第2入力ポート（右側）に接続したデータの各座標が流線の発生ポイントになります。座標のみ利用するため、いろいろなモジュールの出力を接続できます。例えば、データ領域のZ値の中間に面を生成するFPlaneを接続すると以下のようになります。





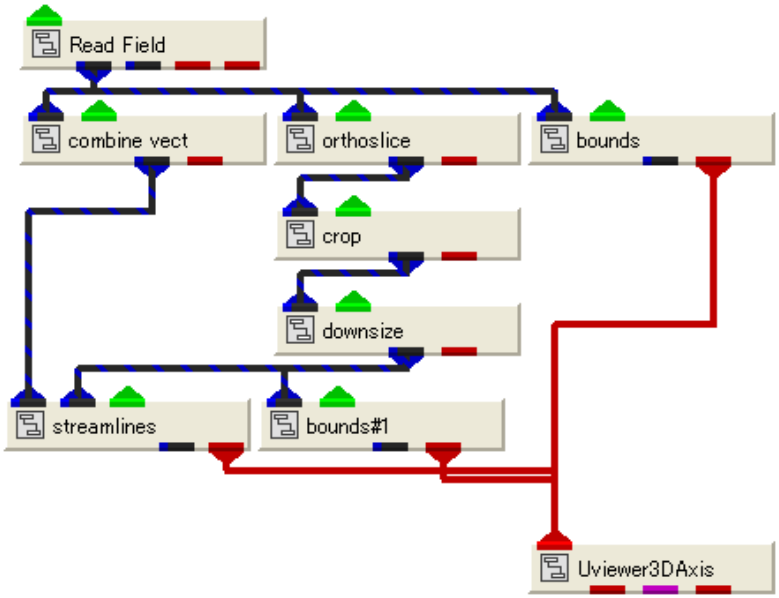


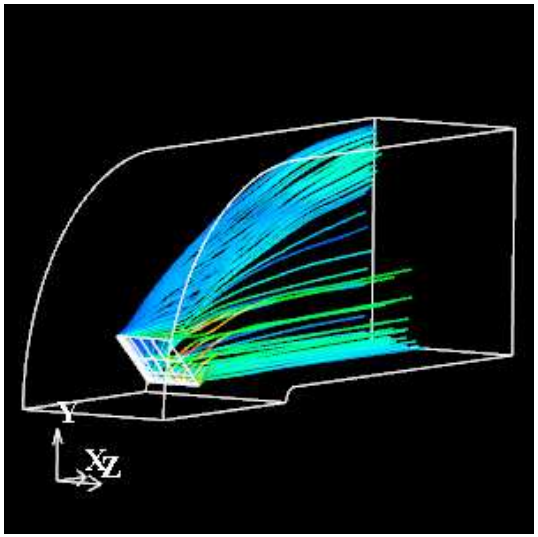


FPlaneのパラメータ（ここでは、8x8のメッシュを生成し、各格子点から流線が発生させている）

参考：
流線やベクトルなど向きを表すものはベクトルコンポーネントを利用します。構造格子型フォーマット（Field）ではベクトルコンポーネントが定義できないため、combine_vectモジュールを利用してExpress上で作成します。非構造格子型（UCD）フォーマットでベクトル成分をコンポーネントとして定義した場合、combine_vectは不要です。combine_vectモジュールのパラメータでベクトルを構成する成分を選択して下さい。

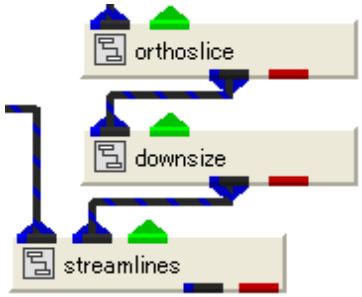
接続例



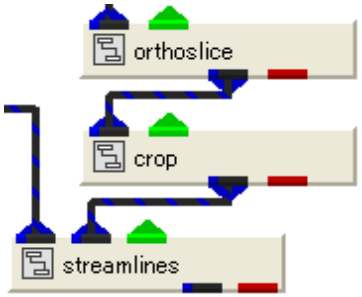


参考：
利用データ：bluntfin.fld
読み込んだデータの格子位置の一部から流線を発生

構造格子データでは格子断面から発生させることもできます。格子断面を抽出するorthoslice、格子領域を抽出するcrop、間引きするdownsizeなど複数のモジュールを組み合わせると流線の発生点を調整できます。

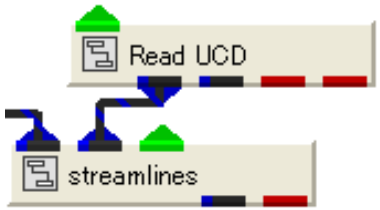


間引きして発生する流線数を減らす



流線を発生する領域を指定する

流線を発生させる座標が決まっている場合、構造格子（Field）や非構造格子（UCD）として離散データを作成し、以下のように接続することもできます。



流線の発生座標を直接指定したい時はRead_XXXの出力を直接接続する

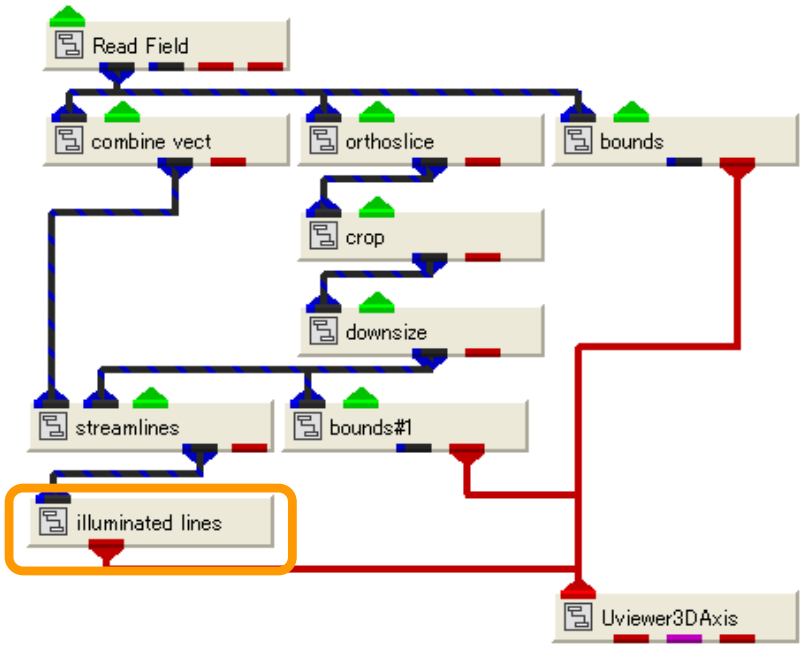
この他、等値面（isosurface）の出力など、青いポートを streamline の入力ポート（右側）に接続すれば流線の発生点になります。

※別資料「断面抽出モジュール」「領域抽出モジュール」の出力を利用することも可能。

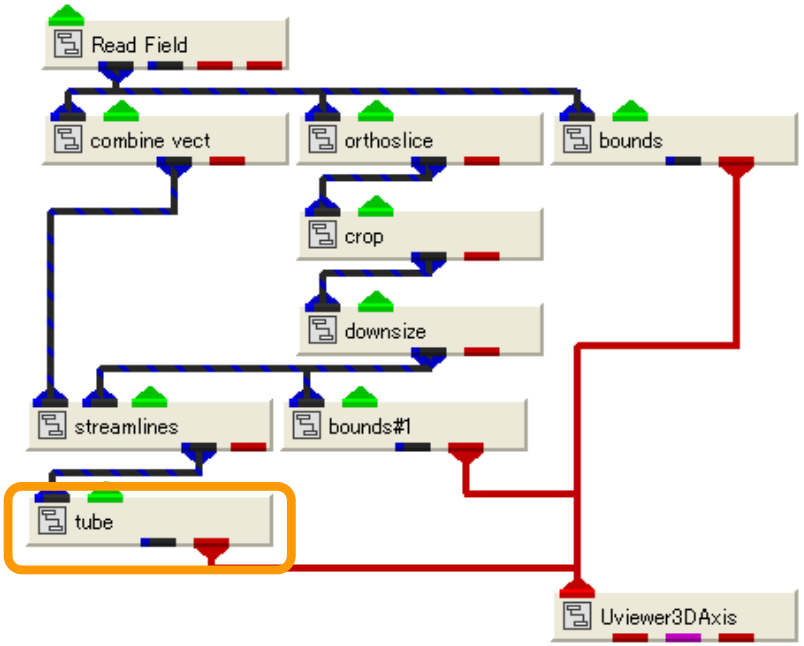
流線の装飾

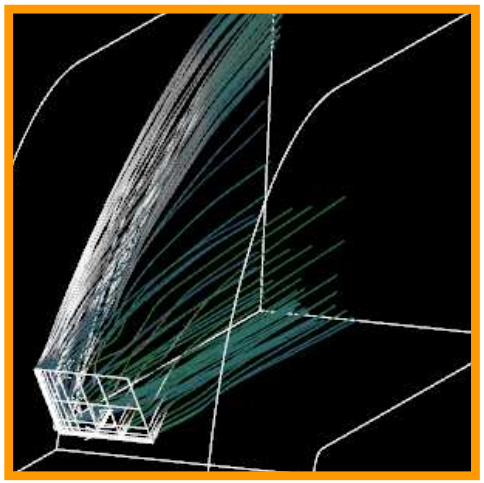
stramline の出力はベクトルの大きさで色付けされます。illuminated line、tube、stream_colorを併用すると流線の装飾ができます。

イルミネーション表示

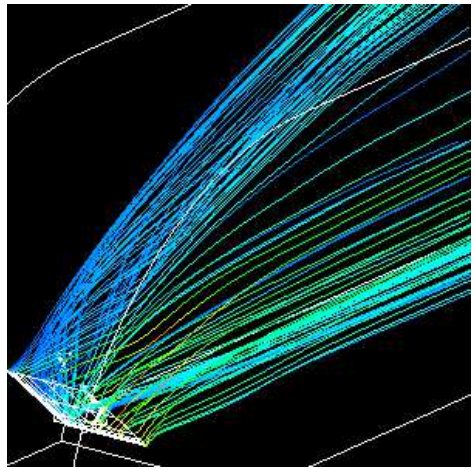


流線をチューブ形状で表示

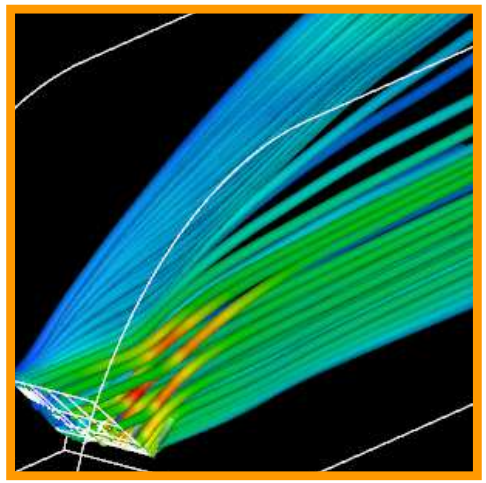




illuminated lines を利用



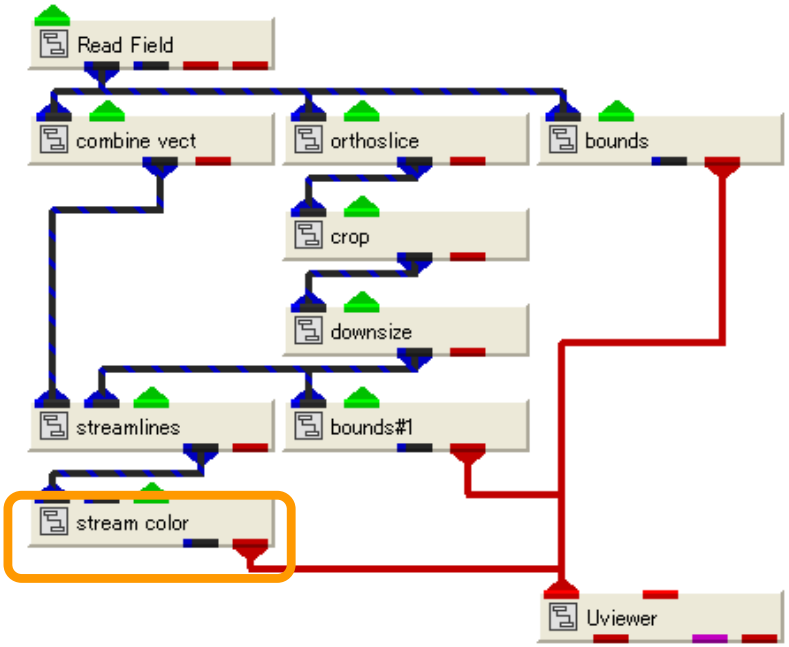
streamline オリジナル



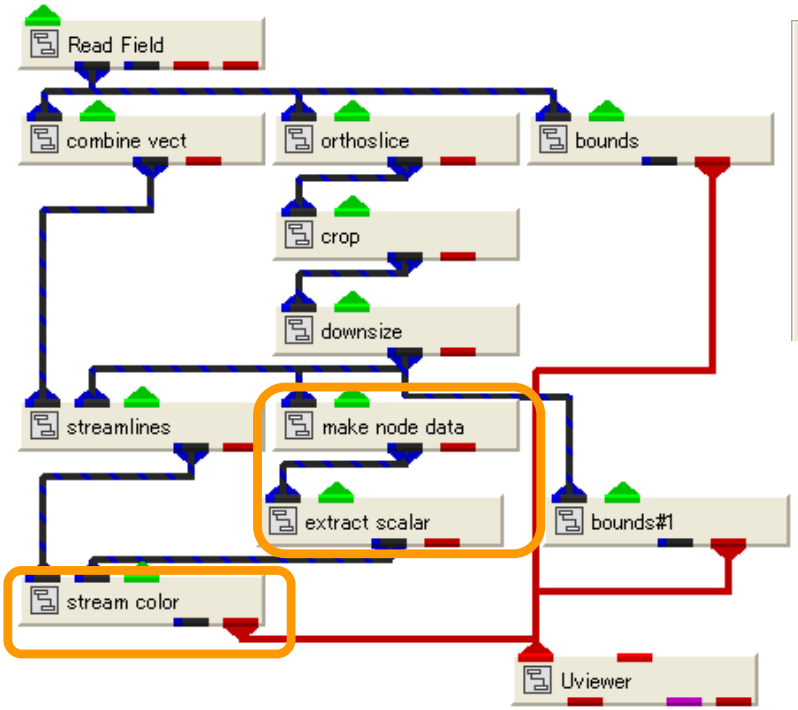
tube を利用

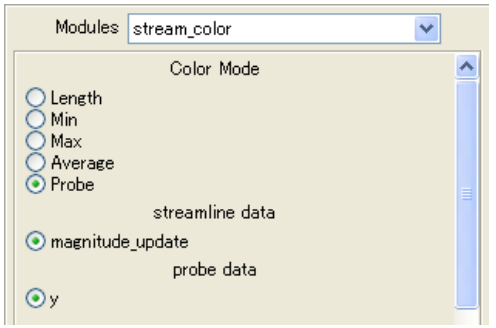
流線の装飾

流線の属性による色付け



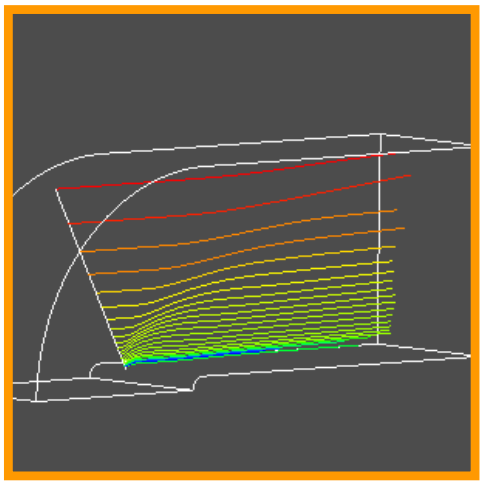
開始位置の値による色付け



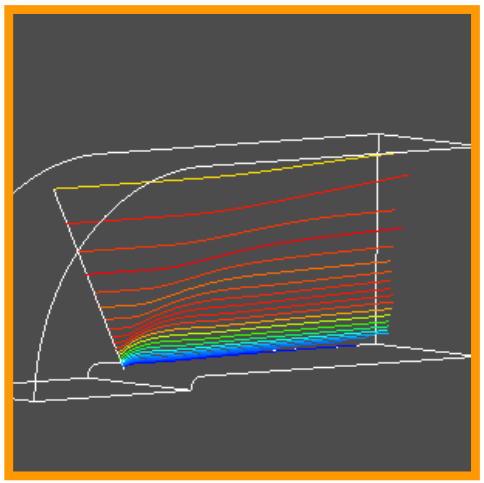


stream_colorパラメータ

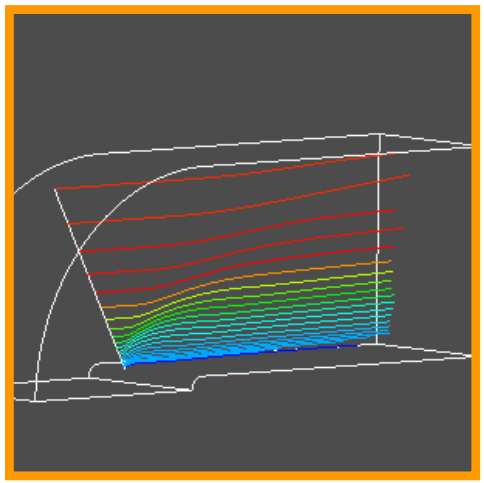
※ make_node_dataで座標を値として持たせ、この値で流線の色を指定しています。



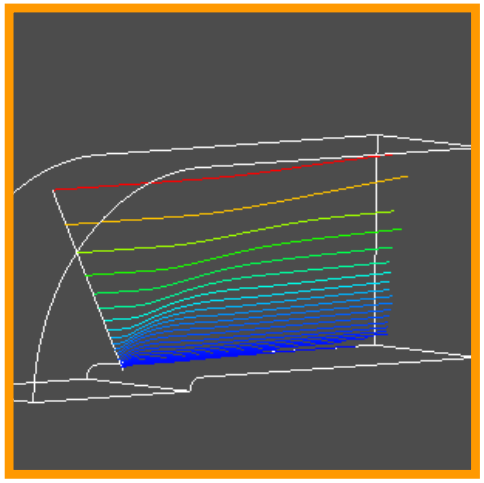
流線の長さによる色付け
（赤：長い、青：短い）



各流線上の一番小さいベクトル値で色付け



各流線上の一番大きなベクトル値で色付け



開始位置の値による色付け
（ここではZ座標を利用）

※配色が正しく表示されない場合、stream_colorを再インスタンスして利用して下さい。