

Load Design ユーザーズマニュアル

Load Design version 0.8



本ソフトウェアは開発途中 (未完成) のソフトウェアです。動作確認などは十分されていません。いかなる保証もいたしません。また、本書の内容も、十分に確認されていません。間違いなどが含まれていることがあります。

山内祐一

Copyright © Yuichi Yamauchi. All rights reserved.

本書の一部または全部を再配布することはできません。また、ソフトウェア Load Design は同梱のソフトウェア使用許諾契約書に記載の条件のもとでお使いください。

Yuichi Yamauchi (山内祐一)

www.load-design.com

用語

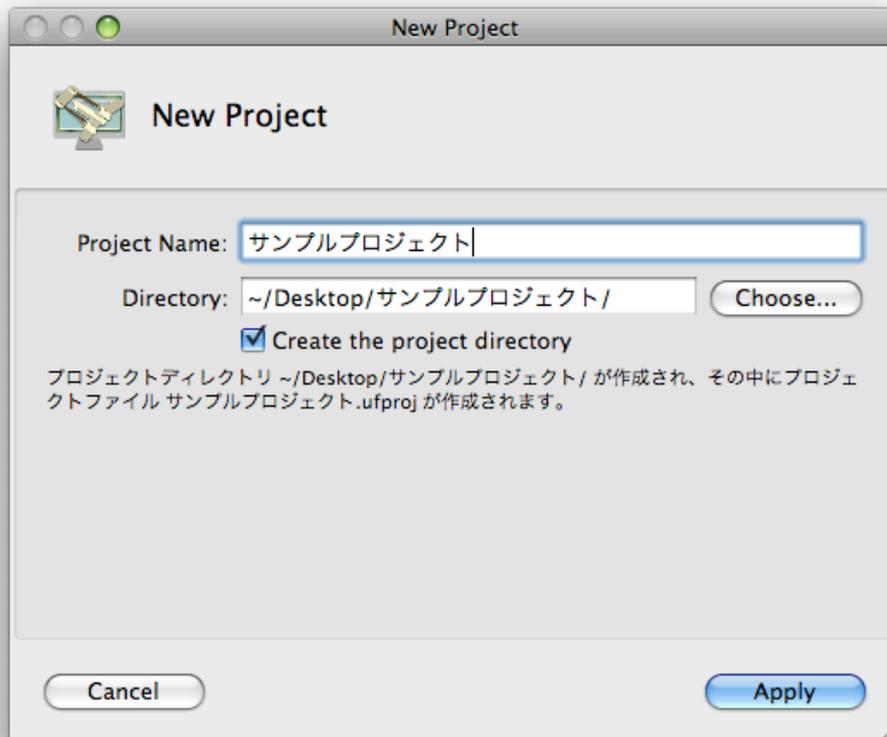
アプリケーション “Load Design” で使われる用語です。

- ・ プロジェクトファイル : Load Design プロジェクトのタイトルやコメント、また、モデルファイルや結果ファイルの名前などを管理しているファイルです。ファイルの拡張子は “.ldproj” です。
- ・ モデルデータ : Load Design で有限要素解析を行うのに必要なデータです。要素分割された要素 (材料特性なども含む) や、計算ケース、拘束条件、荷重条件などが含まれます。モデルファイルに書かれます。また結果ファイルにも計算時のモデルデータが含まれています。
- ・ モデルファイル : Load Design で有限要素解析のモデルデータが書かれているファイルです。テキストファイルなので、他のアプリケーションで作成、編集することもできます。プロジェクトファイルでの設定により、プロジェクトファイルを開くときに自動で読み込むこともできます。(一般的には自動で読み込むようにします。) ファイルの拡張子は “.ldmodel” です。プロジェクトファイルを使用せずに直接モデルファイルを開くこともできます。Load Design が書き出すとき、文字コードは Unicode - UTF-8 です。また読み込みは Unicode - UTF-8 または UTF-16 です。
- ・ 結果ファイル : Load Design で計算したときに作成される計算結果のファイルです。プロジェクトファイルから読み込んだり、モデルファイルに読み込むことができます。プロジェクトファイルでの設定により、プロジェクトファイルを開くときに自動で読み込むこともできます。ただし、プロジェクトファイルやモデルファイルに読み込む場合は、結果ファイル作成時のモデルデータから変化している場合は読み込むことができません。なお、モデルデータも含んでいるので、プロジェクトファイルやモデルファイルを使わずに開くこともできます。ファイルの拡張子は “.ldresult” です。
(注: 結果ファイルは、計算をした Load Design のバージョンと異なるバージョンの Load Design では一般的に読めません。バージョンが上がった場合は、読むことができなくなります。または、モデルデータを用い、再度計算する必要があります。)
- ・ Card : モデルファイルの一行が Card に相当します。Node Card (節点), Element Card (要素), Case Card (計算ケース), Restraint Card (拘束), Force Card (荷重) などがあります。空の行を Card (Null Card) と呼ぶことがあります。Card を組み合わせてモデルデータを作成します。
- ・ Card Filter : メインウィンドウの Card View や Graphic View に表示する Card を制限するときに用います。結果などの値を計算または表示する場合に、計算/表示を行う Card を制限するときに用いることもあります。
- ・ Card Value, Card Value Expression : Card の特性値や計算結果などの値、またはそれらから計算する値を表す式です。

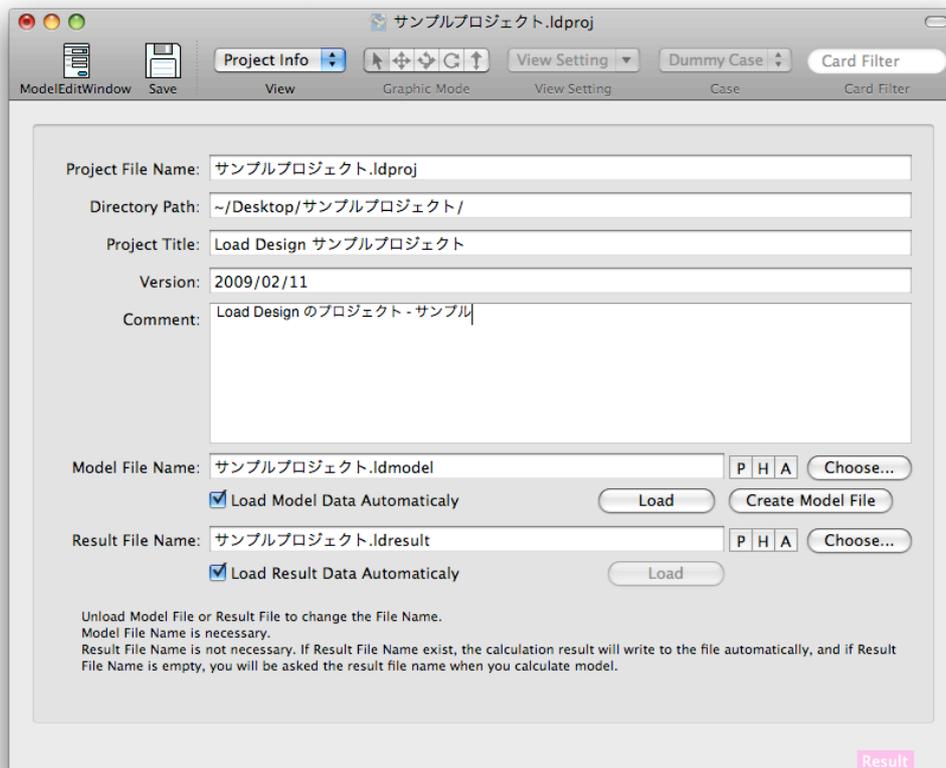
Load Design を使ってみる

1. プロジェクトの新規作成

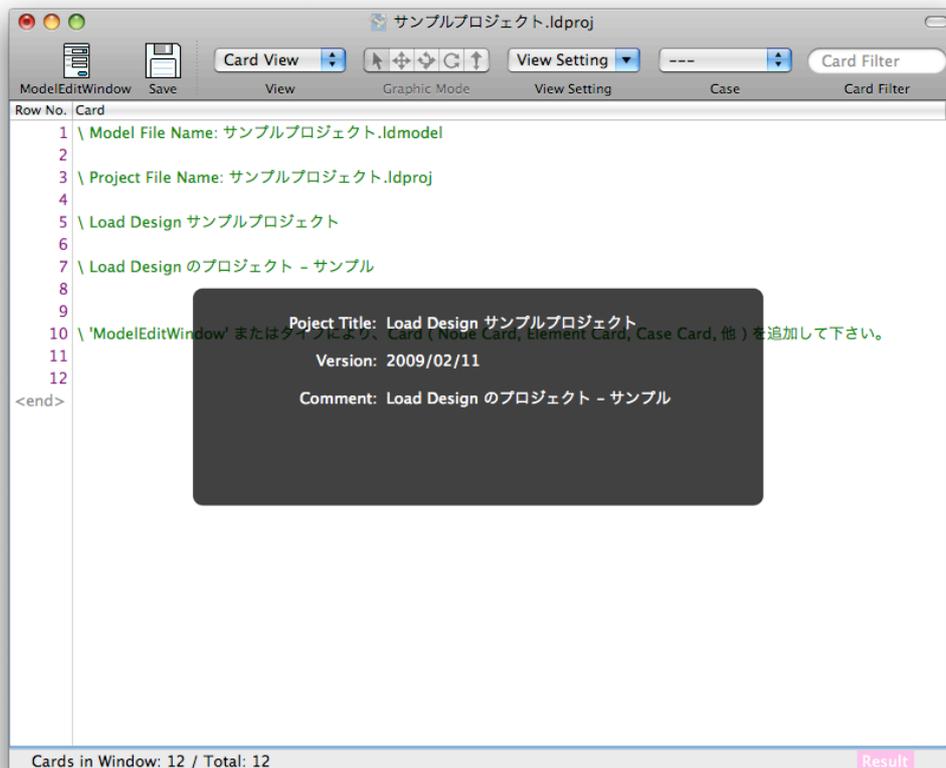
- ・ アプリケーション “Load Design” を Finder で開きます。
- ・ メインメニューの “File” - “New Project...” を選択します。ウィンドウ “New Project” が開きます。



- ・ “Project Name” テキストフィールドに、プロジェクトの名前 (プロジェクトファイルのファイル名になります。) を入力し、また “Directory” テキストフィールドの右の “Choose...” ボタンを押し、プロジェクトを作成するディレクトリを選択します。“Create the project directory” チェックボタンが ON になっていることを確認し、“Apply” ボタンを押します。“Directory” のパスのディレクトリ内に “Project Name” の名前のディレクトリが作成され、その中に “Project Name.ldproj” のプロジェクトファイルが作成されます。そして、メインウィンドウ (View type: Project Info) が開きます。

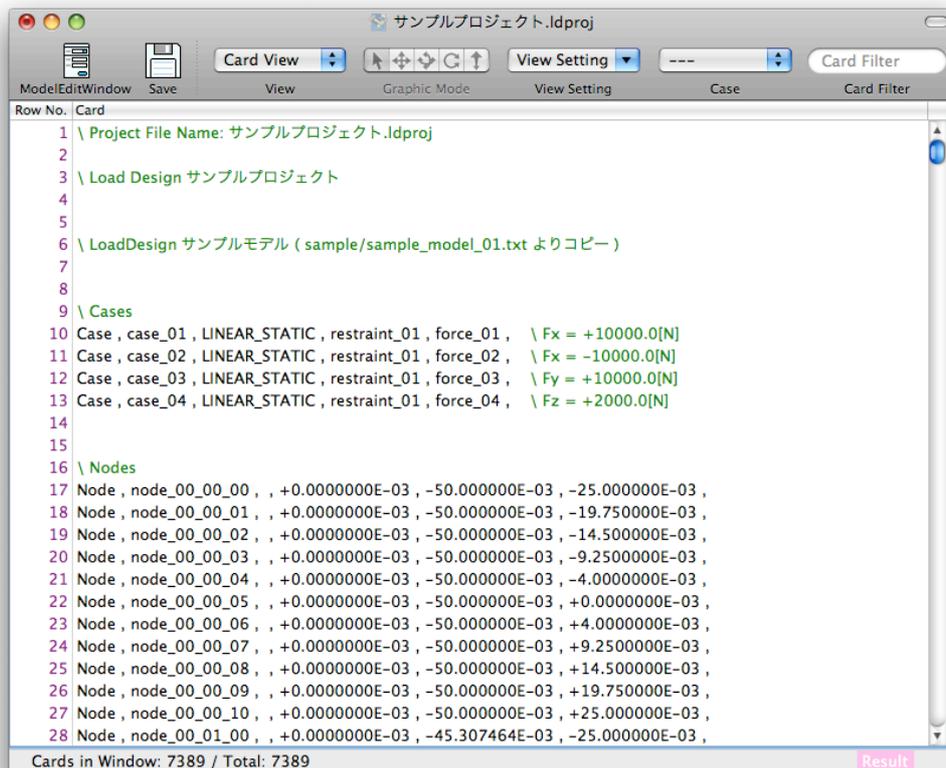


- ・ “Project Title”, “Version”, “Comment” などのテキストフィールドにプロジェクト名、モデルのバージョン情報、コメント文字列を入力します。(すべて任意) (“Comment” テキストフィールド内で改行文字を入力するには、“option” キーを押しながら “return” キーを押して下さい。) また、“Model File Name” テキストフィールドにモデルファイルの名前を入力し、“Create Model File” ボタンを押します。“Model File Name.Idmodel” のモデルファイルが作成されます。また、“Result File Name” テキストフィールドに結果ファイルの名前を入力します。(空にしておくこともできます。これが空の場合は、計算をするときに結果ファイルの名前を確認されます。)
- ・ 上記の入力やモデルファイルの作成が終われば、ツールバーにある “View” ポップアップメニューから “Card View” を選択します。または、メインメニューの “View” から選択することもできます。View が “Card View” に変わります。(“Graphic View” にすることもできます。(モデルが空の場合は “Graphic View” には何も表示されません。)) なお、ここでプロジェクトファイルに行った変更は現時点ではプロジェクトファイルに保存されていません。ツールバーの “Save” ボタン、またはメインメニューの “File” - “Save” からプロジェクトファイルとモデルファイルを保存できます。(プロジェクトファイルを開いているときは、プロジェクトファイルとモデルファイルが同時に保存されます。)

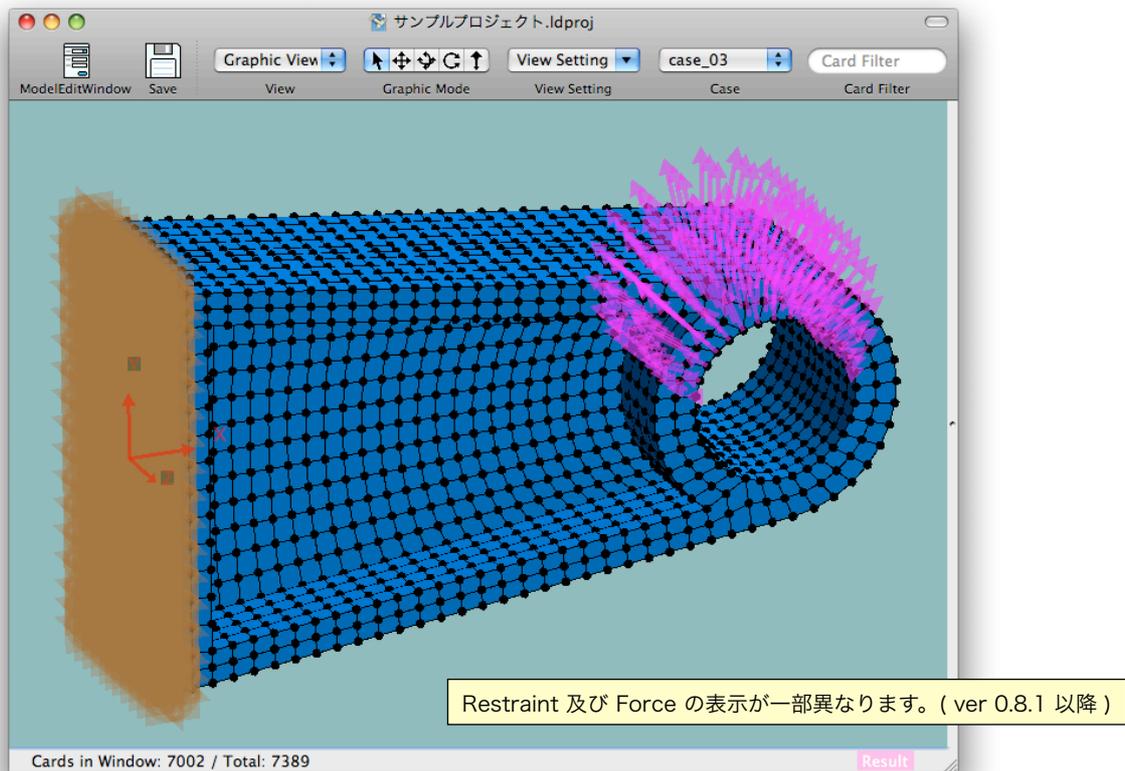


2. 計算モデルの作成

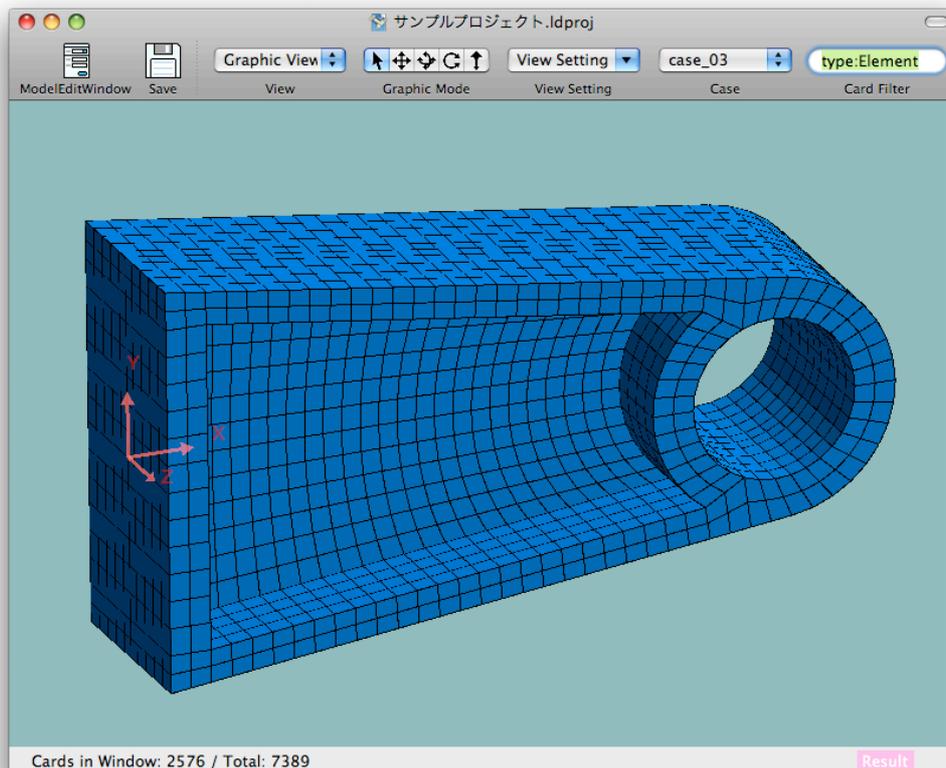
- ・ 計算モデルはモデルファイルに書き込まれます。モデルファイルはテキストファイルです。
- ・ “Model Edit Window”（ツールバーやメインメニューから開くことができます。）で作成したり、“Card View”で直接入力することもできます。または、他のテキストエディタや表計算などのアプリケーションでモデルデータを作成し、それを“コピー & ペースト”で張り付けることもできます。
- ・ モデルデータの詳細は、“モデルデータについて”を参照してください。
- ・ ここでは、“Load Design/sample/sample_model_01.txt”からコピーします。
“sample_model_01.txt”をテキストエディタソフトウェアで開き、全体を“コピー”し、“Load Design”の“Card View”でどの行も選択されていない状態で、“ペースト”してください。いずれかの行が選択されている場合は、“esc”キーを押すか、一番下の灰色で“<end>”と表示されている行（またはそれより下）をクリックすると選択を解除できます。なお、いずれかの行が選択されている状態で“ペースト”すると、選択されている行の上にペーストされます。不要な行は、それらを選択した状態でメインメニューの“Edit” - “Delete”を選択すると消去できます。また、行を編集する場合は、その行をダブルクリックするか、その行を選択した状態で“return”キーを押すと、編集できるようになります。（バックslash “\”以降（緑色の文字）はコメントです。計算に影響しません。自由に記入、編集できます。）



- ・一度ファイルを保存します。 ツールバーの“Save” ボタン、またはメインメニューの“File” - “Save” からプロジェクトファイルとモデルファイルを保存できます。
- ・ツールバーの“View” ポップアップメニュー、またはメインメニューの“View” から“Graphic View” を選択し、モデルをグラフィック表示することができます。 ツールバーの“Case” メニューやメインメニューの“View” - “Case” で Case を選択すると、“Graphic View” では、その Case での拘束や荷重を確認できます。(“Graphic View” で、(色を変更していない状態では) 拘束が茶色、荷重がピンク色で表示されます。)

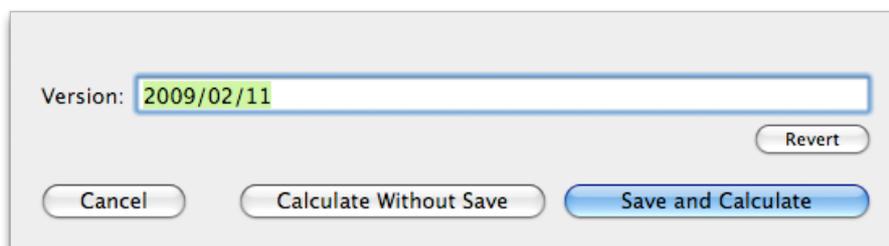


- ・メインメニューの“Window” - “New Window”で、同じドキュメント(プロジェクトやモデル)のウィンドウを追加し、同一のドキュメントの複数のウィンドウを開くことができます。各ウィンドウはそれぞれ表示の設定や表示する部分などを変更することができ、同じモデルを複数のウィンドウで確認しながら、モデルの作成、編集、確認などを行うことができます。
- ・ツールバーの右上には、“Card Filter”テキストボックスがあります。(ツールバーに表示する項目、順番は変更することもできます。)ここに“type:Node”、“type:Element”などを入力すると、それぞれ“Nodeのみ”、“Elementのみ”を表示することができます。Card Filterの詳細は“Card Filter”の項を参照してください。(例えば、Card Filterテキストボックスに“type:Node”と入力すると節点(Node)のみが表示されます。“type:Elem”または“type:Element”と入力すると要素(Element)のみが表示されます。

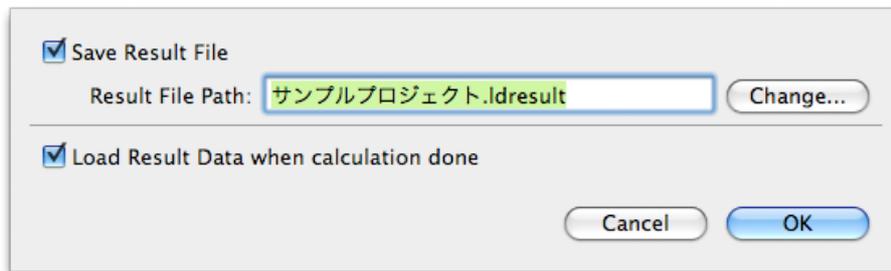


3. 計算

- ・メインメニューの“Calculation” - “Run Calculation” を選択します。プロジェクトファイルを開いている場合は “Version” 文字列の確認のダイアログが表示されます。(モデルファイル (拡張子 “.ldmodel”) を直接開いている場合は表示されません。) ここで、プロジェクトの “Version” の文字列を変更できます。(これは、“Project Info” View にある “Version” と同じです。計算するときに、毎回確認され、変更/入力することができます。) “Save and Calculate” (保存して計算) または “Calculate without Save” (保存しないで計算) ボタンを押します。



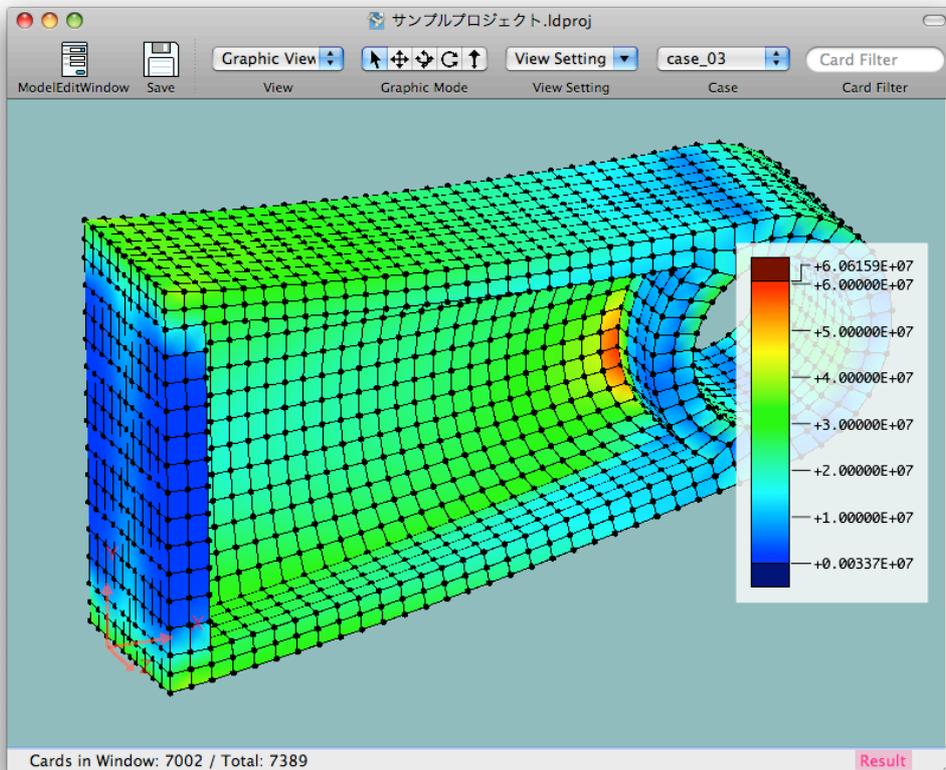
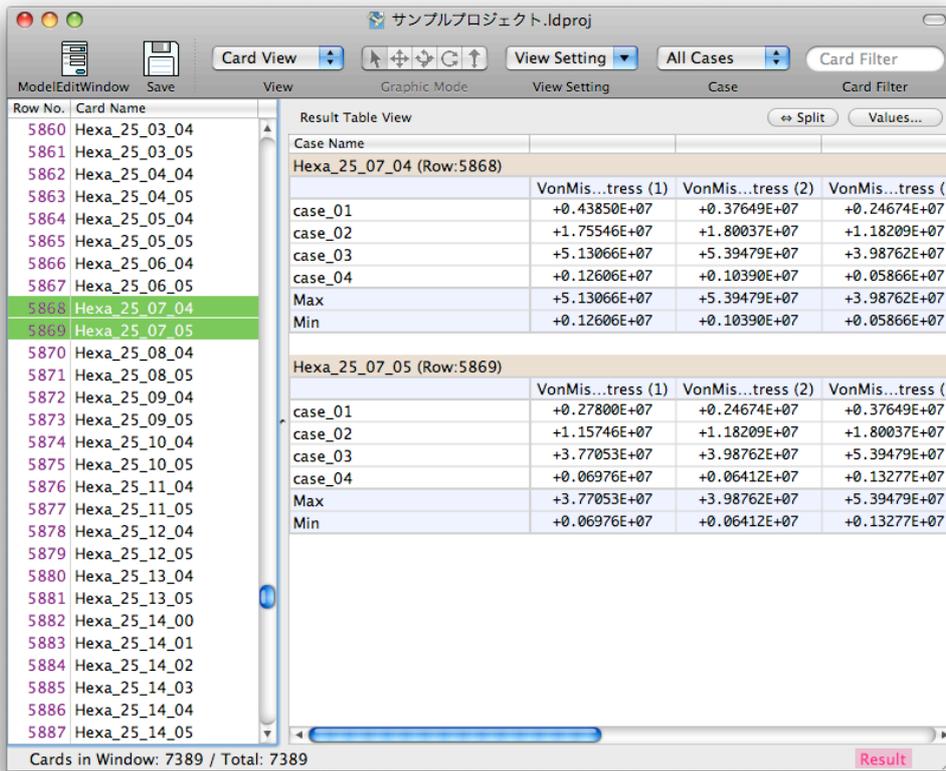
- ・プロジェクトファイルに結果ファイルの名前が入力されているときは、計算を開始します。また、ここでプロジェクトファイルに結果ファイルの名前が入力されていない場合、またはモデルファイルを直接開いている場合は、ダイアログが表示され、結果ファイルの名前を入力します。(結果ファイルを保存しないことも選択できます。(注: 現時点では、後で結果ファイルを保存することはできません。)) また、計算終了後に、結果ファイルを読み込むかどうかを指定します。(結果ファイルを読み込むように指定しなかった場合は、計算終了後に手動で結果ファイルを読み込む必要があります。) “OK” ボタンを押すと計算を開始します。



- ・計算が始まると、“Calculation” ウィンドウが表示され、計算の進捗状況が表示されます。計算の途中で、“Pause”（計算の一時停止），“Quit”（計算を途中で終了）することができます。



- ・プロジェクトファイルで、または計算時に結果を読み込むように設定した場合は、計算終了時に自動的に結果が読み込まれます。読み込むように設定していない場合は、メインメニューの“File” - “Load Result Data” を選択し読み込みます。結果を読み込んでいるときは Window 下部ステータスバー右側に “Result” とピンク色で表示されます。読み込んでいないときは、薄いピンク色で表示されます。
- ・ツールバーの “View Setting” メニュー、またはメインメニューの “View” - “View Setting” より、結果を表示する View Setting を選択すると、その設定により、結果を表示することができます。そのとき、結果を表示する Case を、ツールバーの “Case” メニュー、またはメインメニューの “View” - “Case” メニューにより選択します。View type が “Card View” のとき “View Setting” メニューで “Result” を選択し、メインウィンドウの右端に閉じられている Result Table View を表示すると（ウィンドウの右端付近のマウスポインタの形状が変化するところでマウスを左にドラッグすると Result Table View が開きます）、左のペインで Node Card を選択したときは、その Node の変位、Element Card を選択したときは、Von Mises Stress が表示されます。View type が “Graphic View” のときは “View Setting” メニューで “Displacement”、“VonMisesStress” または “Displacement & VonMisesStress” を選択すると、それらの結果が表示されます。または、“Card View” や “Graphic View” で、“View Setting” の中の “View Setting...” を選択すると、View Setting シートが現れ、そこで表示する内容を指定できます。



4. その他

- ・ 結果ファイルを読み込んでいるとき（結果ファイルと接続しているとき）は、モデルデータ（モデルファイル）を編集することはできません。編集するには、メインメニューの“File” - “Unload Result Data”を選択し、結果ファイルの読み込みを解除します。モデルデータを編集できるようになります。ただし、結果ファイルには計算時のモデルデータが含まれていますが、計算後、モデルデータを編集すると、異なるモデルデータで計算した結果ファイルは読み込めなくなります。（結果ファイルを単独で開くことはできません。）
- ・ アプリケーションを終了するには、メインメニューの“Load Design” - “Quit Load Design”を選択します。

モデルデータについて

1. Load Design での有限要素解析について

- 有限要素解析では、計算対象の構造物などを、単純な形状および変形形態の要素 (Element) に分割して、それらの要素の集合として模擬し、荷重を加えた時の構造物の変形や各部の応力などを計算します。(下図参照) なお、すべての要素は節点 (Node) (または節点の自由度 (DOF) (並進自由度 T_x , T_y , T_z , 回転自由度 R_x , R_y , R_z)) を介して他の要素と接続しています。また、拘束条件や荷重条件は節点に対して適用されます。Load Design では、変位法により計算を行います。要素の剛性マトリクスより系全体の剛性マトリクスを作成し、荷重条件、拘束条件を適用し、剛性方程式 (変位についての方程式) を解き、全節点の変位を求めます。(その他の荷重や応力などは変位の値を用いて計算されます。) なお、解析の形態は“線形静解析”に限られます。なお、有限要素解析 (または有限要素法) の詳細については他の書籍などを参考にしてください。
- Load Design では、単一のモデル内で複数の計算ケース (Case) について計算できます。計算ケースごとに拘束条件、荷重条件を指定できます。なお、要素はすべてのケースに共通です。(計算ケース毎に要素を変更、増減することはできません。) また、計算ケースごとの節点の変位や要素に加わる力、応力などが計算されます。
- Load Design は無次元で計算します。モデルデータ内の節点の座標、要素の断面積、材料のヤング率、荷重条件の力などの値は、一貫性を持った単位系での値である必要があります。また、節点の変位や要素に加わる力、応力などの計算結果はその単位系での値になります。例えば長さの単位に [m] (メートル)、力の単位に [N] (ニュートン) を使うと応力の単位は $[N/m^2] = [Pa]$ (パスカル) になります。長さの単位に [mm]、力の単位に [kgf] を使うと、応力の単位は $[kgf/mm^2]$ になります。
- 解析するにあたって、計算対象の構造物や荷重形態を模擬するようにモデルデータの作成を行います (モデル化)、モデル化は適切に行われる必要があります。要素分割 (要素の選択、要素数、要素の形状など)、拘束条件、荷重条件の設定、線形静解析で良いかどうかの検討などが含まれます。また、要素の特性や性能についても把握している必要があります。場合によっては Load Design で計算できないこともあります。

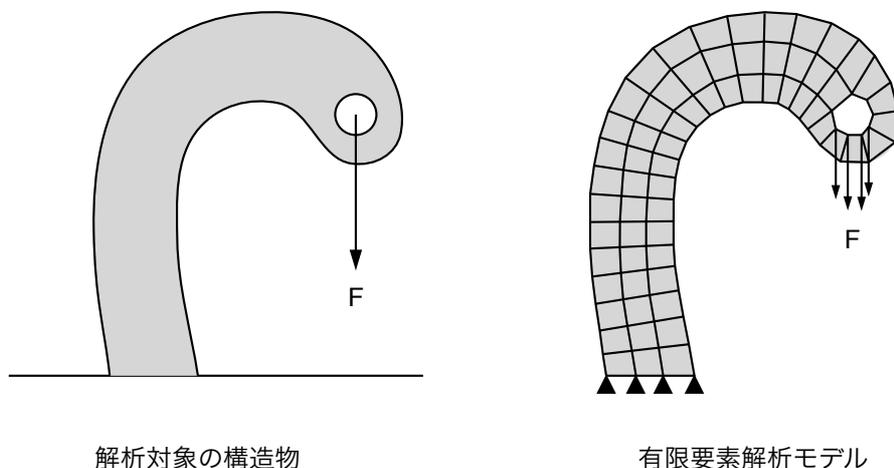


図 - 有限要素解析の概要

2. モデルデータの概要

- ・モデルデータは、有限要素解析の計算モデルのデータです。有限要素解析の計算に必要なすべての情報を含みます。これには、計算を行う物の形状（要素分割された要素や節点）、材料特性、計算ケース、拘束条件、荷重条件などがあります。拘束条件、荷重条件は計算ケース毎に指定します。
- ・モデルデータはモデルファイル（拡張子“.ldmodel”）に保存されます。
- ・モデルファイルはプレーンテキスト形式のテキストファイルです。（文字のエンコードは、読み込み: Unicode - UTF-8 または UTF-16、書き込み: Unicode - UTF-8 です。）他のテキストファイルを作成、編集するソフトウェアを使用して作成することも可能です。
- ・モデルデータは“Card”を組み合わせて作成します。モデルデータの1行がCardに相当します。空の行（コメントを含むこともあります。）をCard（Null Card）と呼ぶこともあります。Null Card以外のCardは複数のカラム（Column）を持っており、コンマ‘,’で区切られています。
- ・各行（各Card）には‘\’（バックスラッシュ）の後にコメントを書くことができます。（日本語キーボードでは‘option’キー + ‘¥’キーで‘\’を入力することができます。）コメントは計算結果に影響しません。
- ・Card（Card Type）には、Node Card（Node Card Type）、Element Card（Element Card Type 以下省略）、Element Property Card、Case Card、Restraint Card、Force Card（及びNull Card）などがあります。Cardの初めのカラムはCard Classの名前（Cardの種類：“Node”、“Rod”、“Quad”、“Case”、“Force”など）で、ここでCardの種類を識別します。（なお、“Rod”、“Quad”は共にElement Cardです。）
- ・Cardの2つめのカラムは“Card名”（Card Name）です。Null Card以外のCardはCard名を持っており、他のCardからCard名で参照されます。（例えば、Element Cardは数個のNodeを参照しています。）Card名には、‘a-z’、‘A-Z’、‘0-9’、‘_’（アンダーバー）、および全角文字などが使えます。その他の半角文字（!“#\$%&’()*+,-./:;<=>?@[\\]^`{|}~ や ‘（スペース））などは使えません。
- ・Node Card、Element Card、Element Property Card、Case Cardについては、複数のCardを同じCard名にすることはできません。Restraint CardやForce Cardは、同じCard名で複数のCardを作成することができます。（Case CardにおいてRestraint CardやForce CardのCard名が指定されたとき、指定されたCard名のすべてのRestraint CardやForce Cardが適用されます。）
- ・Node Cardは、節点の情報です。Elementは節点を介して、隣のElementと繋がります。また、拘束条件や荷重条件はNodeに適用されます。
- ・Element Cardは、Rod（棒状の要素）、Quad（4節点の板状の要素）、Hexa（8節点の6面体形状の要素）などがあります。Element Cardは、数個のNode Cardを参照し、それらを繋いで要素の形状を指定します。また、Nodeを介して隣のElementと繋がります。また、Element Property Cardを参照することがあります。
- ・Element Property Cardは、Element Cardより参照され、Elementの特性などを指定します。これには、RodProp（Rod要素の特性）、ShellProp（Quadなどシェル要素の特性）、SolidProp（Hexa

などソリッド要素の特性)があります。Material Card も Element Property Card と呼ばれることがあります。

- ・ Case Card は、計算する Case を定義します。Calculation Type (線形静解析のみ) や、Restraint Card (拘束条件) や Force Card (荷重条件) の名前を指定します。なお、Restraint Card や Force Card は同じ名前の Card が複数存在することがあります。指定している Card 名のすべての Restraint Card や Force Card がその Case に適用されます。
- ・ Restraint Card は、Case Card より参照され、Node の自由度の拘束 (自由度の固定、または自由度間の方程式) を指定します。Case から参照した Restraint Card が、その Case に適用されます。(同じ Card 名の Restraint Card が複数存在することがあります。) なお線形静解析では、構造全体 (または一部) が剛性がない状態で動かないようにいくつかの自由度を拘束する必要があります。
- ・ Force Card は、Case Card より参照され、荷重を指定します。Case から参照した Force Card が、その Case に適用されます。(同じ Card 名の Force Card が複数存在することがあります。)
- ・ 基本的には Card は任意の順序で書くことができます。例えば Element Card は Node Card を参照していますが、どちらの Card が前にあってもかまいません。ただし、モデルデータ内の Node Card の順番によって計算時間などが変わることがあります。(詳細は後の“その他 (“剛性マトリクスのバンド幅、計算誤差、他について”) の項を参照してください。)

3. 簡単な計算モデルとモデルデータ

図1の計算モデルのモデルデータをモデル1に示します。

モデル1 - モデルデータ

```
\ サンプルモデル - ( 単位系: [mm], [kgf] )

\ Cases - 計算ケース
Case , case01 , LINEAR_STATIC , restraint01 , force01 ,

\ Nodes - 節点
Node , node01 , , 0.0 , 0.0 , 0.0 ,
Node , node02 , , 0.0 , 120.0 , 0.0 ,
Node , node03 , , 0.0 , 0.0 , -60.0 ,
Node , node04 , , 0.0 , 120.0 , -60.0 ,
Node , node05 , , 100.0 , 0.0 , 0.0 ,
Node , node06 , , 100.0 , 120.0 , 0.0 ,

\ Elements - 要素
Rod , rod01 , rodProp01 , node01 , node05 ,
Rod , rod02 , rodProp01 , node02 , node06 ,
Rod , rod03 , rodProp01 , node03 , node05 ,
Rod , rod04 , rodProp01 , node04 , node06 ,
Rod , rod05 , rodProp01 , node05 , node06 ,
Rod , rod06 , rodProp01 , node01 , node06 ,

\ Element Property, Material - 要素特性と材料特性
RodProp , rodProp01 , material01 , 50.0 ,
Material , material01 , 7000.0 , 0.3 ,

\ Restraints - 拘束
FixNode , restraint01 , node01 , 1 , 1 , 1 , 0 , 0 , 0 ,
FixNode , restraint01 , node02 , 1 , 1 , 1 , 0 , 0 , 0 ,
FixNode , restraint01 , node03 , 1 , 1 , 1 , 0 , 0 , 0 ,
FixNode , restraint01 , node04 , 1 , 1 , 1 , 0 , 0 , 0 ,

\ Forces - 荷重
Force , force01 , node05 , , 30.0 , 40.0 , -50.0 , , , ,
Force , force01 , node06 , , 0.0 , 0.0 , -60.0 , , , ,

\ end
```

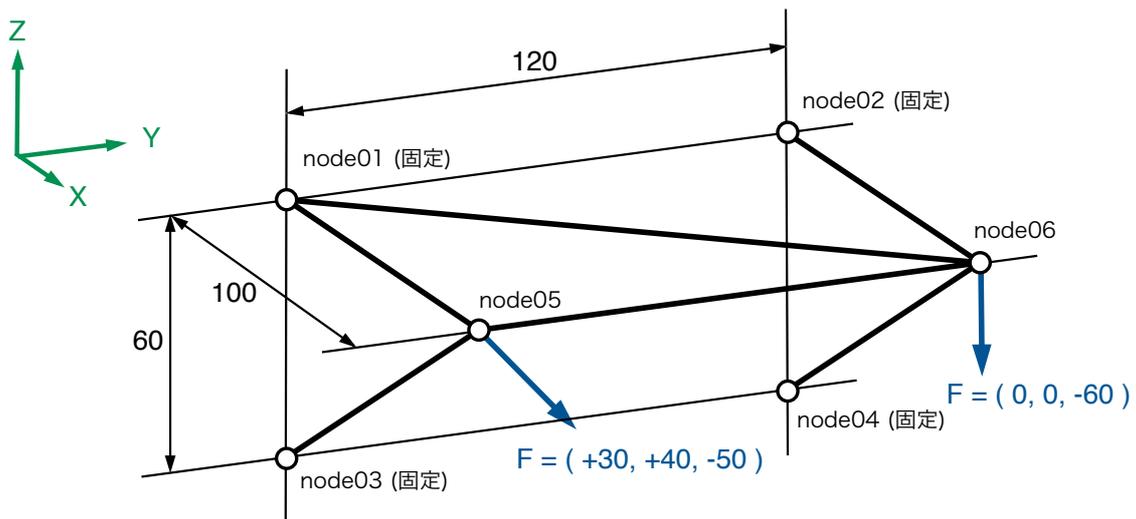


図1 - 計算モデル

- ・ ‘\’ (バックスラッシュ) 以降 (改行まで) はコメントです。次のように、有効な Card の後にコメントを入力することもできます。

```
\ サンプルモデル - ( 単位系: [mm], [kgf] )
```

```
Node , node01 , , 0.0 , 0.0 , 0.0 , \ Card の後にもコメントを記入できます。
```

- ・ 次の “Node” Card は、“node01” という名前の節点 (座標 (x, y, z) = (0.0, 0.0, 0.0)) を定義します。

```
Node , node01 , , 0.0 , 0.0 , 0.0 ,
```

- ・ 次の “Rod” Card は “rod01” という名前の Rod 要素を定義します。なお、“rod01” の両端の節点は “node01” および “node05” です。Rod の特性 (断面積や材料のヤング率) は “rodProp01” を参照します。

```
Rod , rod01 , rodProp01 , node01 , node05 ,
```

- ・ 次の “RodProp” Card は “rodProp01” という名前の Rod の特性 (断面積: 50.0 [mm²]) を定義します。なお、材料特性は “material01” を参照します。

- ・ “Material” Card は “material01” という名前の材料特性 (ヤング率: 7000.0[kg/mm²]、ポアソン比: 0.3) を定義します。

```
RodProp , rodProp01 , material01 , 50.0 ,  
Material , material01 , 7000.0 , 0.3 ,
```

- ・ 次の “Case” Card は “case01” という名前の Case (計算の種類 : “LINEAR_STATIC” (線形静解析)) を定義します。拘束条件 は “restraint01”、荷重条件は “force01” を参照します。

```
Case , case01 , LINEAR_STATIC , restraint01 , force01 ,
```

- ・ 次の “FixNode” Card は “restraint01” という名前で、節点 “node01” の自由度 Tx, Ty, Tz, Rx, Ry, Rz (Tx, Ty, Tz : 並進自由度、Rx, Ry, Rz : 回転自由度) のうち、Tx, Ty, Tz を固定します。(Rx, Ry, Rz は固定しません。)
- ・ モデル1 には “restraint01” という名前の Restraint Card が複数あります。“case01” で “restraint01” が指定されていますが、この場合、“restraint01” の名前のすべての拘束がその Case に適用されます。
- ・ なお線形静解析では、構造全体 (または一部) が剛性がない状態で動かないように固定する必要があります。

```
FixNode , restraint01 , node01 , 1 , 1 , 1 , 0 , 0 , 0 ,
```

- ・ 次の “Force” Card は “force01” という名前で、節点 “node05” に荷重 (Fx, Fy, Fz, Mx, My, Mz) = (+30, +40, -50, 0, 0, 0) [kgf] を加えます。
- ・ モデル1 には “force01” という名前の Force Card が複数あります。“case01” で “force01” が指定されていますが、この場合、“force01” の名前のすべての荷重がその Case に適用されます。

```
Force , force01 , node05 , , 30.0 , 40.0 , -50.0 , , , ,
```

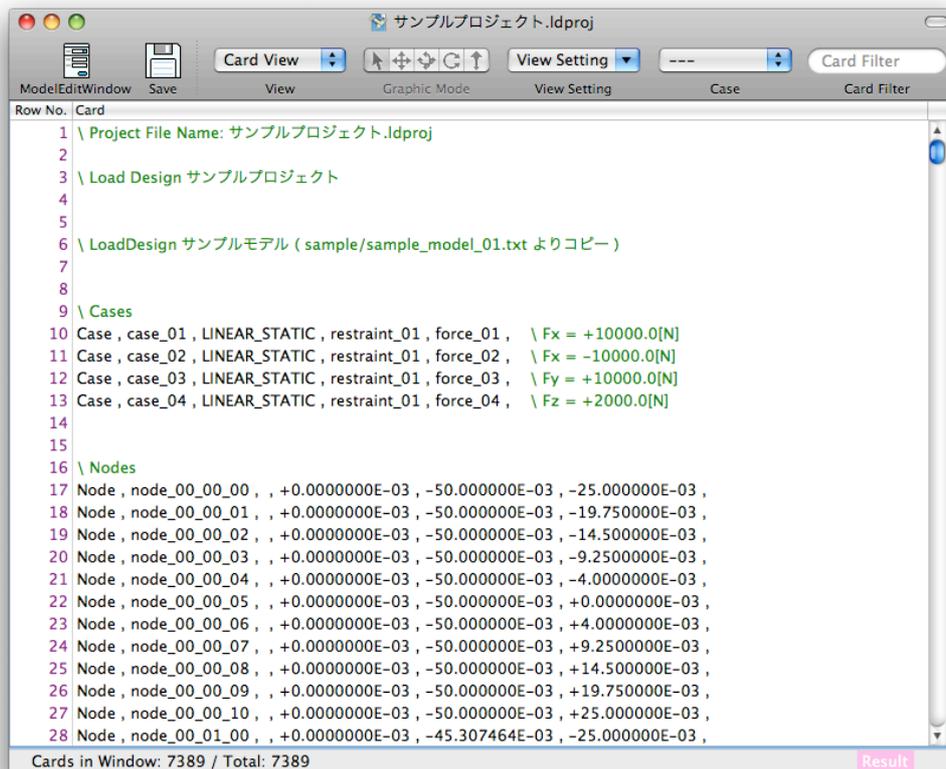
4. 各 Card Class について

- ・ Card Class ごとの詳細は、“Card Class” の項を参照してください。

Load Design の使用方法

1. Load Design のユーザインターフェース

メインウィンドウ - 共通

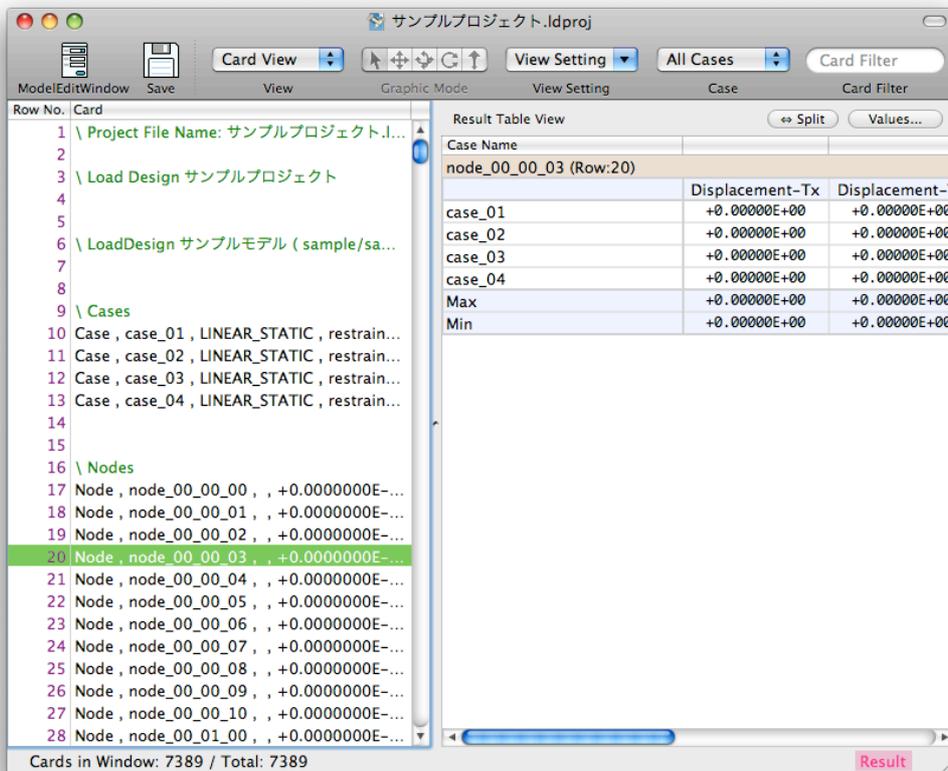


・ ツールバー

- ・ “Model Edit Window” ボタン : “Model Edit Window” を開きます。
- ・ “Save” ボタン : プロジェクトファイル、モデルファイルを保存します。
- ・ “View Type” ポップアップメニュー : View Type を変更します。
- ・ “Graphic Mode” スイッチボタン : Graphic View において、マウスのモードを指定します。
(“Arrow” : Card (Node や Element など) の選択 / “Translate” : 移動 / “Rotate 1” : 回転 1 / “Rotate 2” : 回転 2 / “Scale” : 拡大縮小)
- ・ “View Setting” メニュー : Card View / Graphic View において、それぞれの View の既定の設定のセットを適用します。または “View Setting シート” を表示します。
- ・ “Case” ポップアップメニュー : Case (計算ケース) を指定します。指定した Case に関する Card のみが Card View や Graphic View に表示されます。Case を特に指定しないときは “---” を選択します。すべての Case を選択する場合は “All Cases” を選択します。単一の Case を選択す

る場合は (“Some Cases” サブメニューの外にある) それぞれの Case 名を選択します。複数の Case を選択する場合は “Some Cases” サブメニュー内のそれぞれの Case 名を選択します。Case が指定されている場合は、その Case で有効な Card のみが Card View や Graphic View に表示されます。(指定された Case で使用されない Force Card や Restraint Card は表示されなくなります。)

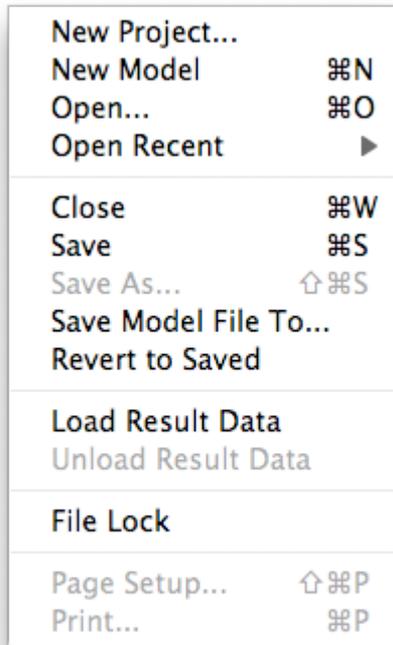
- ・ “Card Filter” テキストフィールド : Card Filter をセットします。Card Filter がセットされている場合は、Card Filter に適合した Card のみが Card View や Graphic View に表示されます。
- ・ ステータスバー : ウィンドウ下部の領域です。左側にモデルの Card の数 (Total) と、ウィンドウに表示されている Card (Card Filter および Case メニューに制限された Card) の数 (Cards in Window) が表示されます。また右側に結果ファイルを読み込んでいるかどうかが表示されます。(結果ファイルを読み込んでいるときは “Result” インジケータが濃いピンク色で表示されます。結果ファイルを読み込んでいないときは “Result” インジケータが薄いピンク色で表示されます。)



- ・ Result Table View : View Type が “Card View” または “Graphic View” のとき、(初期状態では) ウィンドウ右端に Result Table View が閉じられています。ウィンドウの右端のマウスポインタの形状が変化するところでマウスを左にドラッグすると Result Table View が開きます。Result Table View は、そのウィンドウの (左側にある) Card View または Graphic View で選択されている Card の結果などの値を表形式で表示します。“Values” ボタンを押すと Result Table View に表示する値 (Card Value) を設定できます。“⇄ Split” ボタンを押すと、分割方向 (左右分割 / 上下分割) を変更できません。

メインメニュー

・“File” メニュー



・“New Project...”：新規プロジェクトを作成します

・“New Model”：新規モデルウィンドウを開きます。

・“Open...”：ファイルを指定して開きます。

・“Open Recent”：最近使ったファイルを開きます。

・“Close”：ウィンドウを閉じます。

・“Save”：ファイルを保存します。

・“Save As...”：ファイルを別名で保存します。モデルファイルを直接開いているときのみ使用できます。

・“Save Model File To...”：(プロジェクトなどの)モデルデータを別名で保存します。

・“Revert to Saved”：最後に保存した状態に戻します。

・“Load Result Data”：プロジェクトファイルやモデルファイ

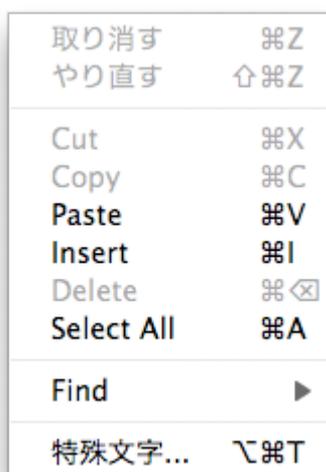
ルに結果ファイルを読み込みます。

・“Unload Result Data”：結果ファイルの読み込みを解除します。

・“File Lock”：選択しているウィンドウのモデルデータを一時的に編集できないようにします。

・“Page Setup...” / “Print...”：使用できません。

・“Edit” メニュー



・“Undo”：直前のモデルデータに対する操作を取り消します。

・“Redo”：“Undo”の取り消しを行います。

・“Cut”：選択範囲をクリップボードにコピーし、選択範囲は消去します。なお、'option'キーを押しながら実行すると、タブ区切りの文字列としてクリップボードにコピーします。

・“Copy”：選択範囲をクリップボードにコピーします。なお、'option'キーを押しながら実行すると、タブ区切りの文字列としてクリップボードにコピーします。

・“Paste”：クリップボード内のデータをペーストします。なお、'option'キーを押しながら実行すると、クリップボード内の文字列をタブ区切りの文字列としてペーストします。

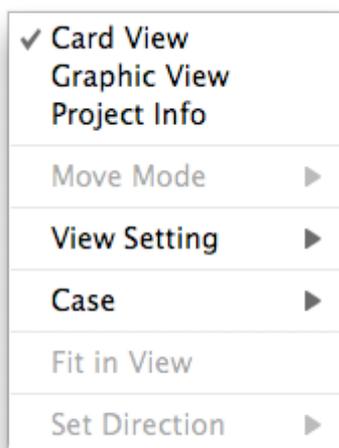
・“Delete”：選択範囲を消去します。

・“Select All”：すべてを選択します。

・ “Find” メニュー

- ・ “Find...” : 検索パネルを表示します。
- ・ “Find Next” / “Find Previous” : 検索パネルに入力された検索文字列を順方向 / 逆方向に検索します。
- ・ “Use Selection for Find” : 選択範囲の文字列を検索文字列にセットします。
- ・ “Scroll to Selection” : 選択範囲が見えるようにスクロールします。(Card View のみ)

・ “View” メニュー



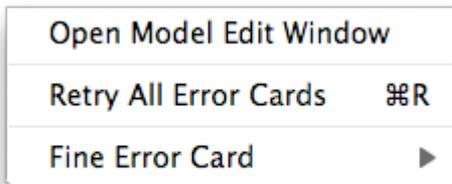
・ “Card View” / “Graphic View” / “Project Info” (/ “Result Info”) : View Type を変更します。ツールバー上の “View Type” メニューと同じです。

・ “Move Mode” : Graphic View において、マウスのモードを指定します。ツールバー上の “Graphic Mode” スイッチボタンと同じです。(“Arrow” : Card (Node や Element など) の選択 / “Translate” : 移動 / “Rotate 1” : 回転 1 / “Rotate 2” : 回転 2 / “Scale” : 拡大縮小)

・ “View Setting” : Card View、Graphic View において、それぞれの View の既定の設定のセットを適用します。または “View Setting シート” を表示します。ツールバー上の “View Setting” メニューと同じです。

- ・ “Case” メニュー : Case (計算ケース) を指定します。指定した Case に関する Card のみが Card View、Graphic View また Result Table View に表示されます。ツールバー上の “Case” ポップアップメニューと同じです。Case を特に指定しないときは “---” を選択します。すべての Case を選択する場合は “All Cases” を選択します。単一の Case を選択する場合は (“Some Cases” サブメニューの外にある) それぞれの Case 名を選択します。複数の Case を選択する場合は “Some Cases” サブメニュー内のそれぞれの Case 名を選択します。Case が指定されている場合は、その Case で有効な Card のみが Card View や Graphic View に表示されます。(指定された Case で使用されない Force Card や Restraint Card は表示されなくなります。)
- ・ “Fit in View” : Graphic View において、モデルが Graphic View にフィットするように、表示位置、大きさを調整します。
- ・ “Set Direction” メニュー : Graphic View において、モデルの表示する角度を指定の方向に変更します。また Graphic View にフィットするように表示位置、大きさを調整します。

・ “Model” メニュー



・ “Open Model Edit Window” : “Model Edit Window” を表示します。Card の追加などが行えます。

・ “Retry All Error Cards” : すべての Error Card を、Card の内容を変更せずに、(再度)登録することを試みます。Error Card のエラーが、Card 名の競合によるものであって、競合の原因となった他の Card などが変更されて

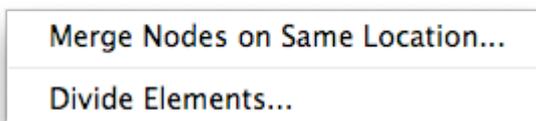
いるときなどは、この操作によりエラーがなくなることがあります。

・ “Find Error Card” メニュー

・ “Next Error Card” / “Previous Error Card” : 順方向 / 逆方向に Error Card を検索します。Error Card がなければ、“hasError” (“Card Filter” の項を参照してください) が YES の Card を検索します。

・ “Next Incomplete Card” / “Previous Incomplete Card” : 順方向 / 逆方向に “hasError” (“Card Filter” の項を参照してください) が YES の Card を検索します。

・ “Tool” メニュー

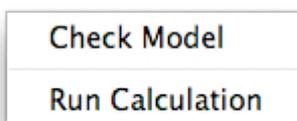


・ “Merge Node on Same Location...” : 同じ位置 (または距離が指定された値以下) の複数の節点 (Node) を統合します。(同じ位置の Node が複数ある場合、一つを残し、他は消去します。) 詳細は “Tool - Merge Nodes on Same Location” の項

を参照してください。

・ “Divide Elements...” : Element を分割 (例えば Quad 要素は 2x2, 3x3 など、Hexa 要素は 2x2x2, 3x3x3 など) します。分割した結果のモデルデータはクリップボードに書き出されます。詳細は “Tool - Divide Elements” の項を参照してください。

・ “Calculation” メニュー



・ “Check Model” : 解析モデルのチェックを行います。

・ “Run Calculation” : 解析モデルの計算を行います。

・ “Window” メニュー



・ “New Window” : 選択されているウィンドウと同じドキュメントの新しいウィンドウを表示します。各ウィンドウはそれぞれ表示の設定や表示する部分などを変更することができます。同じモデルを複数のウィンドウで確認しながら、モデルの作成、編集、確認、計算結果の確認などを行うことができます。なお、モデルを編集したとき、すべてのウィンドウに対して更新が行われます。(例えば、Card View で Node Card を追加したとき、直ちに Graphic View に追加した Node が表

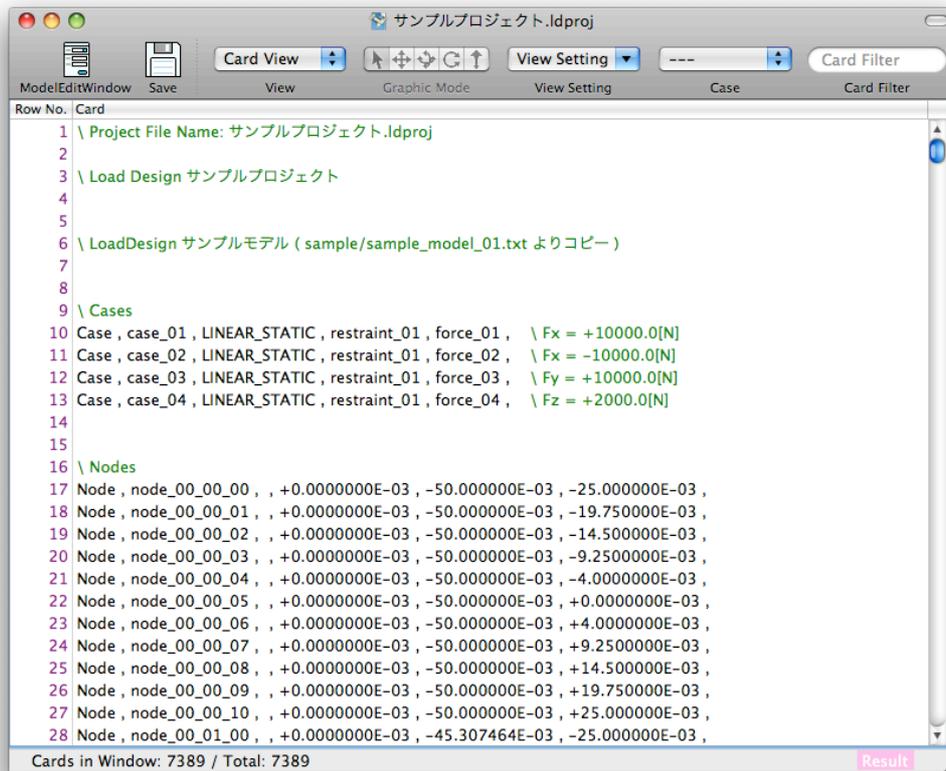
示されます。)

- ・ “Minimize” : ウィンドウを Dock に入れます。
- ・ “Bring All to Front” : Load Design のすべてのウィンドウを前面に表示します。
- ・ “Calculation Window” : “Calculation” ウィンドウを表示します。

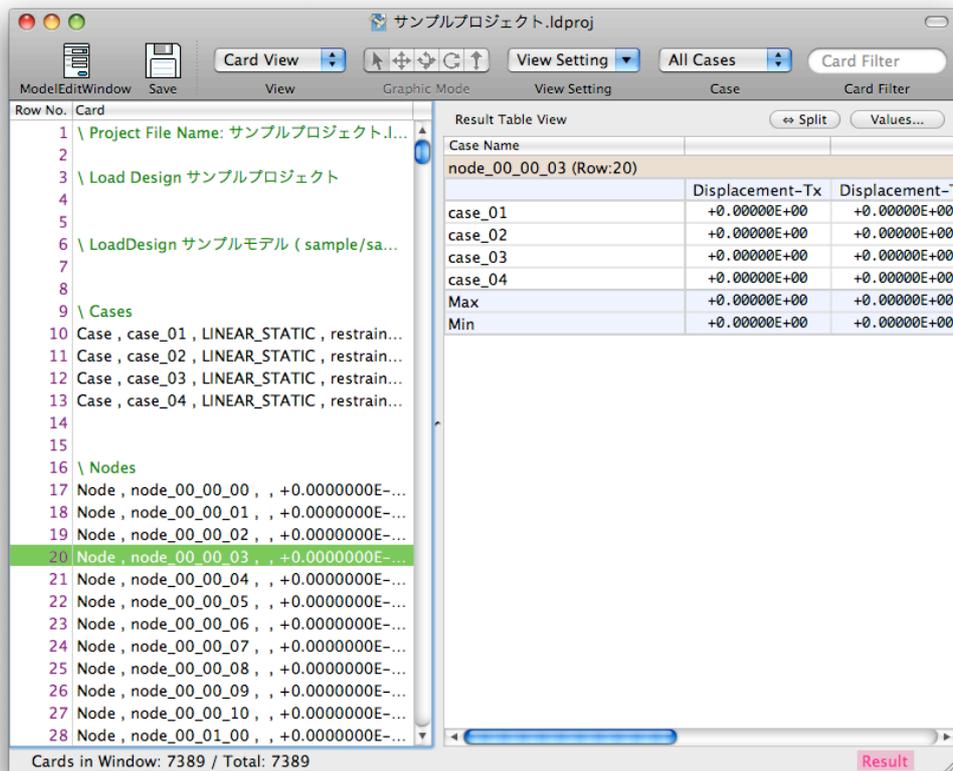
・ “Help” メニュー : ヘルプメニュー



メインウィンドウ - Card View

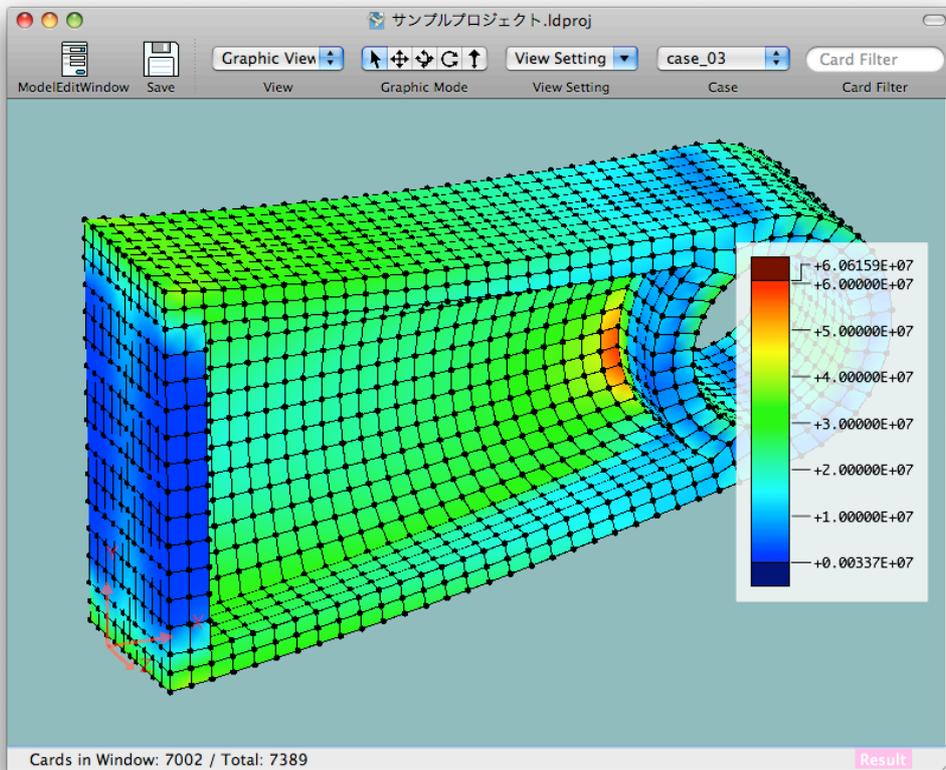
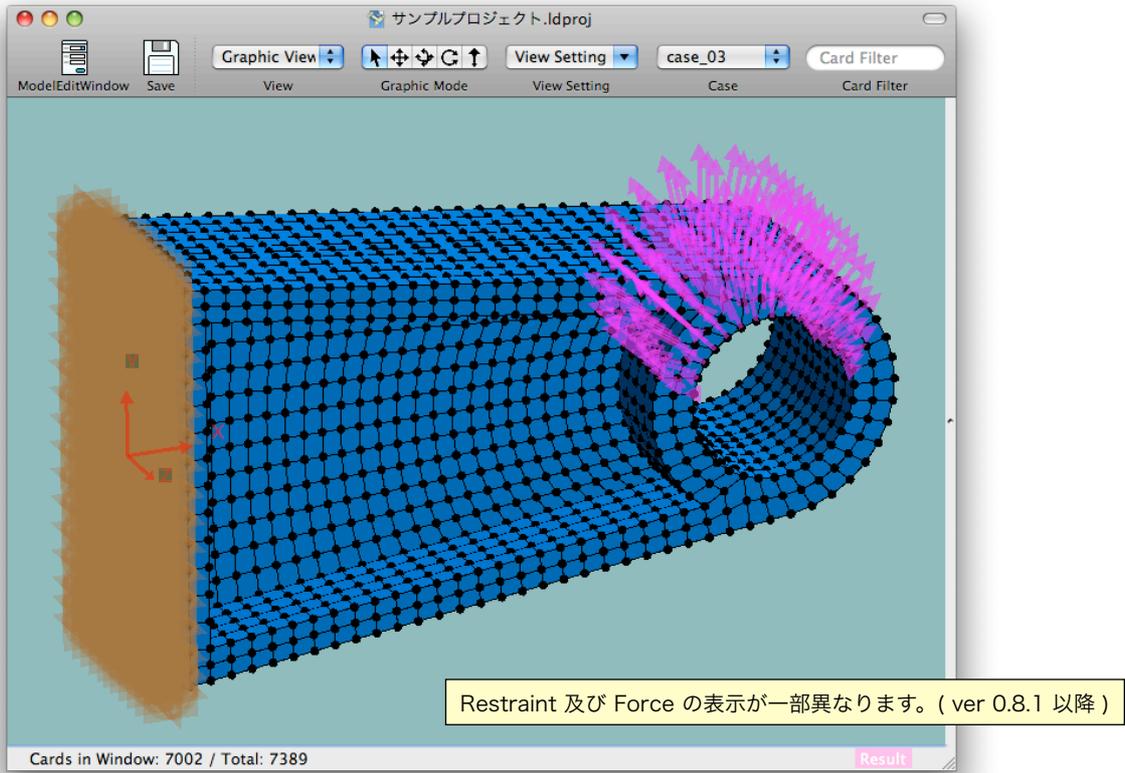


- ・ 初期状態では、モデルデータをテキスト形式で表示します。
- ・ “View Setting” メニューで、表示する項目（行番号、Card の内容、Card 名、エラーの内容）を設定できます。また Result Table View で表示する値（Card Value）をセットできます。
- ・ “Card Filter” や “Case” メニューで Card Filter、Case が指定されている場合は、それらに適合した Card のみが Card View に表示されます。

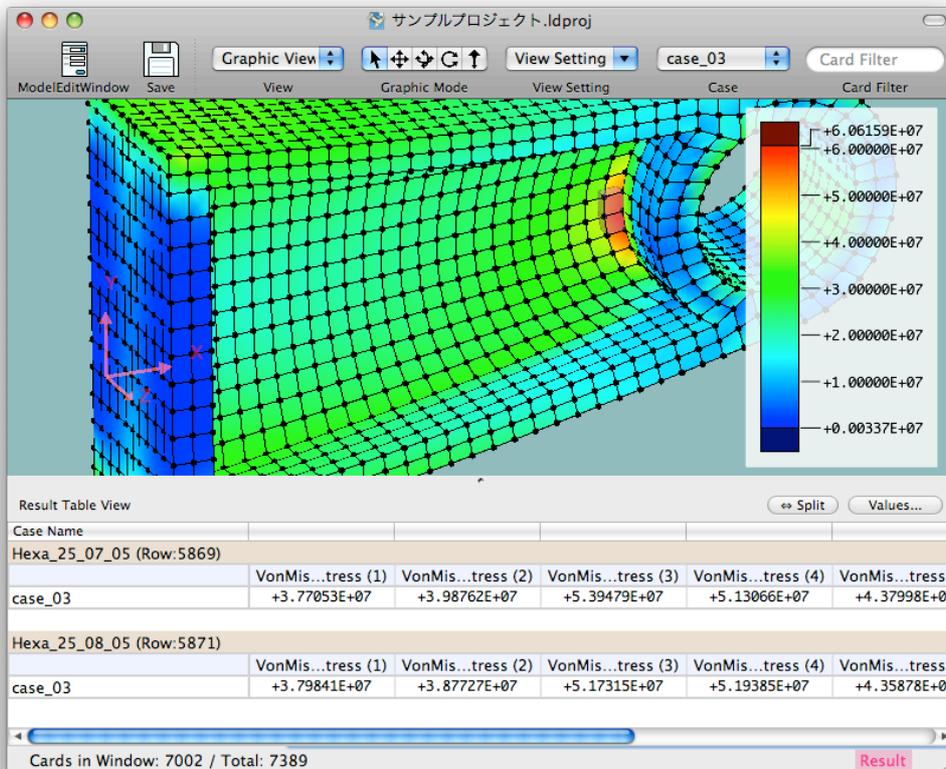


- ・ (初期状態では) 右側に Result Table View が閉じられています。Result Table View を開くと、Card View で選択した Card の計算結果などの値を Result Table View に表示できます。

メインウィンドウ - Graphic View

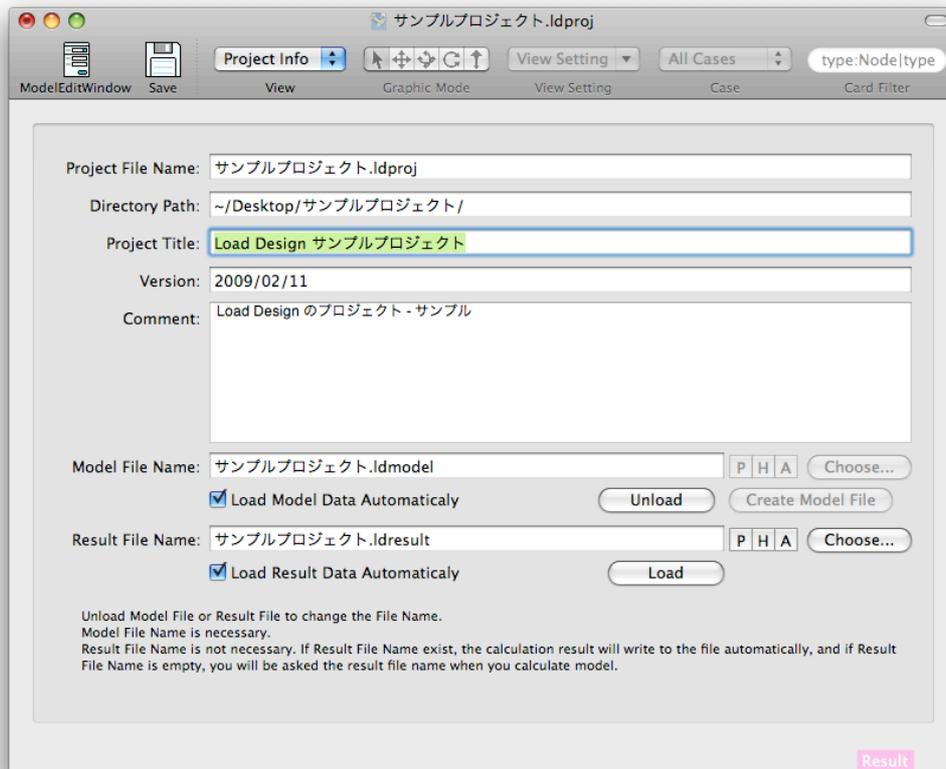


- ・モデル (Node、Element、Force、Restraint など) をグラフィックで表示します。計算結果などのスカラー値 (Von Mises 応力など) をカラーコンター図で表示したり、変位 (変形) を表示することもできます。
- ・ツールバー上の “Graphic Mode” スイッチボタン、またはメインメニューの “View” - “Move Mode” メニューにより、マウスのモード (“Arrow” : Card (Node や Element など) の選択 / “Translate” : 移動 / “Rotate 1” : 回転 1 / “Rotate 2” : 回転 2 / “Scale” : 拡大縮小) を指定します。モードが “Arrow” のときは、グラフィックビューの Node や Element などの上でクリックするとその Card が選択されます。ドラッグするとその範囲の Card (Node や Element 他) が選択されます。モードが “Translate”、”Rotate 1”、”Rotate 2”、”Scale” のときは、グラフィックビュー上でドラッグすると表示しているモデルを移動、回転、拡大縮小できます。(“Translate”、”Rorate 1” は、マウスをドラッグした方向 (上下左右) に移動、回転します。”Rotate 2” はマウスを左右にドラッグすると、その方向に回転します。”Scale” はマウスを上下にドラッグすると拡大/縮小します。)
- ・“View Setting” メニューで、表示する方法 (Element の色、Force や Restraint の表示の ON/OFF、変位(変形)表示の ON/OFF、カラーコンター表示の ON/OFF と表示する値など) を設定できます。なお、(Card View とは異なり) Graphic View の View Setting シートでは Result Table View で表示する値をセットできません。Result Table View 右上の “Values” ボタンからセットして下さい。
- ・“Card Filter” や “Case” メニューで Card Filter、Case が指定されている場合は、それらに適合した Card のみが Graphic View に表示されます。”Case” メニューで Case を指定したとき、荷重は、指定した Case の (“ForceSet” Card を用いている場合は、係数なども考慮した) 荷重条件を表示します。



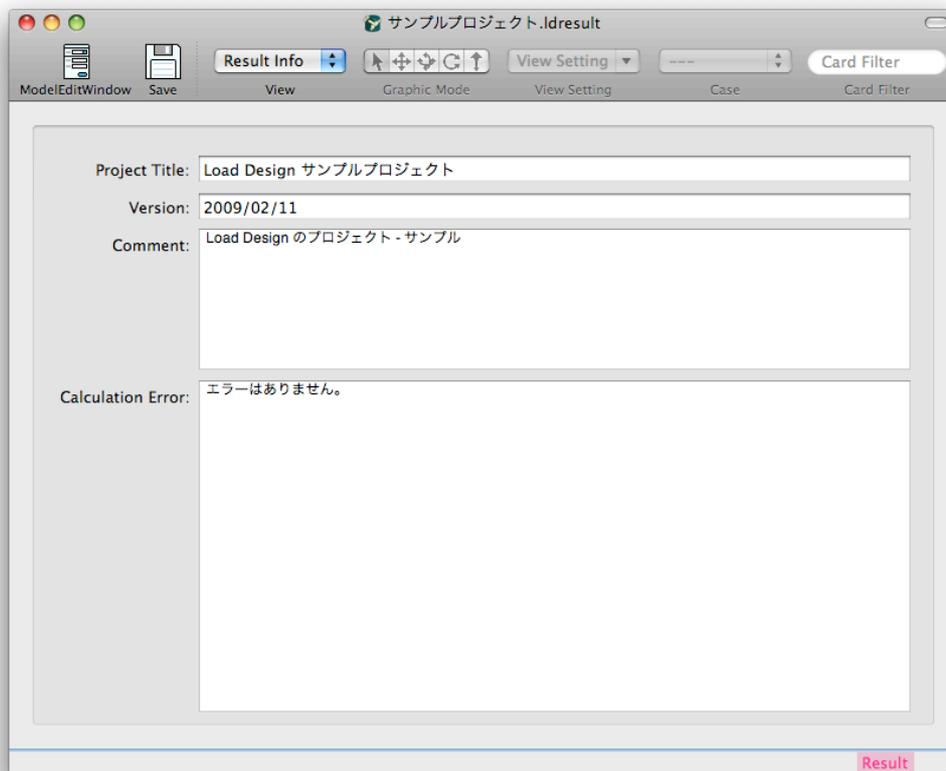
- ・(初期状態では) 右側に Result Table View が閉じられています。Result Table View を開くと、Graphic View で選択した Card の計算結果などの値を Result Table View に表示できます。

メインウィンドウ - Project Info View (プロジェクトファイルを開いている時のみ)



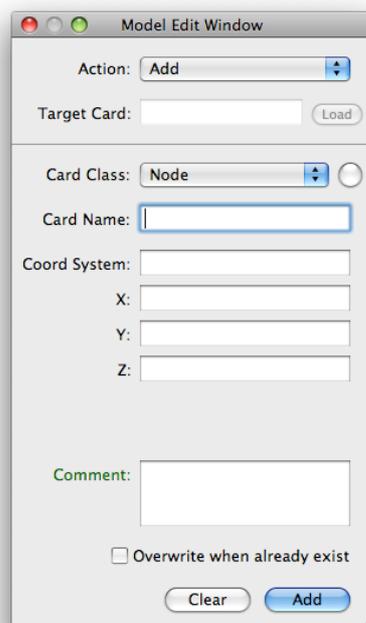
- ・プロジェクトファイルの情報を表示します。
- ・プロジェクトタイトル、Version 文字列、コメント文字列の他、モデルファイルや結果ファイルのファイル名（ファイルパス）や、（結果ファイルがあれば）結果ファイルを自動で読み込むかどうかなどをセットします。なお、コメント文字列のテキストフィールド内で改行文字を入力するには、“option” キーを押しながら“return” キーを押して下さい。
- ・新規プロジェクトを作成するときは、モデルファイル名を入力して、“Create Model File” ボタンを押すと、モデルファイルを作成します。既にモデルファイルがあるときは、ファイル名を入力し、“Load” ボタンを押し、読み込むこともできます。一般に“Load Model Data Automatically” チェックボタンは ON にします。（ON にすると、プロジェクトファイルを開いたときに、モデルファイルを自動で読み込みます。）
- ・結果ファイルの名前が入力されているとき、計算を行うとこのファイル名で結果ファイルを作成します。（結果ファイル名が入力されていないときは、計算を行うときに結果ファイルの名前を確認します。）“Load Result Data Automatically” チェックボタンが ON のときは、プロジェクトファイルを開いたときに結果ファイルがあれば、また読み込みに成功した場合は自動で読み込みます。（結果ファイルがあるにも関わらず、読み込まれない場合は“Load” ボタンを押して下さい。原因などが表示されます。）
- ・Project Info View 中の“P”、“H”、“A” または“Choose...” などのボタンは動作しません。

メインウィンドウ - Result Info View (結果ファイルを開いている時のみ)



- ・ 結果ファイルの情報を表示します。

Model Edit Window

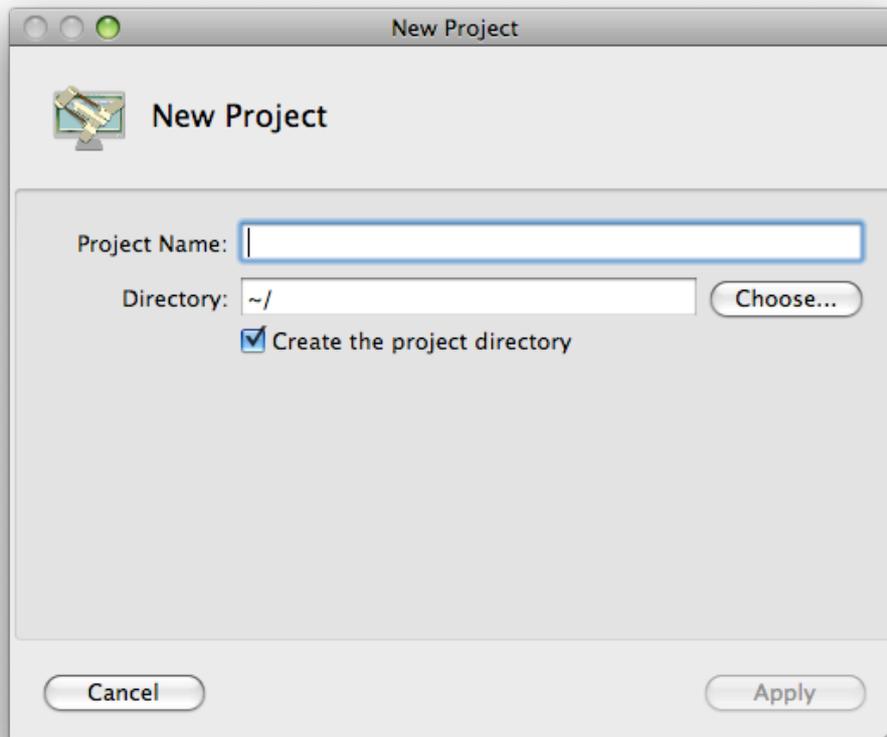


- ・ Card の追加などができます。

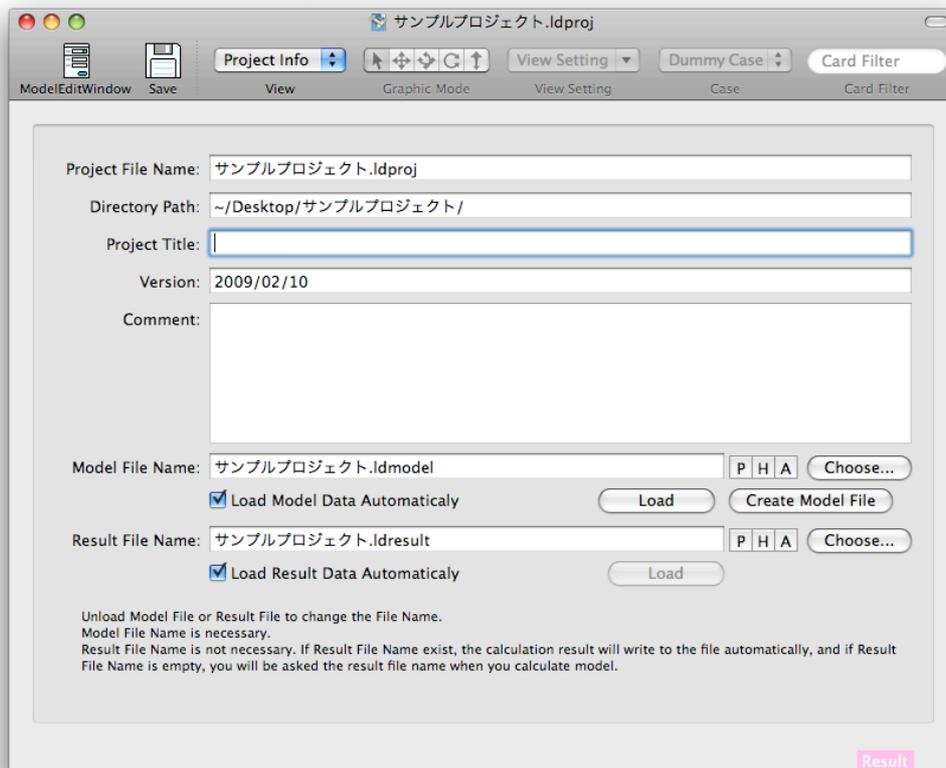
2. 新規プロジェクト、または新規モデルの作成

新規プロジェクトの作成

- ・メインメニューの“File” - “New Project...” を選択します。ウィンドウ “New Project” が開きます。



- ・“Project Name” テキストフィールドに、プロジェクトの名前 (プロジェクトファイルのファイル名になります。) を入力し、また “Directory” テキストフィールドの右の “Choose...” ボタンを押し、プロジェクトを作成するディレクトリを選択します。または、 “Directory” テキストフィールドにディレクトリのパスを入力します。 “Create the project directory” チェックボタンの ON/OFF を選択し、 “Apply” ボタンを押します。 “Create the project directory” チェックボタンが ON のときは、 “Directory” のパスのディレクトリ内に “Project Name” の名前のディレクトリが作成され、その中に “Project Name.lproj” のプロジェクトファイルが作成されます。 “Create the project directory” チェックボタンが OFF のときは、 “Directory” のパスのディレクトリ内に “Project Name” の名前のディレクトリの中に直接 “Project Name.lproj” のプロジェクトファイルが作成されます。そして、メインウィンドウ (View type: Project Info) が開きます。



- ・ “Project Title”, “Version”, “Comment” などのテキストフィールドにプロジェクトファイル、モデルのバージョン情報、コメント文字列を入力します。(すべて任意) (“Comment” テキストフィールド内で改行文字を入力するには、“option” キーを押しながら “return” キーを押して下さい。) また、“Model File Name” テキストフィールドにモデルファイルの名前 (またはファイルパス) を入力し、“Create Model File” ボタンを押します。“Model File Name.Idmodel” のモデルファイルが作成されます。(既にモデルファイルがあるときは、ファイル名を入力し、“Load” ボタンを押し、読み込むこともできます。“Result File Name” テキストフィールドに結果ファイルの名前 (またはファイルパス) を入力します。(空にしておくこともできます。) これが入力してある場合は、計算をするとこのファイル名で結果ファイルを作成します。
- ・ 上記の入力やモデルファイルの作成、読み込みが終われば、ツールバーにある “View” ポップアップメニューから “Card View” を選択します。または、メインメニューの “View” から選択することもできます。View が “Card View” に変わります。(“Graphic View” にすることもできます。(モデルが空の場合は “Graphic View” には何も表示されません。)) なお、ここでプロジェクトファイルに行った変更は現時点ではプロジェクトファイルに保存されていません。ツールバーの “Save” ボタン、またはメインメニューの “File” - “Save” からプロジェクトファイルとモデルファイルを保存できます。(プロジェクトファイルを開いているときは、プロジェクトファイルとモデルファイルが同時に保存されます。)

新規モデルの作成

- ・ 新規モデルを作成するには、メインメニューの “File” - “New Model” を選択します。空のモデルファイルが作成されます。

3. 既存プロジェクト、既存モデル、または結果ファイルの読み込み

- ・ 既存プロジェクト、既存モデル、または結果ファイルを開くときは、“Finder”上でダブルクリックして下さい。またはメインメニューの“File” - “Open...”を選択し、Openダイアログで開くファイルを選択して下さい。

4. モデルの作成 (モデルの入力, 編集)

- ・ 空白行を入力 (追加) するにはメインメニューの“Edit” - “Insert” (または“⌘” + “I”キー)を選択して下さい。行 (Card) が選択されていれば、その上に挿入されます。行が選択されていなければ、モデルデータの最後に追加されます。
- ・ Card Table の行の上でダブルクリック、または行が選択された状態で“return”キーを押すと、その行を変更できます。(直接入力) なお、Graphic View で Card (Node, Element など、Graphic で表示されているもの) をダブルクリックしたときなどは、ツールバー下に入力用のテキストフィールドが表示されます。
- ・ “Model Edit Window”を用いて Card を入力できます。追加するには Model Edit Window の “Action” ポップアップメニューで “Add” を選択します。行 (Card) が選択されていれば、その位置に追加されます。行が選択されていなければ、モデルデータの最後に追加されます。なお、“Model Edit Window”はツールバーのボタン、またはメインメニューの“Model” - “Open Model Edit Window”で表示できます。
- ・ 他のアプリケーション (テキストエディタ、表計算ソフトウェア、また各種プログラミング言語など) で作成したデータをペーストすることができます。テキストエディタでモデルデータをコピー (メインメニューの“編集” - “コピー”を選択) し、Load Design で挿入したい位置の行を選択した状態でペースト (メインメニューの“Edit” - “Paste”を選択) します。また、表計算ソフトウェアからペーストするときは、表計算ソフトウェアでコピーした後、Load Design で挿入したい位置の行を選択した状態で、“option”キーを押しながら“ペースト” (“option”キーを押しながらファイルメニューの“Edit” - “Paste as Tab-Separated”を選択、または“⌘” + “option” + “V”キー) します。(注: この操作 (“option”キーを押しながら“ペースト”) は、カラムが“Tab”で区切られたテキストデータ (下図参照) を Load Design にペーストするときに使用します。(多くの表計算ソフトでは、“Tab”で区切った文字列としてコピーされます。)) なお、Load Design から表計算ソフトウェアにモデルデータをコピーするときには、Load Design で“option”キーを押しながらコピーをし、表計算ソフトウェアでペーストすると、表計算ソフトでセル毎に分かれてペーストされます。(注: “option”キーを押しながら“コピー”または“切り取り”は、モデルデータの Card のカラムを“Tab”で区切った文字列をクリップボードにコピーします。)

	A	B	C	D	E	F	G
1	\ Nodes						
2	Node	Node_00_00_00		+0.000000E-03	-50.000000E-03	-25.000000E-03	
3	Node	Node_00_00_01		+0.000000E-03	-50.000000E-03	-14.500000E-03	
4	Node	Node_00_00_02		+0.000000E-03	-50.000000E-03	-4.000000E-03	
5	Node	Node_00_00_03		+0.000000E-03	-50.000000E-03	+4.000000E-03	
6	Node	Node_00_00_04		+0.000000E-03	-50.000000E-03	+14.500000E-03	
7	Node	Node_00_00_05		+0.000000E-03	-50.000000E-03	+25.000000E-03	
8	Node	Node_00_01_00		+0.000000E-03	-40.614927E-03	-25.000000E-03	
9	Node	Node_00_01_01		+0.000000E-03	-40.614927E-03	-14.500000E-03	
10	Node	Node_00_01_02		+0.000000E-03	-40.614927E-03	-4.000000E-03	
11	Node	Node_00_01_03		+0.000000E-03	-40.614927E-03	+4.000000E-03	
12	Node	Node_00_01_04		+0.000000E-03	-40.614927E-03	+14.500000E-03	
13	Node	Node_00_01_05		+0.000000E-03	-40.614927E-03	+25.000000E-03	
14	Node	Node_00_02_00		+0.000000E-03	-27.076618E-03	-25.000000E-03	
15	Node	Node_00_02_01		+0.000000E-03	-27.076618E-03	-14.500000E-03	
16	Node	Node_00_02_02		+0.000000E-03	-27.076618E-03	-4.000000E-03	
17	Node	Node_00_02_03		+0.000000E-03	-27.076618E-03	+4.000000E-03	

(表計算ソフトウェア上のモデルデータ)

- ・ ツールバーまたはメインメニューの “View Setting” メニューから “Model with Error” を選択すると、エラーが “Error / Warning” の列に表示されます。(“View Setting” の詳細については、“Card View” での結果の表示 (Result Table View での表示) と View Setting” の項を参照してください。) また、メインメニューの “Model” - “Find Error Card” - “Next Error Card/Previous Error Card” などを選択すると、次の(または前の) Error Card を選択します。エラーがある場合は修正する必要があります。

5. Card Filter、Case Menu について

- ・ “Card Filter” テキストフィールドに “type:Node” や “type:Element” などの文字列を入力することにより、Card View や Graphic View で表示する Card を制限することができます。Card Filter の文字列については、“Card Filter について” を参照してください。
- ・ “Case Menu” で計算ケースを指定すると、その Case で使われていない Card (Case Card, Restraint Card, 及び Force Card) が Card View や Graphic View で表示されなくなります。また Graphic View では、指定された Case の (“ForceSet” Card を用いている場合は、係数なども考慮した) 荷重条件が表示されます。また、Result Table View で、Case Menu に指定された Case の値が表示されます。

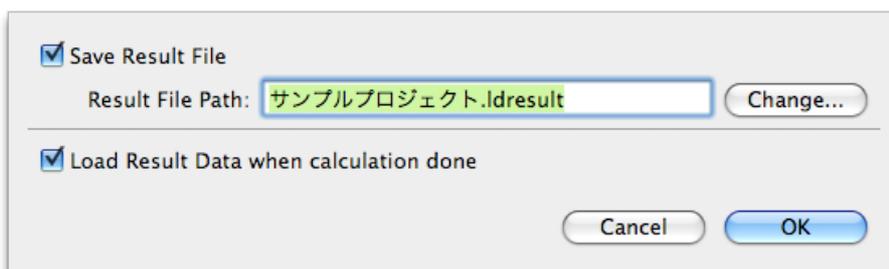
6. 計算

- ・ モデルのチェックをするには、メインメニューの “Calculation” - “Check Model” を選択します。
- ・ 計算を行うには、メインメニューの “Calculation” - “Run Calculation” を選択します。プロジェクトファイルを開いている場合は “Version” 文字列の確認のダイアログが表示されます。(モデルファイル

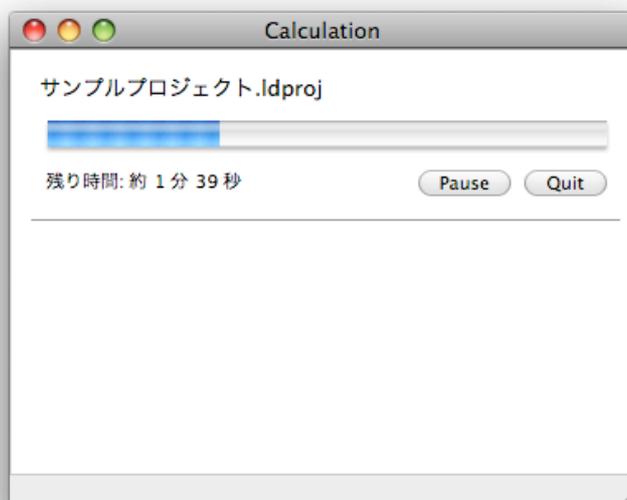
(拡張子 “.ldmodel” を直接開いている場合は表示されません。) ここで、プロジェクトの “Version” の文字列を変更できます。(これは、“Project Info” View にある “Version” と同じです。計算するときに、毎回確認され、変更、入力することができます。) “Save and Calculate” (保存して計算) または “Calculate without Save” (保存しないで計算) ボタンを押します。



- プロジェクトファイルに結果ファイルの名前が入力されているときは、計算を開始します。また、ここでプロジェクトファイルに結果ファイルの名前が入力されていない場合、またはモデルファイルを直接開いている場合は、ダイアログが表示され、結果ファイルの名前を入力します。(結果ファイルを保存しないことも選択できます。(注: 現時点では、後で結果ファイルを保存することはできません。)) また、計算終了後に、結果ファイルを読み込むかどうかを指定します。(結果ファイルを読み込むように指定しなかった場合は、計算終了後に手動で結果ファイルを読み込む必要があります。) “OK” ボタンを押すと計算を開始します。



- 計算が始まると、“Calculation” ウィンドウが表示され、計算の進捗状況が表示されます。計算の途中で、“Pause” (計算の一時停止) , “Quit” (計算を途中で終了) することができます。



- ・プロジェクトファイル、または計算時に結果を読み込むように設定した場合は、計算終了時に自動的に結果が読み込まれます。読み込むように設定していない場合は、メインメニューの“File” - “Load Result Data” を選択し読み込みます。結果を読み込んでいるときはウィンドウ下部のステータスバー右側に“Result”とピンク色で表示されます。読み込んでいないときは、薄いピンク色で表示されます。

7. Card View での結果の表示 (Result Table View での表示) と View Setting

- ・Card View で、ツールバーまたはメインメニューの“View Setting”メニューから、“Model”、“Model with Error”、または“Result”を選択すると View Setting を変更できます。“Result”にした場合は、Card View で選択している Card の結果 (Node の変位、Element の Von Mises 応力) が “Result Table View” に表示されます。(Result Table View は、初期状態では Card View の右側に閉じられています。Result Table View の開き方は“1. Load Design のインターフェース”の項を参照してください。)
- ・ツールバーまたはメインメニューの“View Setting”メニューから“View Setting...”を選択すると、Card View の“View Setting シート”が開きます。(“View Setting”メニュー中の“Model”、“Model with Error”、“Result”は、この View Setting の既定のセットです。) Card View に表示する列や Result Table View に表示する内容を変更できます。“Values”テキストフィールド内で改行文字を入力するには、“option”キーを押しながら“return”キーを押して下さい。
- ・Result Table View に表示する内容は、Result Table View 右上にある“Values...”ボタンを押して表示されるシートでも設定できます。
- ・Result Table View に表示する内容の指定方法 (“Values”の欄に記入する文字列) や、Result Table View についての詳細は、“Result Table View での表示”の項を参考にしてください。

8. Graphic View での結果の表示 と View Setting

- ・Graphic View で、ツールバーまたはメインメニューの“View Setting”メニューから、“Model”、“Displacement”、“VonMisesStress”または“Displacement & VonMisesStress”を選択すると View Setting を変更できます。“Displacement”、“VonMisesStress”などを選択した場合は、変形 (節点の変位) や Von Mises 応力による色を適用して表示します。
- ・ツールバーまたはメインメニューの“View Setting”メニューから“View Setting...”を選択すると、Card View の“View Setting シート”が開きます。(“View Setting”メニュー中の“Model”、“Displacement”、“VonMisesStress”、“Displacement & VonMisesStress”は、この View Setting の既定のセットです。) Card View に表示する列や Result Table View に表示する内容を変更できます。
- ・Graphic View では View Setting で Result Table View の設定はできませんが、Graphic View でも Result Table View を使用できます。Result Table View に表示する内容は、Result Table 右上にある“Values...”ボタンを押して表示されるシートで設定できます。Graphic View で選択している Card の結果などが Result Table View に表示されます。

- ・ Result Table View に表示する内容の指定方法 (“Values” の欄に記入する文字列) や、Result Table View についての詳細は、”Result Table View での表示” の項を参考にしてください。

9. Result Table View での表示

- ・ Result Table View には、計算結果などを表示します。ただし、計算結果以外の値も表示できます。指定された値を Case ごとに表示します。なお表示する Case はツールバーやメインメニューの “Case Menu” で設定します。(Case Menu については、”Card Filter、Case Menu について” の項も参照してください。) 複数の Case が指定されているときは、最大値、及び最小値も表示します。Case Menu で “---” を選択しているときは、Case には依存しない値を表示します。
- ・ Result Table View は、初期状態では Card View や Graphic View の右側に閉じられています。Result Table View の開き方は “1. Load Design のインターフェース” の項を参照してください。
- ・ 初期状態では Result Table View は Card View や Graphic View の右側 (左右に分割) にありますが、Result Table 右上にある “⇄ Split” ボタンを押すと Card View や Graphic View の下側 (上下に分割) に移動できます。再度、”⇄ Split” ボタンを押すと右側に戻ります。
- ・ Result Table View の表示内容は Result Table 右上にある “Values...” ボタンを押して表示されるシートで設定できます。(シート上の ”Values” テキストフィールド内で改行文字を入力するには、”option” キーを押しながら “return” キーを押して下さい。) また Card View に付属している Result Table View では、Card View の View Setting でも表示内容を設定できます。
- ・ Result Table View の表示内容は、Card Value Expression のリスト (Card Value Expression List) で指定します。”例 2” のように、Card Filter を指定し、選択している Card により表示する値 (Card Value Expression) を変更することができます。Card Value Expression List の詳細は “Card Value Expression List について” を参照して下さい。

\ 例 1 - 複数の Card Value Expression (カンマ区切り / 最後はセミコロン ‘;’)

```
VonMisesStress, RodForce;
```

\ 例 2 - Card Filter で、選択している Card により表示する値を変更する場合

```
 #(type:Node) : Displacement;  
 #(type:Element) : VonMisesStress, ElementNodeForce;  
 #(class:Rod) : RodForce;
```

- ・ メインメニューの “Edit” - “Copy” で、Result Table View の選択してある行をクリップボードにコピーできます。タブ区切りでコピーするには、’option’ キーを押しながら、メインメニューの “Edit” - “Copy as Tab-Separated” でコピーします。

Card Filter について

1. Card Filter について

- Card Filter は、メインウィンドウの Card View や Graphic View に表示する Card を制限するときに用います。結果などの値を計算または表示する場合に、計算/表示を行う Card を制限するときに用いることもあります。
- Card Filter は Card Filter Function と演算子 (“&” (論理積)、 “|” (論理和)、 “!” (否定)) 及び “()” (括弧) を組み合わせて指定します。
- Card Filter Function には引数を用いないもの、1個用いるもの、2個用いるものなどがあります。書式は次の通りです。引数を持つ Card Filter Function は、引数の前に ‘:’ (コロン) があります。文字 ‘:’ も Card Filter Function の一部です。(‘:’ の前に空白文字などを入れることはできません。) なお、引数の文字列内に特殊な文字を含む場合は、引数を ‘”’ (ダブルコーテーション) で囲みます。

hasError	\ Function: “hasError” – 引数なし
type:Node	\ Function: “type:” – 引数 1個 (“Node”)
column:2 contain:”_x00_”	\ Function: “column:contain:” – 引数 2個 (“2” と “_x00_”)

- Card Filter Function には、次のものがあります。

書式	説明
type:<CardType>	Card Type が指定された Card Type のとき YES (Card Type には “Node”, “Element” または “Elem”, “ElementProp”, または “ElemProp”, “Case”, “Restraint”, “Force”, “Null” を指定します。) 例: type:Elem
class:<CardClass>	Card Class が指定された Card Class のとき YES 例: class:Hexa
hasError	Card Type が Error Card、またはその他のエラーがあるとき YES 例: hasError
case:<CaseName>	指定された Case で使われる Card のとき YES (すべての Node や Element は YES) 例: case:case_01

書式	説明
caseArray:{<CaseName1>, <CaseName2>, ... }	指定された Case (複数) のいずれかで使われる Card のとき YES (すべての Node や Element は YES) 例: caseArray:{ case_01, case_02, case_03 }
contain:<文字列>	指定された文字列を含むとき YES 例: contain:"node_x00_"
column:<Column番号> contain:<文字列>	指定された Column が、指定された文字列を含むとき YES 例: column:2 contain:"_x00_"

- 演算子 (“&” (論理積)、“|” (論理和)、“!” (否定)) は次のように使用します。“&” や “|” は 2個以上の Card Filter をつなげることができます。

演算子	書式	説明
& (論理積)	<CardFilter1> & <CardFilter2> <CardFilter1> & <CardFilter2> & <CardFilter3>	“&” で連なったすべての Card Filter が YES のとき YES
(論理和)	<CardFilter1> <CardFilter2> <CardFilter1> <CardFilter2> <CardFilter3>	“ ” で連なった Card Filter のうち 1つでも YES であれば YES
! (否定)	!<CardFilter>	Card Filter が NO であれば YES

- 括弧 “()” で囲まれた範囲は先に計算されます。
- 演算子 (“&” (論理積)、“|” (論理和)、“!” (否定)) と括弧 “()” の優先順位は次の通りです。(注: “&” と “|” の優先順位は同じです。同じ階層で “&” と “|” が共存することはできません。先に計算する方を括弧で囲む必要があります。)

(優先順位:高) 括弧 “()” > 否定 “!” > 論理積 “&” 及び 論理和 “|” (優先順位:低)

- Card Filter の例を示します。

hasError	\ エラーを持つ Card
type:Node	\ Card Type: Node Card Type の Card
column:2 contain:"_x00_"	\ 2 番目の Column が文字列 “_x00_” を含む Card
type:Elem & !class:Rod	\ Class: Rod 以外の Element Card Type の Card
hasError & (type:Node type:Elem)	\ Node Card Type または Element Card Type でエラーを含む Card

Card Value Expression について

1. Card Value Expression の概要

- Card Value Expression は Card から直接得ることができる値 (Node の位置、変位、Element の応力など) や、それらの値を含む計算の式です。例えば次のようなものです。Card や Case が指定されると、実際の値が計算されます。(Case の指定は、計算結果などを使用する値のときなどに必要です。)

VonMisesStress	\ Card から直接得る値
SectionArea * YoungsModulus	\ 簡単な演算を含む式 (断面積 × ヤング率)

- Graphic View で、Color Scale を用いて要素の応力などの値を表示するときや、Result Table View で Card 毎の値を表示するとき用います。(Result Table View では Card Value Expression List を指定します。Card Value Expression List は、複数の Card Value Expression のリストです。詳細は後の “Card Value Expression List について” の項を参照してください。)
- Card Value Expression が返す値 (Card Value) は、スカラーではなく、複数の値の配列の場合があります。例えば Node の変位は { Tx, Ty, Tz, Rx, Ry, Rz } (並進自由度 Tx, Ty, Tz, 回転自由度 Rx, Ry, Rz) の 6 個の値を持ちます。また Element の Von Mises 応力は、Element の複数の位置での値を含む配列です。複数の位置がベクトルなどを持つ場合もあります。詳細は後の “Card Value Type について” 及び “Card Value Expression について” の項を参照してください。

2. Card Value Type について

- Card Value Type は、Card から直接得ることができる値のタイプ(名称)です。Card から得られる Card Value Type は Card の Class などにより異なります。例えば、Node は “Location” (Node の位置), “Displacement” (変位), Rod は “SectionArea” (断面積), “VonMisesStress” (Von Mises 応力) などを返すことができます。
- Card Value Type の値は、(単一の値ではなく) 複数の値の配列の場合があります。(単一の値の場合もあります。) 最大 3次元の配列で、Point、Section Point、Vector Value を次元に持ちます。各次元の大きさは、”Number of Points”、”Number of Section Points”、”Number of Vector Values” です。Number of Points、Number of Section Points、Number of Vector Values は Card Class や Card Value Type などにより異なります。Card Value の値の総数は (“Number of Points” × “Number of Section Points” × “Number of Vector Values”) です。例えば、Hexa の Card Value Type: “ElementStress” (要素の応力) は、各 Point (Node 8点) につき、Vector Value: σ_x , σ_y , σ_z , σ_{yz} , σ_{zx} , σ_{xy} (6 個の値) の、合計 48 個の値を返します。

参考:

C言語の

```
double aCardValue[NumberOfPoints][NumberOfSectionPoints][NumberOfVectorValues];
```

で定義された配列に似ています。

次元	説明
Point	<p>Element Card などで、Element 上の位置により変わる値などで、位置を表します。一般的には、要素を構成する Node の位置を表しますが、応力評価点などを指定した場合などは要素を構成する Node の位置と異なる場合があります。</p> <p>Element Card 以外の Card では、一般的にこの次元はありません。(Number of Points = 1)</p> <p>Card Class や Card Value Type により異なります。該当する Card Class の項を参照して下さい。</p> <p>例: Hexa 要素で Card Value Type: VonMisesStress の Card Value において、Number of Points は 8 (要素を構成する Node の数に一致) で、各 Point での値は、要素(Hexa)を構成する Node の位置での Von Mises 応力です。</p>
Section Point	<p>多くの場合、この次元はありません。(Number of Section Points = 1)</p> <p>一般的に Quad Card (シェル要素) の表面/裏面など、同じ Point (Node) 上で複数の応力評価点などがある場合の位置などを表します。</p> <p>Card Class や Card Value Type により異なります。該当する Card Class の項を参照して下さい。</p> <p>例: Quad 要素で Card Value Type: VonMisesStress@shellFace の Card Value において、Number of Section Points は 2 (なお、Number of Points は 4 (要素を構成する Node の数に一致)) で、各 Section Point での値は、表面/裏面での Von Mises 応力です。</p>
Vector Value	<p>Card Value Type により Vector Value の数が決まります。Card (Card Class など) には依存しません。</p> <p>例: Card Value Type: Displacement (変位) の Number of Vector Values は 6 (並進自由度 Tx, Ty, Tz, 回転自由度 Rx, Ry, Rz) です。</p> <p>Card Value Type: VonMisesStress (Von Mises 応力) や SectionArea (断面積) などの Number of Vector Values は 1 (スカラー値) です。</p> <p>Card Value Type: ElementStress (要素の応力) の Number of Vector Values は 6 ($\sigma_x, \sigma_y, \sigma_z, \sigma_{yz}, \sigma_{zx}, \sigma_{xy}$) です。</p>

- ・ Card Value Type は Major Card Value Type (“VonMisesStress” など) のみ、または Major Card Value Type と Minor Card Value Type (“@shellFace” など) を組み合わせて指定します。多くの場合 Major Card Value Type のみで良いですが、一部 Minor Card Value Type を指定する必要がある場合があります。例えば、Quad 要素では、表面と裏面で要素の応力が異なるため、“ElementStress” (要素の応力 $\sigma_x, \sigma_y, \sigma_z, \sigma_{yz}, \sigma_{zx}, \sigma_{xy}$) を得るには “ElementStress@shellFace” と指定します。なお、Quad 要素では、“VonMisesStress” (Von Mises 応力) も表面と裏面で異なりますが、単に “VonMisesStress” と指定した場合は表面/裏面での値の最大値が返ります。(詳細は Card ごとの説明を参照して下さい。)
- ・ Card が返すことができる Card Value Type は Card Class に依存します。Card Class 毎の Card Value Type については、Card Class の項を参照して下さい。

3. Card Value Expression について

- ・ Card Value Expression は Card Value Type などより構成された式で、“+” (加算)、“-” (減算)、* (乗算)、/ (除算) や“()” (括弧) などを使用することもできます。例えば次のようなものがあります。(現時点で、演算の種類は加減乗除などの簡単な演算のみです。)

VonMisesStress	\ 単一の Card Value Type
SectionArea * YoungsModulus	\ 簡単な演算を含む式 (断面積 × ヤング率)

- ・ “+”、“-”、“*”、“/” (加減乗除) の計算では、演算子の前後の Card Value の Number of Points、Number of Section Points、Number of Vector Values は各々、一致するか、1 である必要があります。一方が 1 で他方が 1 以外の場合は、それが他方 (1 ではない方) の大きさに拡張され、それぞれの成分ごとに “+”、“-”、“*”、“/” の演算が行われます。

4. Card Value Expression List について

- ・ Card Value Expression List は、複数の Card Value Expression のリストです。対象となる Card を制限する Card Filter を指定することもできます。Result Table View で Card 毎の値を表示するときに用います。例を下に示します。(Card Value Expression List は複数の行になることがあります。下の “例 2” では、2行が Card Value Expression List です。

\ 例 1

\ 複数の Card Value Expression

```
VonMisesStress, RodForce;
```

\ 例 2

\ Card Filter で対象の Card により表示する値を変更

\ (Card Type が Node のときは Displacement, Element のときは VonMisesStress)

```
 #(type:Node) : Displacement;
```

```
 #(type:Element) : VonMisesStress;
```

- Card Value Expression List は次の 2通りの書式を連ねたものです。最後の文字はセミコロン ';' です。'#(<Card Filter>) : ' 内、及び <Card Value Expression> 内以外の場所では任意の位置で改行することができます。(改行は影響しません。) なお、Card Value Expression はカンマ ';' で区切り、複数続けることができます。

\ 書式 1 - Card Filter なし

```
<CardValueExpression-1>, <CardValueExpression-2>;
```

\ 書式 2 - Card Filter あり

```
#( <CardFilter> ) : <CardValueExpression-1>, <CardValueExpression-2>;
```

- "書式 1" の Card Filter がない場合、その後 (';' まで) の Card Value Expression は任意の Card に対して有効です。"書式 2" の Card Filter がある場合、その後 (';' まで) の Card Value Expression は <Card Filter> に適合した Card に対して有効です。複数の行 (';' で区切られたブロック) がある場合は、各ブロックで有効な Card Value Expression がすべて有効になります。

Card Class

1. それぞれの Card Class の概要

以下の Card Class があります。

Card Type	Card Class	説明
Node	Node	節点を定義します。
Element	Rod	ロッド要素を定義します。
	Quad	四辺形シェル要素を定義します。
	Hexa	六面体ソリッド要素を定義します。
Element Prop	RodProp	ロッド要素のプロパティを定義します。
	ShellProp	シェル要素のプロパティを定義します。
	SolidProp	ソリッド要素のプロパティを定義します。
	Material	要素の材料特性を定義します。
Case	Case	計算ケースを定義します。
Restraint	RestraintSet	複数の名前の拘束条件を集めた拘束条件を定義します。
	FixNode	節点の複数の自由度を拘束する拘束条件を定義します。
	ForceDisp	節点の自由度の強制変位の拘束条件を定義します。
	RestraintEquation	拘束方程式（自由度間の方程式）を定義します。
Force	ForceSet	複数の名前の荷重条件を集めた荷重条件を定義します。
	Force	節点に荷重を加える荷重条件を定義します。

2. Column Type について

Card の各 Column は、次の Column Type のいずれかです。

Column Type	説明
Card 名	定義する Card の Card 名（Column No: 2）や、他の Card を参照するときの Card 名を指定する時などに使用します。 例: node_120_r60_40 n01234567 01234567

Column Type	説明
実数	<p>実数値。節点の座標、材料特性、荷重などの値を指定するときに使用します。</p> <p>例: +12.34 -0.1234 12.34E+06</p>
真偽値 (BOOL)	<p>真偽値。YES (“1” または “YES”)、NO (“0” または “NO”) を指定します。FixNode Card で拘束の ON / OFF を指定するときなどに使用します。</p> <p>Column の文字列: YES (ON) の場合 : “1” または “YES” NO (OFF) の場合 : “0” または “NO”</p>
自由度	<p>自由度。拘束する自由度など自由度を指定する時に使用します。Node 名と方向 (Tx, Ty, Tz, Rx, Ry, Rz) で指定します。Node 名と方向 (TX など (大文字)) の間に文字 ‘@’ を入れます。</p> <p>例: node_01@TX node_120_r60_40@RZ</p>
文字列	<p>Card で指定された文字列を指定するときに使用します。</p> <p>例: LINEAR_STATIC</p>

なお、Card名 には、'a-z'、'A-Z'、'0-9'、'_' (アンダーバー)、および全角文字などが使えます。その他の半角文字 (! " # \$ % & ' () * + , - . / : ; < = > ? @ [\] ^ ` { | } ~ や ' (スペース)) などは使えません。

3. それぞれの Card Class

次頁以降

Node

(Card Type: Node Card Type)

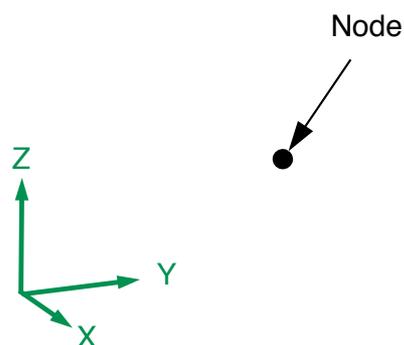
節点の定義

\ フォーマット

Node , <Node Name> , , <Location X> , <Location Y> , <Location Z> ,

\ 例

Node , node_01 , , 100.0 , -20.0 , 50.0 ,



Column No.	Column	Column Type	説明
1	"Node"		Card Class 名
2	<Node Name>	Card 名	定義する Node の Card 名
3	空白		
4	<Location X>	実数	X 座標
5	<Location Y>	実数	Y 座標
6	<Location Z>	実数	Z 座標

説明 :

- ・ Node 名: <Node Name> の節点を定義します。
- ・ <Node Name> は Element Card, Force Card, Restraint Card から参照されます。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
Location	<なし>	1・1・3	節点の座標 { X , Y , Z }
Displacement	<なし>	1・1・6	節点の変位 { Tx , Ty , Tz , Rx , Ry , Rz }

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

Rod

(Card Type: Element Card Type)

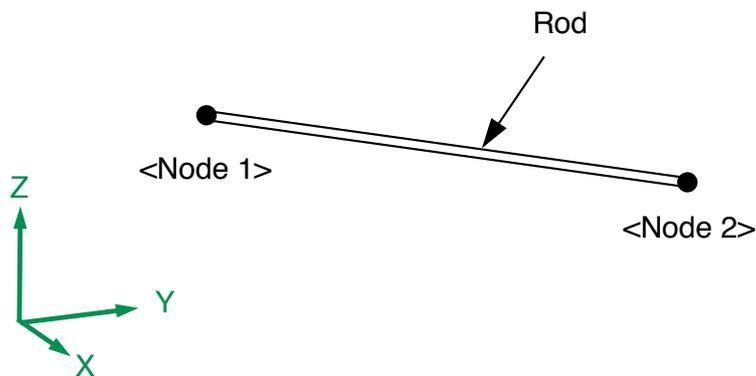
ロッド要素の定義

\ フォーマット

Rod , <Element Name> , <RodProp Name> , <Node Name 1> , <Node Name 2> ,

\ 例

Rod , rod_01 , rodProp_01 , node_01 , node_02 ,



Column No.	Column	Column Type	説明
1	"Rod"		Card Class 名
2	<Element Name>	Card 名	定義する Rod の Card 名
3	<RodProp Name>	Card 名	RodProp Card 名 (Rod のプロパティ)
4	<Node Name 1>	Card 名	Node 1 の Card 名
5	<Node Name 2>	Card 名	Node 2 の Card 名

説明 :

- ・ Element 名: <Element Name>、両端の節点: <Node Name 1>, <Node Name 2> のロッド要素を定義します。
- ・ Rod 要素は軸方向(引張り/圧縮)の剛性を持ちます。曲げ剛性、せん断剛性、ねじり剛性は持ちません。
- ・ ロッドの断面積、材料特性は Card 名: <RodProp Name> の Card を参照します。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
Length	<なし>	1・1・1	Rod の長さ (節点間距離)
SectionArea	<なし>	1・1・1	Rod の断面積
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν
RodForce	<なし>	2・1・1	Rod に加わっている力 軸力 (+ : 引張り / - : 圧縮) (計算点: 節点位置)
VonMisesStress	<なし>	2・1・1	Von Mises 応力 (計算点: 節点位置)
ElementStress	<なし>	2・1・6	要素内応力 { $\sigma_x, \sigma_y, \sigma_z, \sigma_{yz}, \sigma_{zx}, \sigma_{xy}$ } (計算点: 節点位置)
ElementStrain	<なし>	2・1・6	要素内歪み { $\varepsilon_x, \varepsilon_y, \varepsilon_z, \varepsilon_{yz}, \varepsilon_{zx}, \varepsilon_{xy}$ } (計算点: 節点位置)
ElementNodeForce	<なし>	2・1・6	要素が節点より受けている力 { $F_x, F_y, F_z, M_x, M_y, M_z$ }

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

- ・ Card Value Type “ElementStress”, “ElementStrain” の座標系は要素座標系です。
- ・ Card Value Type “ElementNodeForce” の座標系は全体座標系です。
- ・ 要素座標系は、x軸が <Node 1> から <Node 2> の向きです。(y軸、z軸は x-y-z が右手系をなす任意の向きです。)
- ・ Card Value Type “ElementStress” は σ_x 以外すべて 0 (ゼロ) です。
- ・ Card Value Type “ElementStrain” は $\varepsilon_{yz}, \varepsilon_{zx}, \varepsilon_{xy}$ は 0 (ゼロ) で、 $\varepsilon_y = \varepsilon_z = -\nu \cdot \varepsilon_x$ です。(ここで ν はポアソン比)

Column No.	Column	Column Type	説明
7	<Node Name 4>	Card 名	Node 4 の Card 名

説明：

- ・ Element 名: <Element Name>、4つの頂点の節点: <Node Name 1>, <Node Name 2>, <Node Name 3>, <Node Name 4> の四辺形シェル要素を定義します。なお Node 1, 2, 3, 4 の位置は、図のように Quad 要素を一周するようになっている必要があります。
- ・ 板厚、材料特性は Card 名: <ShellProp Name> の Card を参照します。
- ・ すべての節点が同一面内でない場合、曲面の形状は双一次曲面になります。
- ・ 節点の法線まわりの回転剛性は基本的にはありません。(ただし、剛性方程式を解く時にエラーが起きないように小さな剛性が加えられています。) 節点に法線まわりの大きなモーメントが加わらないようにする必要があります。

対応する Card Value Type：

Major Card Value Type	Minor Card Value Type	Size	説明
Thickness	<なし>	1.1.1	板厚 t
YoungsModulus	<なし>	1.1.1	ヤング率 E
PoissonsRatio	<なし>	1.1.1	ポアソン比 ν
ShellForce	<なし>	4.1.9	Shell に加わっている力 { Fx/l, Fy/l, Fz/l, Fyz/l, Fzx/l, Fxy/l, Mx/l, My/l, Mxy/l } (計算点: 節点位置)
VonMisesStress	<なし>	4.1.1	Von Mises 応力 - 両面の最大値 (計算点: 節点位置)
	@shellFace	4.2.1	Von Mises 応力 - 表面/裏面 (計算点: 節点位置)
ElementStress	@shellFace	4.2.6	要素内応力 - 表面/裏面 { $\sigma_x, \sigma_y, \sigma_z, \sigma_{yz}, \sigma_{zx}, \sigma_{xy}$ } (計算点: 節点位置)
ElementStrain	@shellFace	4.2.6	要素内歪み - 表面/裏面 { $\varepsilon_x, \varepsilon_y, \varepsilon_z, \varepsilon_{yz}, \varepsilon_{zx}, \varepsilon_{xy}$ } (計算点: 節点位置)
ElementNodeForce	<なし>	4.1.6	要素が節点より受けている力 { Fx, Fy, Fz, Mx, My, Mz }

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

- ・ Card Value Type “ElementStress”, “ElementStrain” の座標系は要素座標系です。
- ・ Card Value Type “ElementNodeForce” の座標系は全体座標系です。
- ・ 面上の任意の位置での要素座標系は、<Node 1, 4 の中点> から <Node 2, 3 の中点> へのベクトル Ref Vector において、
 - z 軸：法線方向 - Node 1, 2, 3, 4 で右ネジが進む向きを正
 - x 軸：z 軸に垂直で z 軸 と Ref Vector を含む面内のベクトル (Ref Vector に近い向き)
 - y 軸：z 軸 (正方向) のベクトルと Ref Vector ベクトルの外積の向きとする座標系です。
- ・ Card Value Type “ElementStress”, “ElementStrain” は Minor Card Value Type “@shellFace” の指定が必要です。
- ・ Minor Card Value Type “@shellFace” を指定した時、Section Point: 1 は要素座標系 z 軸 - 側 (マイナス側) / Section Point: 2 は同 z 軸 + 側 (プラス側) の表面での値です。
- ・ Card Value Type “VonMisesStress” で Minor Card Value Type “@shellFace” を指定しないときは、両側の表面の応力の大きい方の値になります。

Hexa

(Card Type: Element Card Type)

六面体要素の定義

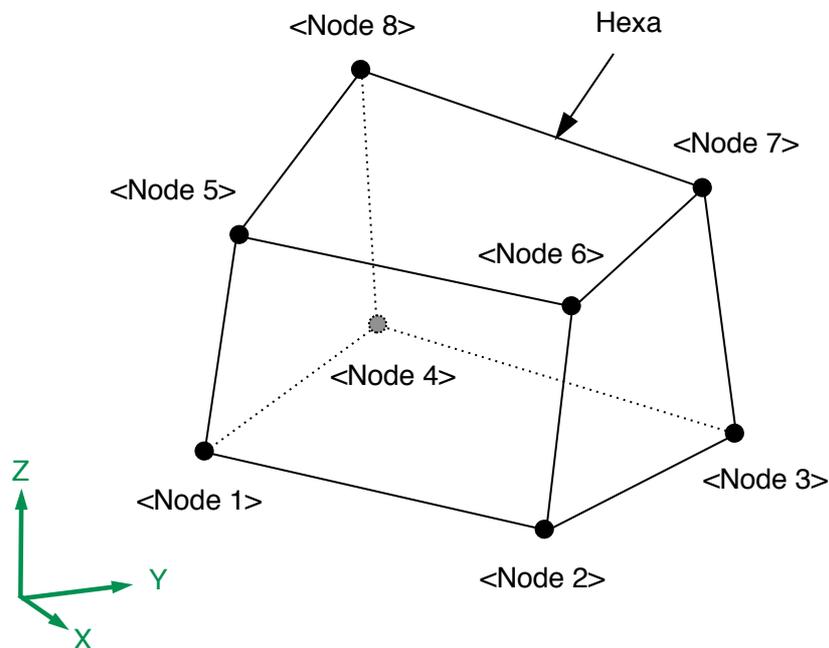
\ フォーマット

Hexa , <Element Name> , <SolidProp Name> , <Node Name 1> , <Node Name 2> ,
 , <Node Name 3> , <Node Name 4> , <Node Name 5>
 , <Node Name 6> , <Node Name 7> , <Node Name 8> ,

\ 例

Rod , rod_01 , rodProp_01 , node_01 , node_02 , node_03 , node_04 , node_05 , node_06
 , node_07 , node_08 ,

"フォーマット", "例" の項で複数行に分かれています、実際には一つの Card を複数行に分けることはできません。



Column No.	Column	Column Type	説明
1	"Hexa"		Card Class 名
2	<Element Name>	Card 名	定義する Hexa の Card 名
3	<SolidProp Name>	Card 名	SolidProp Card 名 (Solid のプロパティ)
4	<Node Name 1>	Card 名	Node 1 の Card 名
5	<Node Name 2>	Card 名	Node 2 の Card 名
6	<Node Name 3>	Card 名	Node 3 の Card 名
7	<Node Name 4>	Card 名	Node 4 の Card 名

Column No.	Column	Column Type	説明
8	<Node Name 5>	Card 名	Node 5 の Card 名
9	<Node Name 6>	Card 名	Node 6 の Card 名
10	<Node Name 7>	Card 名	Node 7 の Card 名
11	<Node Name 8>	Card 名	Node 8 の Card 名

説明：

- ・ Element 名: <Element Name>、8つの頂点の節点: <Node Name 1>, <Node Name 2>, <Node Name 3>, ... , <Node Name 8> の六面体要素を定義します。なお Node 1, 2, 3, 4, 5, 6, 7, 8 の位置は、図のように、Node 1, 2, 3, 4、Node 5, 6, 7, 8 が Hexa 要素の対する面を一周し、また Node 5, 6, 7, 8 は Node 1, 2, 3, 4 の右ネジの進む向き側に必要があります。
- ・ 材料特性は Card 名: <SolidProp Name> の Card を参照します。

対応する Card Value Type：

Major Card Value Type	Minor Card Value Type	Size	説明
Volume	<なし>	1・1・1	Hexa の体積
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν
VonMisesStress	<なし>	2・1・1	Von Mises 応力 (計算点: 節点位置)
ElementStress	<なし>	2・1・6	要素内応力 { $\sigma_x, \sigma_y, \sigma_z, \sigma_{yz}, \sigma_{zx}, \sigma_{xy}$ } (計算点: 節点位置)
ElementStrain	<なし>	2・1・6	要素内歪み { $\varepsilon_x, \varepsilon_y, \varepsilon_z, \varepsilon_{yz}, \varepsilon_{zx}, \varepsilon_{xy}$ } (計算点: 節点位置)
ElementNodeForce	<なし>	2・1・6	要素が節点より受けている力 { $F_x, F_y, F_z, M_x, M_y, M_z$ }

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

- ・ Card Value Type “ElementStress”, “ElementStrain” の座標系は全体座標系です。(要素座標系は全体座標系に一致します。)

- ・ Card Value Type “ElementNodeForce” の座標系は全体座標系です。

RodProp

(Card Type: Element Prop Card Type)

ロッドのプロパティの定義

\ フォーマット

RodProp , <RodProp Name> , <Material Name> , <Section Area> ,

\ 例

RodProp , rodProp_01 , material_01 , 10.0 ,

Column No.	Column	Column Type	説明
1	"RodProp"		Card Class 名
2	<RodProp Name>	Card 名	定義する RodProp の Card 名
3	<Material Name>	Card 名	Material Card 名 (材料特性)
4	<Section Area>	実数	ロッドの断面積

説明 :

- ・ Element Prop 名: <RodProp Name> のロッドのプロパティを定義します。
- ・ <RodProp Name> は Rod Card から参照されます。
- ・ 材料特性は Card 名: <Material Name> の Card を参照します。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
SectionArea	<なし>	1・1・1	Rod の断面積
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

ShellProp

(Card Type: Element Prop Card Type)

シェルのプロパティの定義

\ フォーマット

ShellProp , <ShellProp Name> , <Material Name> , <Thickness> ,

\ 例

ShellProp , shellProp_01 , material_01 , 0.10 ,

Column No.	Column	Column Type	説明
1	"ShellProp"		Card Class 名
2	<ShellProp Name>	Card 名	定義する ShellProp の Card 名
3	<Material Name>	Card 名	Material Card 名 (材料特性)
4	<Thickness>	実数	シェルの板厚 t

説明 :

- ・ Element Prop 名: <RodProp Name> のシェルのプロパティを定義します。
- ・ <ShellProp Name> は Quad Card から参照されます。
- ・ 材料特性は Card 名: <Material Name> の Card を参照します。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
Thickness	<なし>	1・1・1	シェルの板厚 t
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

SolidProp

(Card Type: Element Prop Card Type)

ソリッドのプロパティの定義

\ フォーマット

SolidProp , <SolidProp Name> , <Material Name> ,

\ 例

SolidProp , solidProp_01 , material_01 ,

Column No.	Column	Column Type	説明
1	"SolidProp"		Card Class 名
2	<SolidProp Name>	Card 名	定義する SolidProp の Card 名
3	<Material Name>	Card 名	Material Card 名 (材料特性)

説明 :

- ・ Element Prop 名: <SolidProp Name> のソリッドのプロパティを定義します。
- ・ <SolidProp Name> は Hexa Card から参照されます。
- ・ 材料特性は Card 名: <Material Name> の Card を参照します。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

Material

(Card Type: Element Prop Card Type)

* 材料特性の定義 *

\ フォーマット

Material , <Material Name> , <Youngs Modulus> , <Poissons Ratio> ,

\ 例

Material , material_01 , 210.0E+09 , 0.3 ,

Column No.	Column	Column Type	説明
1	"SolidProp"		Card Class 名
2	<SolidProp Name>	Card 名	定義する Material の Card 名
3	<Material Name>	実数	ヤング率 E
4	<Poissons Ratio>	実数	ポアソン比 ν

説明 :

- ・ Element Prop 名: <Material Name> の材料特性を定義します。
- ・ <Material Name> は ElementProp Card (RodProp, ShellProp, SolidProp) から参照されます。

対応する Card Value Type :

Major Card Value Type	Minor Card Value Type	Size	説明
YoungsModulus	<なし>	1・1・1	ヤング率 E
PoissonsRatio	<なし>	1・1・1	ポアソン比 ν

Size は順に <Number of Points>・<Number of Section Points>・<Number of Vector Values> です。

Case

(Card Type: Case Card Type)

計算ケースの定義

\ フォーマット

Case , <Case Name> , <Analysis Type> , <Restraint Name> , <Force Name> ,

\ 例

Case , case_01 , LINEAR_STATIC , restraint_01 , force_01 ,

Column No.	Column	Column Type	説明
1	"Case"		Card Class 名
2	<Case Name>	Card 名	定義する Case の Card 名
3	<Analysis Type>	文字列	解析のタイプ "LINEAR_STATIC" (線形静解析) のみ
4	<Restraint Name>	Card 名	適用する Restraint の Card 名 (拘束条件)
5	<Force Name>	Card 名	適用する Force の Card 名 (荷重条件)

説明：

- ・ Case 名: <Case Name> の計算ケースを定義します。
- ・ <Analysis Type> は "LINEAR_STATIC" (線形静解析) のみです。
- ・ モデルデータ内で定義されているすべての Element Card が計算対象になります。
- ・ <Restraint Name> の拘束条件、<Force Name> の荷重条件を適用します。

対応する Card Value Type : なし

ForceSet

(Card Type: Force Card Type)

荷重セットの定義

\ フォーマット

ForceSet , <Force Name> , <Factor> , <Force Name-child> ,

\ 例

ForceSet , force_101 , 1.50 , force_01 ,

Column No.	Column	Column Type	説明
1	"ForceSet"		Card Class 名
2	<Force Name>	Card 名	定義する Force の Card 名
3	<Factor>	実数	荷重 <Force Name-child> の係数
4	<Force Name-child>	Card 名	係数<Factor> を乗じられる Force の Card 名

説明 :

- ・ Force 名: <Force Name> の荷重を定義します。
- ・ Case Card、または他の "ForceSet" Card から参照されます。
- ・ Case Card、または他の "ForceSet" Card で指定されたとき、荷重 <Force Name-child> に係数 <Factor> を荷重が適用されます。
- ・ 複数の同じ Card 名の Force Card が存在することができます。"ForceSet" Card と "Force" Card が同じ名前でもかまいません。
- ・ <Force Name-child> に、別の "ForceSet" Card の (または "含む") Force Card 名を指定できます。ただし、循環参照 (ある Card 名の "ForceSet" Card から参照される Force Card ("ForceSet" Card) をたどっていったときに、元の Card 名が参照される) してはいけません。
- ・ 次のようにすると、荷重 force_Acceleration_x, force_Acceleration_y, force_Acceleration_z, ... に係数を加えた荷重として、荷重 force_case_Braking を定義できます。

\ 荷重 : force_case_Braking

ForceSet , force_case_Braking , -9.80665 , force_Acceleration_x ,
ForceSet , force_case_Braking , 0.0 , force_Acceleration_y ,
ForceSet , force_case_Braking , -9.80665 , force_Acceleration_z ,
...

\ ax = -9.80665
\ ay = 0.0
\ az = -9.80665
\ その他の荷重

対応する Card Value Type : なし

Force

(Card Type: Force Card Type)

節点荷重の定義

\ フォーマット

Force , <Force Name> , <Node Name> , , <Fx> , <Fy> , <Fz> , <Mx> , <My> , <Mz> ,

\ 例

Force , force_01 , node_01 , , 100.0 , 200.0 , -300.0 , , , ,

Column No.	Column	Column Type	説明
1	"Force"		Card Class 名
2	<Force Name>	Card 名	定義する Force の Card 名
3	<Node Name>	Card 名	荷重を加える Node の Card 名
4	空白		
5	<Fx>	実数	荷重 Fx
6	<Fy>	実数	荷重 Fy
7	<Fz>	実数	荷重 Fz
8	<Mx>	実数	モーメント荷重 Mx
9	<My>	実数	モーメント荷重 My
10	<Mz>	実数	モーメント荷重 Mz

説明 :

- ・ Force 名: <Force Name> の荷重を定義します。
- ・ Case Card、または "ForceSet" Card から参照されます。
- ・ 節点 <Node Name> に、荷重 Fx, Fy, Fz, Mx, My, Mz を加えます。Fx ~ Mz の値が 0.0 の場合は空白にすることができます。

対応する Card Value Type : なし

RestraintSet

(Card Type: Restraint Card Type)

拘束セットの定義

\ フォーマット

RestraintSet , <Restraint Name> , <Restraint Name-child> ,

\ 例

RestraintSet , restraint_101 , restraint_01 ,

Column No.	Column	Column Type	説明
1	"RestraintSet"		Card Class 名
2	<Restraint Name>	Card 名	定義する Restraint の Card 名
3	<Restraint Name-child>	Card 名	適用される Restraint の Card 名

説明 :

- ・ Restraint 名: <Restraint Name> の拘束を定義します。
- ・ Case Card、または他の "RestraintSet" Card から参照されます。
- ・ Case Card、または他の "RestraintSet" Card で指定されたとき、拘束 <Restraint Name-child> が適用されます。
- ・ 複数の同じ Card 名の Restraint Card が存在することができます。"RestraintSet" Card と "FixNode" Card, "RestraintEquation" Card などの他の Restraint Card が同じ名前でもかまいません。
- ・ <Restraint Name-child> に、別の "RestraintSet" Card の (または "含む") Restraint Card 名を指定できます。ただし、循環参照 (ある Card 名の "RestraintSet" Card から参照される Restraint Card ("RestraintSet" Card) をたどっていったときに、元の Card 名が参照される) してはいけません。
- ・ 次のようにすると、拘束 restraint_01, restraint_02 , restraint_03 を組み合わせた拘束として、拘束 restraint_101 を定義できます。

\ 拘束 : restraint_101

RestraintSet , restraint_101 , restraint_01 ,
RestraintSet , restraint_101 , restraint_02 ,
RestraintSet , restraint_101 , restraint_03 ,

対応する Card Value Type : なし

FixNode

(Card Type: Restraint Card Type)

* 節点拘束の定義 *

\ フォーマット

FixNode , <Restraint Name> , <Node Name> , <Flag Tx> , <Flag Ty> , <Flag Tz> , <Flag Rx> , <Flag Ry> , <Flag Rz> ,

\ 例

FixNode , restraint_01 , node_01 , 1 , 1 , 1 , 0 , 0 , 0 ,

”フォーマット” の項で複数行に分かれています、実際には一つの Card を複数行に分けることはできません。

Column No.	Column	Column Type	説明
1	“FixNode”		Card Class 名
2	<Restraint Name>	Card 名	定義する Restraint の Card 名
3	<Node Name>	Card 名	拘束する Node の Card 名
4	<Flag Tx>	Bool 値	節点の自由度 Tx を拘束するかどうか
5	<Flag Ty>	Bool 値	節点の自由度 Ty を拘束するかどうか
6	<Flag Tz>	Bool 値	節点の自由度 Tz を拘束するかどうか
7	<Flag Rx>	Bool 値	節点の自由度 Rx を拘束するかどうか
8	<Flag Ry>	Bool 値	節点の自由度 Ry を拘束するかどうか
9	<Flag Rz>	Bool 値	節点の自由度 Rz を拘束するかどうか

説明 :

- ・ Restraint 名: <Restraint Name> の拘束を定義します。
- ・ Case Card、または “RestraintSet” Card から参照されます。
- ・ 節点 <Node Name> の自由度 Tx, Ty, Tz, Rx, Ry, Rz を、<Flag Tx> ~ <Flag Rz> に従って拘束します。なお、拘束する場合は対応する Flag を “YES” または “1” にします。拘束しない場合は対応する Flag を “NO”、“0”、または空白にします。
- ・ ある計算ケースで適用される複数の Restraint Card が、単一の自由度 (節点の Tx, Ty, ... , Rz) を重複して拘束することはできません。ただし “FixNode” 間でのみ重複することができます。つまり、ある計算ケースにおいて、“FixNode” が拘束する自由度は、他の “FixNode” が拘束する自由度と重複すること

ができますが、他の “ForceDisp” や “RestraintEquation” が拘束する自由度と重複することはできません。

対応する Card Value Type : なし

ForceDisp

(Card Type: Restraint Card Type)

強制変位拘束の定義

\ フォーマット

ForceDisp , <Restraint Name> , <Restraint DOF> , <Value> ,

\ 例

ForceDisp , restraint_01 , node_01@TX , 0.55 ,

Column No.	Column	Column Type	説明
1	"ForceDisp"		Card Class 名
2	<Restraint Name>	Card 名	定義する Restraint の Card 名
3	<Restraint DOF>	自由度	拘束する自由度 (例: node_01@TX , node_02@RZ)
4	<Value>	実数	拘束する自由度の強制変位値

説明 :

- ・ Restraint 名: <Restraint Name> の拘束を定義します。
- ・ Case Card、または "RestraintSet" Card から参照されます。
- ・ 節点の自由度 <Restraint DOF> を、<Value> で指定された値に拘束します。
- ・ ある計算ケースにおいて "ForceDisp" が拘束する自由度は、この計算ケースにおいて有効なすべての Restraint Card は拘束する自由度と一致することはできません。(計算ケースで適用される複数の Restraint Card が、単一の自由度 (節点の Tx, Ty, ... , Rz) を重複して拘束することはできません。ただし "FixNode" 間でのみ重複することができます。)

対応する Card Value Type : なし

式から取り除かれます。なお、<Value 2>, <DOF 2> 以降はオプションです。(増減できます。) <Value 1>, <DOF 1> は必須です。

\ 拘束方程式

$$\langle \text{Value} \rangle \times \langle \text{Restraint DOF} \rangle + \langle \text{Value 1} \rangle \times \langle \text{DOF 1} \rangle + \dots + \langle \text{Value N} \rangle \times \langle \text{DOF N} \rangle = 0$$

\ または

$$\langle \text{Restraint DOF} \rangle = - (\langle \text{Value 1} \rangle \times \langle \text{DOF 1} \rangle + \dots + \langle \text{Value N} \rangle \times \langle \text{DOF N} \rangle) \div \langle \text{Value} \rangle$$

- ある計算ケースにおいて“RestraintEquation”が拘束する自由度は、この計算ケースにおいて有効なすべての Restraint Card は拘束する自由度と一致することはできません。(計算ケースで適用される複数の Restraint Card が、単一の自由度(節点の Tx, Ty, ..., Rz)を重複して拘束することはできません。ただし“FixNode”間でのみ重複することができます。)
- 適切な方程式を使用してください。例えば、下の node_01, node_02, node_03 において、node_02 の自由度 Tx を node_01, node_03 の Tx から内分される変位にしようとする場合、“正しい拘束方程式”の代わりに“間違った拘束方程式”のようにしたときは node_01, node_02, node_03 に不正な荷重が付加されたような間違った計算結果になります。一般的に、計算時にエラーなどは生じません。十分注意してください。(参考:“RestraintEquation”を使用している節点(独立自由度, 従属自由度)や、要素全体が剛体変位をしないように“FixNode”Cardなどで拘束している節点(に接続している Element)の荷重(拘束の反力)の合力などを詳しく調べると分かることもあります。)

\ 例: 正しい拘束方程式と間違った拘束方程式

\ Node (同一直線上にあり、y 座標のみ異なる 3つの節点。)

```
Node, node_01, , 0.0, 100.0, 0.0,
Node, node_02, , 0.0, 106.0, 0.0,
Node, node_03, , 0.0, 110.0, 0.0,
```

\ 正しい拘束方程式 (node_02 の変位 Tx を node_01 と node_03 の Tx を内分する変位とする。)

\ node_02@TX = 0.4 × node_01@TX + 0.6 × node_03@TX

```
RestraintEquation, restraint_01, -10.0, node_02@TX, +4.0, node_01@TX,
+6.0, node_03@TX,
```

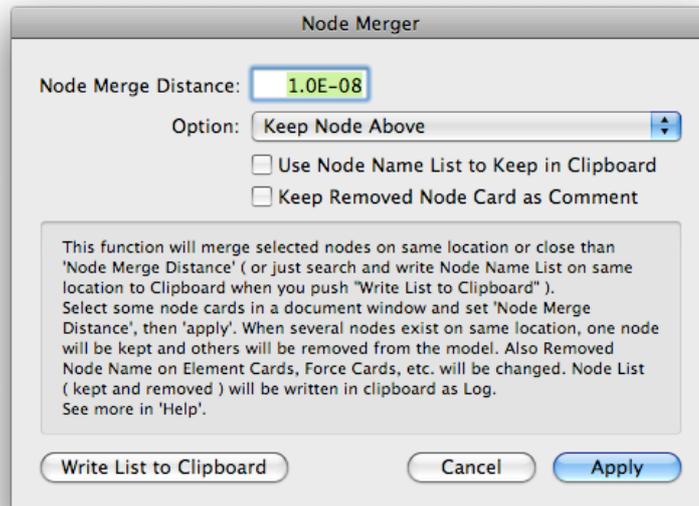
\ 間違った拘束方程式 (“+4.0” と “+6.0” が逆)

```
RestraintEquation, restraint_01, -10.0, node_02@TX, +6.0, node_01@TX,
+4.0, node_03@TX,
```

対応する Card Value Type : なし

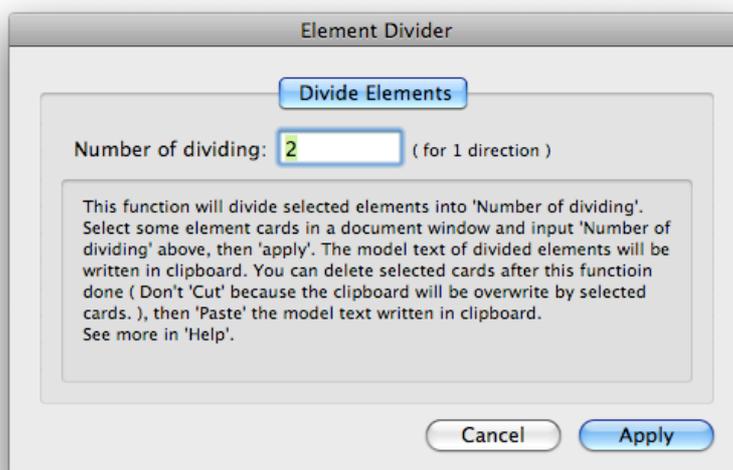
その他の機能

1. Tool - Merge Nodes on Same Location について



- ・ 選択した Node の中で、同じ位置（または距離が指定された値以下）の複数の節点（Node）を統合します。（同じ位置の Node が複数ある場合、一つを残し、他は消去します。）なお、消去される Node を参照していた Element Card, Force Card, Restraint Card などの消去される Node を参照している部分は、統合で残される Node に変更されます。
- ・ “Node Merge Distance” テキストフィールドに、同じ位置と判断する Node 間の距離を入力します。
- ・ “Option” ポップアップメニューの “Keep Node Above” / “Keep Node Below” で、同じ位置の Node が見つかったときに、先にある Node を残すか、後にある Node を残すかを選択できます。また、“Use Node Name List to Keep in Clipboard” をチェックすると、クリップボード内の Card を残すようにします。あらかじめ Card 名のリスト（Card 名を改行文字またはカンマで区切ったテキスト）をクリップボードにコピーしておきます。
- ・ “Write List to Clipboard” ボタンを押すと、同じ位置の Node のリストをクリップボードに書き出します。同じ行の Node Card 名が、同じ位置にある Node のリストで、左にある Node が（同じ設定で Apply を押したときに）残されます。“Apply” ボタンを押したときに、統合される Node を確認できます。
- ・ “Apply” ボタンを押すと、統合を実行し、統合した Node のリスト（“Write List to Clipboard” ボタンを押したときに書き出されるものと同じです）をクリップボードに書き出します。
- ・ モデルの作成において、複雑な形状を、複数の領域に分け、それらの領域の境界の Node の位置を同じにしておき、それをこの機能で接続することができます。

2. Tool - Divide Elements について



この機能は、本アプリケーションの検証、動作確認などのために作成されました。この機能を使用して要素を細かくすることを前提とせず、初めから適正な分割でモデルを作成することを推奨します。

- ・ 選択した Element を分割 (例えば Quad 要素は 2x2, 3x3 など、Hexa 要素は 2x2x2, 3x3x3 など、Rod 要素は分割しません。) します。分割した結果のモデルデータはクリップボードに書き出されます。実際のモデルは変更されません。
- ・ 選択した Card の中に Quad 要素や Hexa 要素があれば、その要素を構成する Node の Node Card 名、及び Node の位置より、要素を分割するときに追加する Node を作成し、それをクリップボードに書き出します。(追加する Node の位置は、要素を構成していた Node を(直線で)内挿した位置になります。また、追加する Node の Card 名は、分割する要素を構成していた Node の名前を連ねた形の名前(一般的に非常に長い名前になります)になります。)(要素を構成する Node は選択範囲に含まれている必要はありません。)また、Quad 要素や Hexa 要素は分割され、分割した Element Card がクリップボードに書き出されます。(分割された(後の) Element Card の名前は、分割前の Element Card の名前の後に番号を追加した形の名前になります。)なお、Quad 要素や Hexa 要素以外の Card は、そのままクリップボードに書き出されます。ただし、変更されずに書き出された Card / 要素の分割で追加された Node Card / 分割された Element Card は、それぞれまとめて書き出されます。
- ・ 荷重 (Force Card) や拘束 (Restraint Card) は、変更 (追加など) されません。分割前の荷重点、拘束点のみが荷重、拘束され、要素の分割で追加された Node に荷重や拘束などは追加されません。追加された Node に合わせて、荷重や拘束の追加または再作成が必要になることがあります。
- ・ 選択していた Card を消去し、クリップボードに書き出されたモデルデータを追加すると、分割したモデルになります。(なお、選択している Card を消去するときは、メインメニューの "Edit" - "Delete" を使用して下さい。"Edit" - "Cut" を使用すると、書き出されたクリップボード内のモデルデータが失われます。)テキストエディタソフトウェアなどにペーストし、確認、変更などを行うこともできます。

- ・ 選択していた Card を消去し、書き出されたモデルデータをそのままの順番で追加した場合は、一般的に、剛性マトリクスバンド幅は非常に大きくなります。
- ・ 分割して変更されたモデルを、(“Undo” 以外の方法で) Element を統合して、元の分割前のモデルにすることはできません。(“Undo” は、ファイルを閉じる後は使用できません。) 分割前のモデルデータをあらかじめ保存しておくことを推奨します。

その他

1. 剛性マトリクスバンド幅、他について

注: 以下の情報は、Load Design ver.0.8.1.5 についての情報です。将来の Load Design で変わることがあります。

- ・ Load Design ver.0.8.1.5 において、開発途中ですが Node のオーダリング (全体の剛性マトリクスでの Node (自由度) 順序の並び替え) の機能が実装されました。これは剛性マトリクスバンド幅に影響し、バンド幅が小さいほど、使用メモリは少なく計算時間は短くなります。(この機能の開発を継続する予定です。)
- ・ 剛性方程式において同一の拘束条件の計算ケース (同一の Restraint Card 名を指定している Case Card) はまとめて計算されますが、同一の拘束条件の計算ケースがモデルデータ内で連続している必要はありません。間に異なる拘束条件の Case がある場合も、同一の拘束条件の Case はまとめて計算されます。
- ・ 剛性方程式の計算は IEEE 754 - 64bit (倍精度浮動小数点数) で行われます。

サポートについて

1. サポートについて

Web ページ <http://www.load-design.com/> を参照してください。

改訂履歴

- ・ 2009/2/17 : Load Design version 0.8 ユーザーズマニュアル 作成
- ・ 2009/5/20 : 一部改訂 (誤記修正・その他) (ver 0.8.0.3)
- ・ 2009/7/7 : 一部改訂 (拘束と荷重の表示方法変更による注釈の追加) (ver 0.8.1)
- ・ 2011/8/10 : 一部改訂 (サンプルモデルデータの間違いの修正) (ver 0.8.1.4.2)
- ・ 2011/10/5 : Node オーダリングと計算ケースの順序に関する変更 (主に page 74) (ver 0.8.1.5)