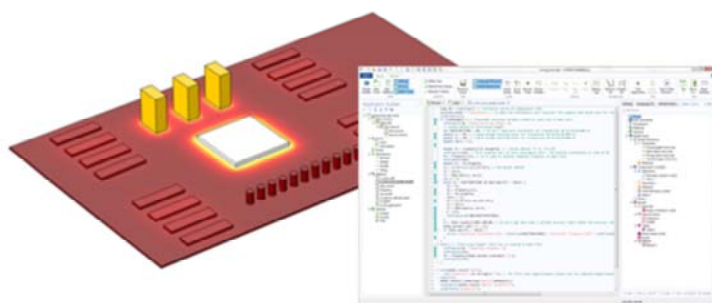


# Automating Your Modeling Workflow With Model Methods in Comsol Multiphysics®

Comsol Multiphysics®のモデルメソッドによるモデリングの自動化について



*Bjorn Sjodin*  
**COMSOL, Inc.**

ビヨン ショディン  
COMSOL社



# Application Methods vs. Model Methods

アプリケーションメソッドとモデルメソッドの違い

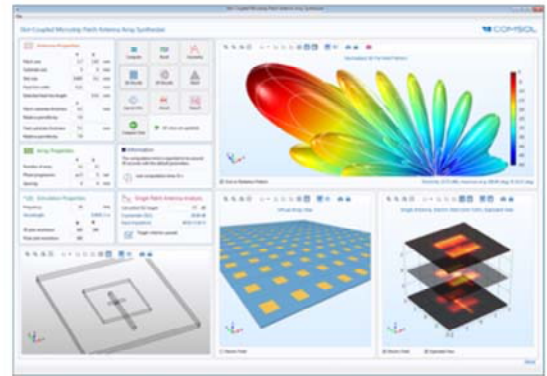
- **Application Method**    アプリケーションメソッド
  - Used to enhance the functionality of an app    アプリの機能強化
  - Doesn't change the original model in the MPH file    Mphファイルの元モデルを変更不可
  - Changes a copy of the original model    元モデルのコピーを変更
- **Model Method**    モデルメソッド
  - Used to enhance the functionality of a model    モデルの機能強化
  - Changes the original/current model    元/現行モデルの変更

# What are COMSOL Applications?

COMSOLアプリケーションとは

- Application = Model + User Interface
- Can be password protected
- Can configure double-clicking MPH file to
  - Edit app
  - Run app

アプリケーション=モデル+ユーザーインターフェース  
パスワードをつけることができる  
Mphファイルのダブルクリックでアプリの編集・実行が可能

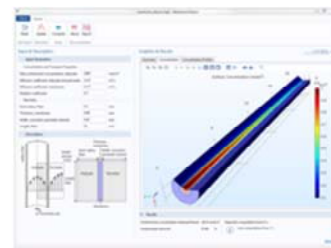


COMSOL

# Why Create Simulation Apps?

シミュレーション用のアプリをなぜ作るのか

- Sophisticated way of parameterizing your model モデルをパラメタコントロールする優れた方法を提供
  - More powerful よりパワフルに利用できる
  - Easier to pick up at a later time あとで利用するとき簡単
- Easy to share with colleagues and customers who are not COMSOL Multiphysics users 同僚とシェアしたりCOMSOLユーザーでないお客とのシェアが簡単にできる
  - For packaged simulations シミュレーションのパッケージ化
  - For viewing results ビューアーにもなる
- Link to legacy code in C, Java, FORTRAN
- Get an additional "solver license". Just create a simple app and solve with COMSOL Server.



C, Java, FORTRANとのリンク  
ソルバーライセンスを取得し、簡単なアプリ作成とCOMSOLサーバーで計算

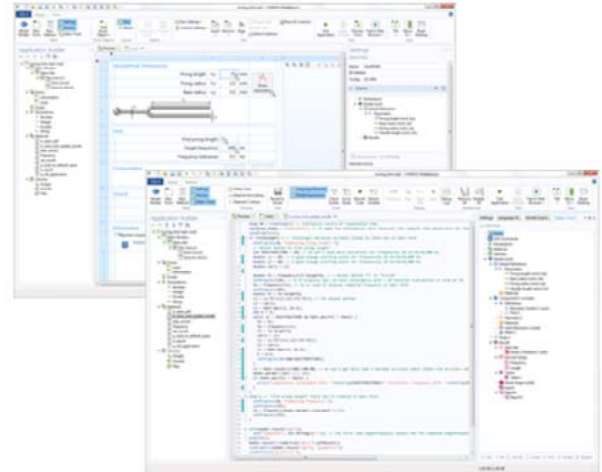
COMSOL

# Two Editors

二種類のエディター(編集機能)

- The Form Editor
  - For user interface design
- The Method Editor
  - For programming

フォームエディター  
ユーザーインターフェースのデザイン用  
メソッドエディター  
プログラミング用



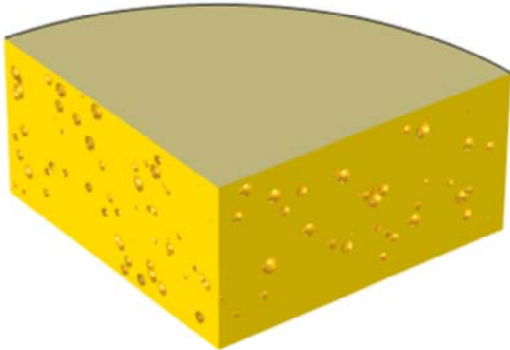
COMSOL

# Some Examples of Using Methods

メソッドの利用例

# Geometry

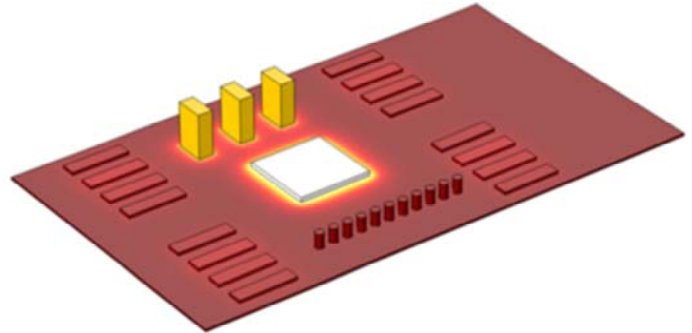
ジオメトリ作成への適用例



COMSOL BLOG

How to Create a Randomized Geometry Using Model Methods

モデルメソッド利用による乱数を使ったジオメトリ作成



COMSOL BLOG

Automate Model Preprocessing with the Application Builder

アプリケーションビルダでのモデル前処理の自動化



# Physics and Study Choice

フィジックスとスタディの選択への適用例

- Laminar or turbulent flow? 層流か乱流かの選択をしたい
- Small or large deformation solid mechanics? 固体力学で、微小変形か大変形かの選択をしたい
- Custom Optimization Algorithm 最適化アルゴリズムのカスタマイズ


COMSOL BLOG

How to Automate Physics Choices and Studies  
Using Model Methods

モデルメソッドを利用してフィジックスやスタディの選択を自動化する方法

 COMSOL





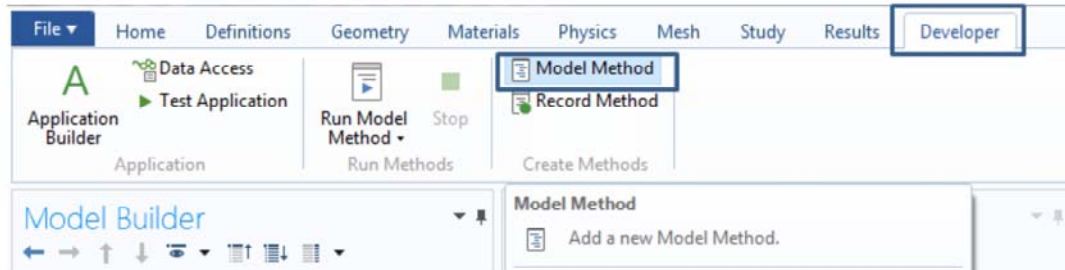
# Model Method Demo: an efficient solver sequence for elastoplasticity

モデルメソッドのデモ  
弾塑性体のための効率よいソルバシーケンスの設定



# Step 1: Add a Model Method

ステップ1 モデルメソッドを追加します。



# Step 2: Write the Method

ステップ2. メソッドを記述します。

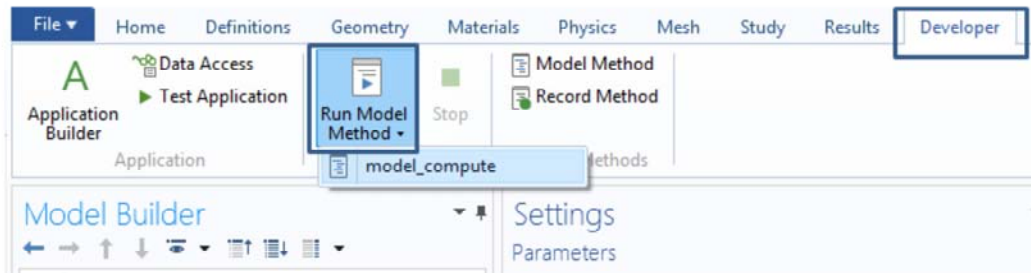
The screenshot displays the COMSOL Multiphysics software interface. The top ribbon is set to the 'Method' tab, with the subtitle 'ステップ2. メソッドを記述します。' (Step 2: Describe the method). The ribbon includes sections for 'Main' (Model Builder, New Form, New Method), 'Libraries' (Utility Class, Java External Java Library, External C Library), 'Edit' (Revert to Saved), and 'Code' (Language Elements, Model Expressions, Check Syntax, Go to Node, Record Code, Use Shortcut, Create Local Variable). The 'Language Elements' and 'Editor Tools' buttons are highlighted with blue boxes. On the left, the 'Application Builder' tree shows the project structure for 'elastoplastic\_plate.mph', with 'Methods' and 'model\_compute' highlighted by red and blue boxes respectively. The central code editor shows the following code for 'model\_compute':

```
1 //Run elastic analysis
2 model.study("std1").run();
3 //Compute the maximum stress in study 1
4 model.result().numerical("max1").run();
5 //Compare allowable and computed stresses
6 double VmMax, VmAllow, StressRatio, p_lower, p_next;
7 VmMax = model.result().numerical("max1").getReal()[0][0];
8 VmAllow = model.param().evaluate("SY_scaled");
9 StressRatio = VmMax/VmAllow;
10 //Do elastoplasticity if necessary
11 if (StressRatio > 1) {
12     p_lower = 1/StressRatio;
13     model.param().set("p_lower", p_lower);
14     p_next = Math.ceil(p_lower/0.05)*0.05;
15     model.param().set("p_next", p_next);
16     model.study("std2").run();
17 }
```

A blue arrow points to the closing brace of the 'if' statement on line 17. On the right, the 'Language Elements' pane is open, showing a search filter and a list of 'Language constructs' including Array operations, Block statements, Conversions, Server file handling utilities, External and utility libraries, User interface, Utility functions, and Variables.

# Step 3: Run the Model Method

ステップ3. モデルメソッドを実行します。



COMSOL

# Accessing Mesh & Solution Data

メッシュデータと解データの入手方法(get)

```
double[][] d_Vtx = model.mesh("mesh1").getVertex();  
  
int numTets = model.mesh("mesh1").getNumElem("tet");  
  
int numPyrs = model.mesh("mesh1").getNumElem("pyr");  
  
int[][] Ei = model.mesh("mesh1").getElem("tet");
```

## COMSOL BLOG

Exporting Meshes and Solutions Using the  
Application Builder

アプリケーションビルダを利用した  
メッシュと解のエクスポート

## COMSOL Multiphysics

Programming Reference Manual

COMSOL Multiphysics  
プログラミング リファレンス マニュアル





---

# Thank You!

ご清聴ありがとうございました。

 COMSOL