

Babieca OVMモデラー Ver.2.0

操作説明書（OVM編）

2023/01/25

BABIECA, LLC / CAD SOLUTIONS Inc.

目次

0. OVMモデラーによる「組み合わせ問題」の解決

1. OVMモデル

- OVMの記法

2. OVMモデラーの起動

- ユーザーを選択する、ユーザーを登録する

3. プロジェクト、フォルダの作成

- 「プロジェクト」を開く、「プロジェクト」を作成する
- 「パッケージ」を作成する
- 「プロジェクト」を保存する

4-1. OVMモデリング（その1）

- 「ダイアグラム」を作成する、「ダイアグラム」を開く
- 「可変点」を作成する
- 「選択枝」を作成する
- 「可変点～選択枝」間の関係を定義する
- 「可変点～選択枝」間の関係を変更する
- 「選択枝」の最大選択数を定義する
- 「要求関係」「排他関係」を定義する
- 「関係」線のレイアウトを変更する
- 解の数を表示
- モデル要素へのアノテーション記述
- 表示フォントの変更

4-2. OVMモデリング（その2）

4-3. OVMモデリング（その3）

5. モデルとビュー

- 「モデル」と「ビュー」
- モデル要素を削除する

6. 属性の作成

- 「属性」とは
- 「属性項目」を定義する
- 「属性値」を設定する
- 「属性」を表示する

7. AIソルバーの実行

- AIソルバーが評価対象とする「式」を設定
- AIソルバーで解く「制約」と「目的関数」を設定
- AIソルバーの実行例

（参考）OVMモデリングの定石

0. OVMモデラーによる「組み合わせ問題」の解決

「OVMモデラー」とは

OVM : Orthogonal Variability Model (ISO 26550)

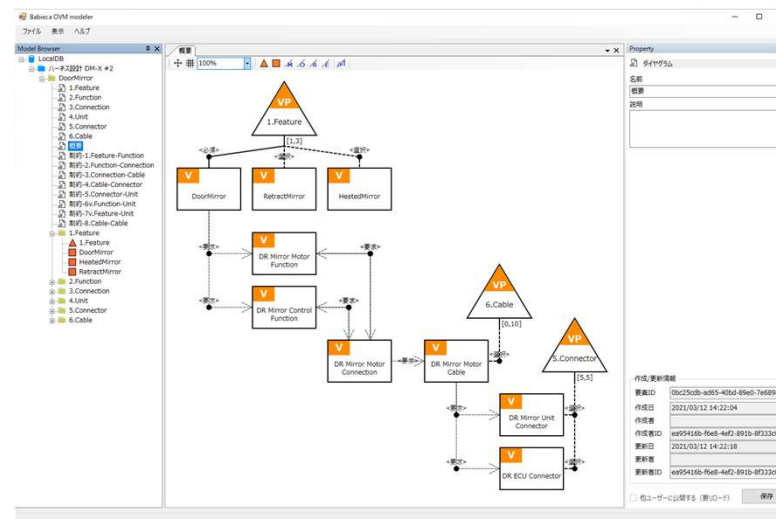
- “**組み合わせ**”と“**制約**”に特化したモデリング言語
- PLM/ALMの世界で提唱されたものであり、**製造業との親和性**が高い
- UMLやSysML等と比べ、仕様が非常に単純で**習得が容易**

OVMモデラーによって、プラットフォームや部品間の「制約」や「ルール」を可視化し、AIソルバーによって、「実現可能な組み合わせ」「最適な組み合わせ」を解く

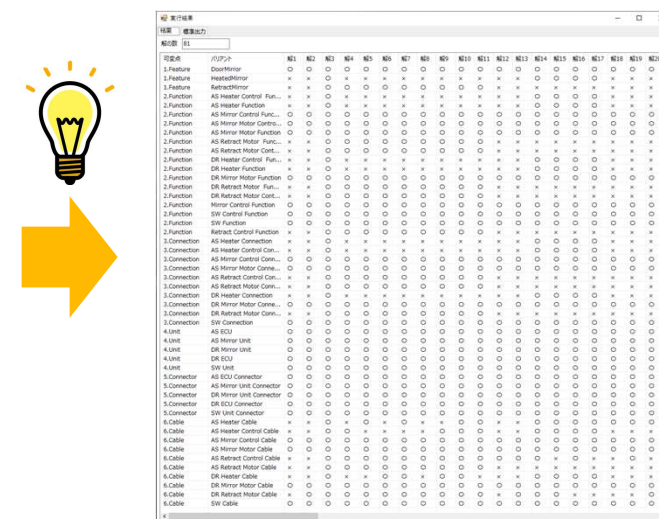
プラットフォーム、パーツ



OVMモデラー



AIソルバー



いろいろな選択肢

制約、ルール、暗黙知

可視化、データ化して集約

制約を満たす組み合わせ

最適な組み合わせ

「組み合わせ問題」の例：子供が遠足に持っていくお菓子を選ぶとしたら？

お菓子は5種類（キャンディ、クッキー、うまい棒、ポテトチップス、チョコレート）

飲み物は3種類（お茶、ジュース、コーラ）

但し、

- ・金額は300円以内
- ・必ずクッキーは持参すること
- ・チョコを選んだら飲み物はお茶のみ（甘い飲み物はダメ）
- ・ポテチとコーラの組み合わせはダメ



どんな組み合わせが考えられるか？：制約充足問題

最適な組み合わせは何か？：制約最適化問題



「組み合わせ問題」をOVMによって定型化/可視化する

お菓子は5種類（キャンディ、クッキー、うまい棒、ポテトチップス、チョコレート）

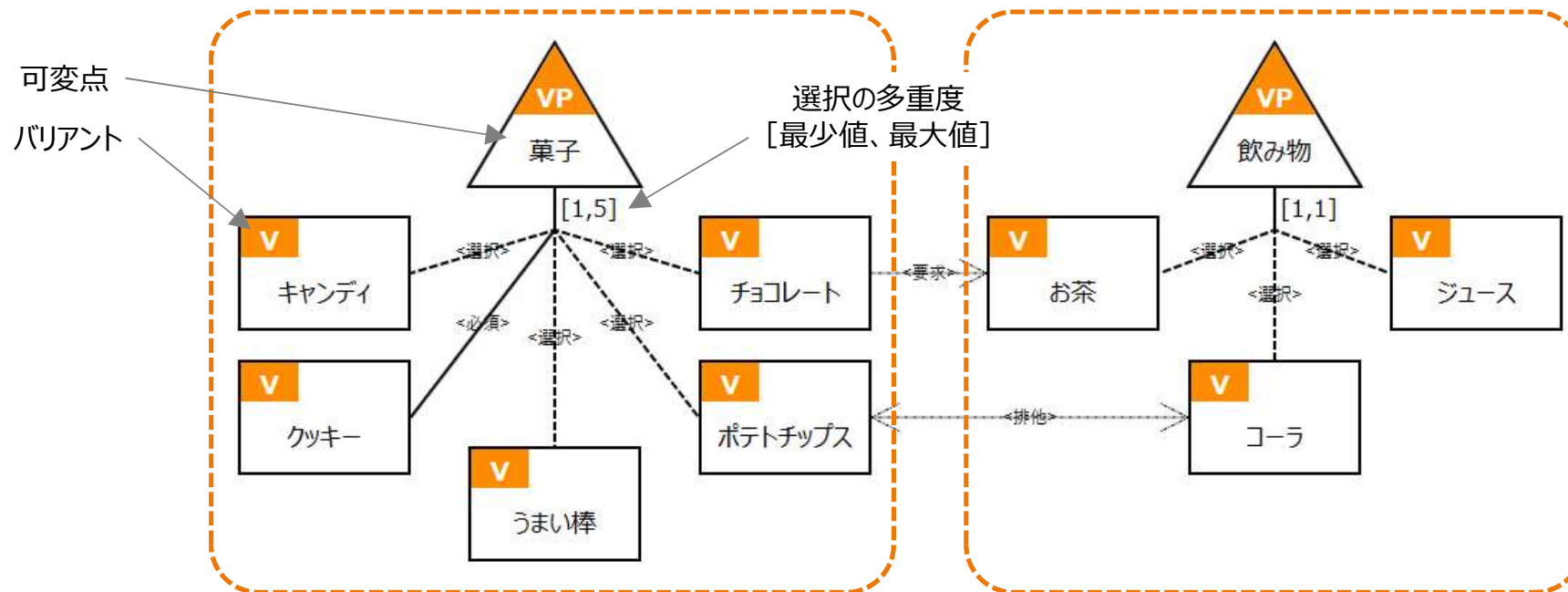
飲み物は3種類（お茶、ジュース、コーラ）

但し、金額は300円以内

必ずクッキーは持参すること

チョコを選んだら飲み物はお茶のみ（甘い飲み物はダメ）

ポテチとコーラの組み合わせはダメ



「組み合わせ問題」をOVMによって定型化/可視化する

お菓子は5種類（キャンディ、クッキー、うまい棒、ポテトチップス、チョコレート）

飲み物は3種類（お茶、ジュース、コーラ）

但し、金額は300円以内

必ずクッキーは持参すること

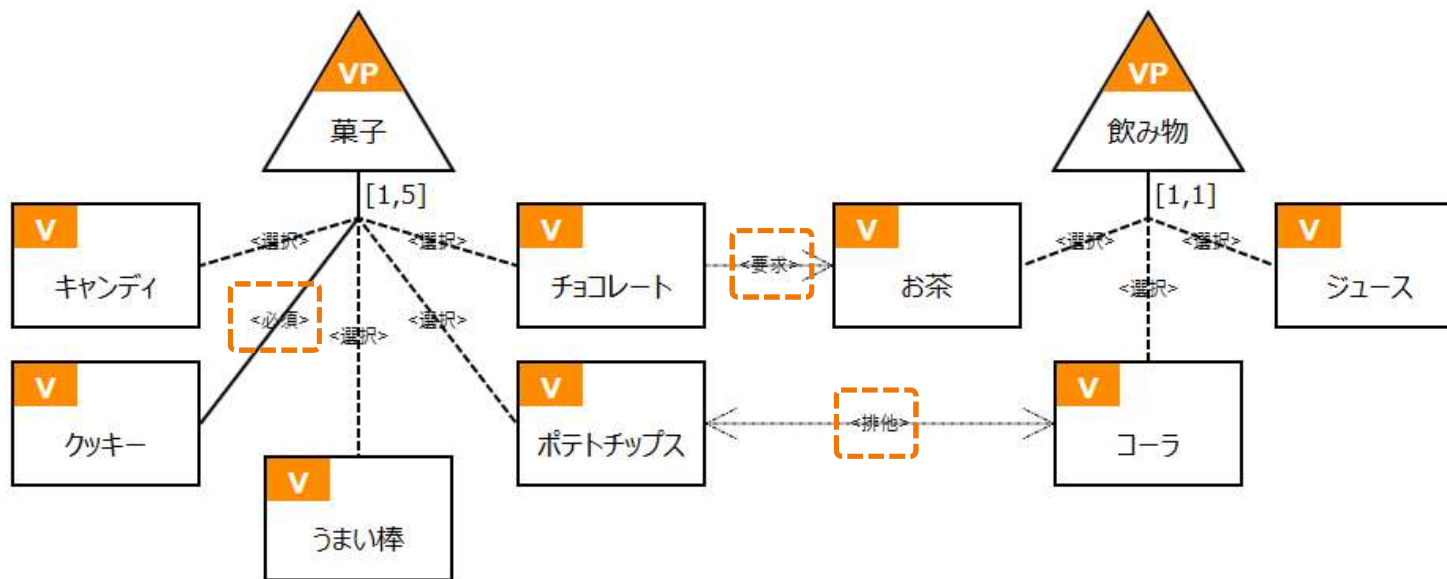
必須

チョコを選んだら飲み物はお茶のみ

要求

ポテチとコーラの組み合わせはダメ

排他



「組み合わせ問題」をAIによって解く：ふたつのアプローチ



量子
コンピューター

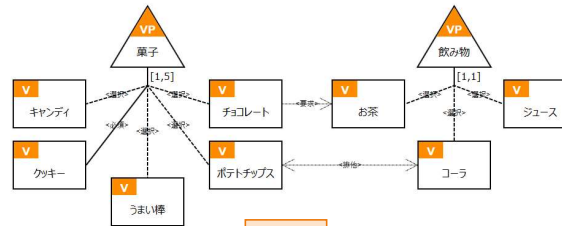
経験を積む(学習する)と、
直感で答えが出せるよう
になる

満足するまで何度も
試行錯誤する



機械学習ソルバー
(ディープラーニング)

OVMモデルによって定型化/可視化
した問題をソルバーに投入し、
答えを得る



AIソルバー

あまりに複雑な問題だと
時間がかかってあきらめて
しまう

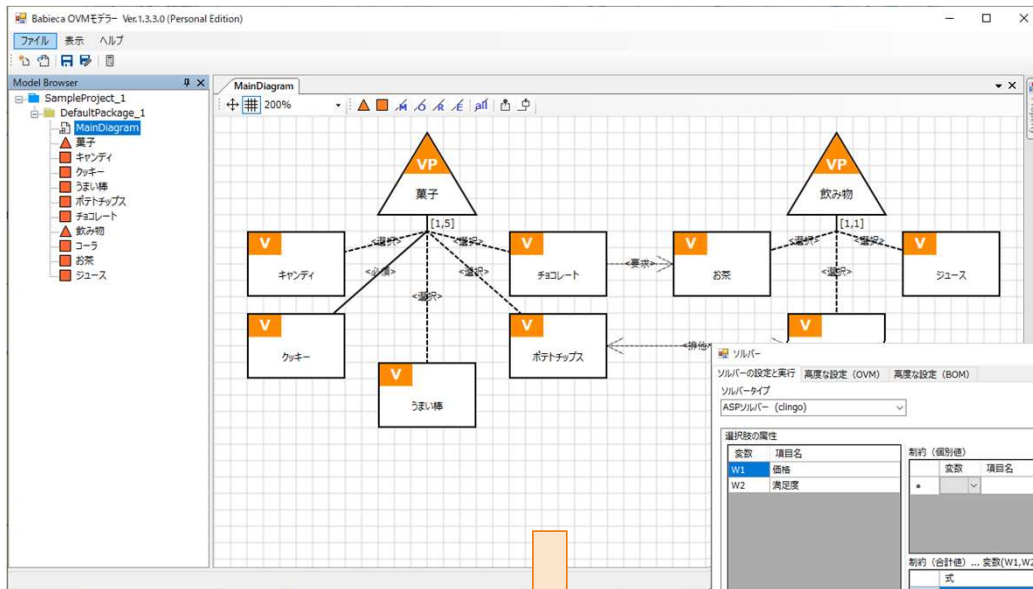
正しい答えを得るた
めのロジックを考えて
計算する



数学ソルバー
(SAT/ASP等)

本バージョンでは右側、ASPソルバーを実装しています

AIソルバーの実行例



The Solver configuration window shows the selection of the ASP solver (clingo) and the definition of variables W1 (Price) and W2 (Satisfaction). The window includes sections for '変数' (Variables), '制約' (Constraints), and '目的関数' (Objective Function).

価格の合計が300円以下となる組み合わせは？

実行結果

結果 標準出力

解の数 25

可...	選択肢	制約/目的...	解1	解2	解3	解4	解5	解6	解7	解8	解9	解10	解11	解12
飲み物	お茶		○	○	○	×	×	○	×	×	×	×	×	○
飲み物	コーラ		×	×	×	×	○	×	×	×	○	○	×	×
飲み物	ジュース		×	×	×	○	×	×	○	○	×	×	×	×
お菓子	クッキー		○	○	○	○	○	○	○	○	○	○	○	○
お菓子	うまい棒		×	×	○	×	×	○	×	×	○	○	○	×
お菓子	キャンディ		×	○	×	×	×	○	○	○	○	○	○	×
お菓子	チョコレート		×	×	×	×	×	×	×	×	×	×	×	○
お菓子	ポテトチップス		×	×	×	×	×	×	×	×	×	×	×	×
	W1		80	100	110	110	130	130	130	140	150	160	160	180

価格の合計が300円以下で、満足度が最大となる組み合わせは？

実行結果

結果 標準出力

解の数 1

可...	選択肢	制約/目的...	解1
飲み物	お茶		×
飲み物	コーラ		×
飲み物	ジュース		○
お菓子	クッキー		○
お菓子	うまい棒		○
お菓子	キャンディ		○
お菓子	チョコレート		×
お菓子	ポテトチップス		○
	W1		280
	W2		27

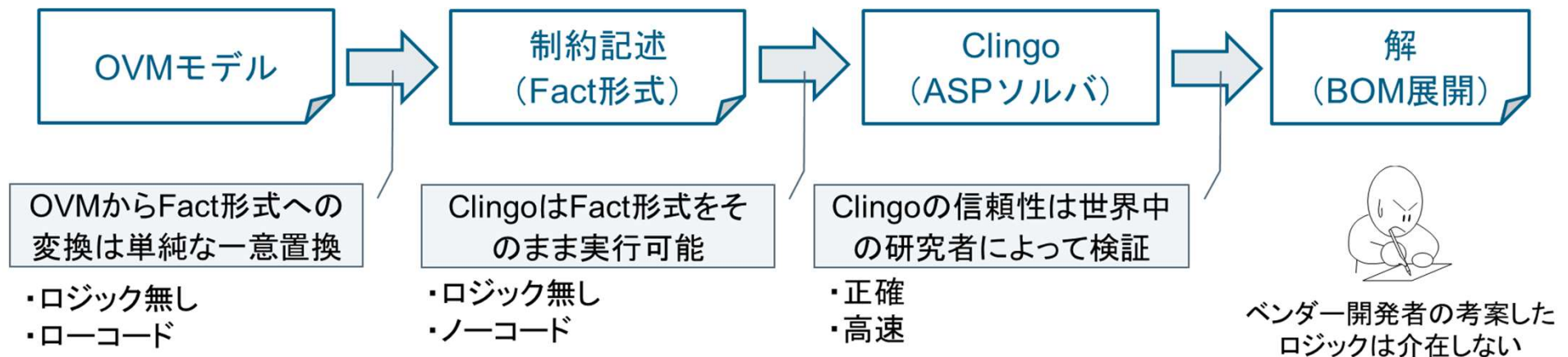
備考：Why OVM, Why ASP

BOMシステムは、企業活動の根幹となる重要なシステムであり、高度な信頼性が要求され、万が一にも「間違い」は許されない。

OVMモデラーでは、

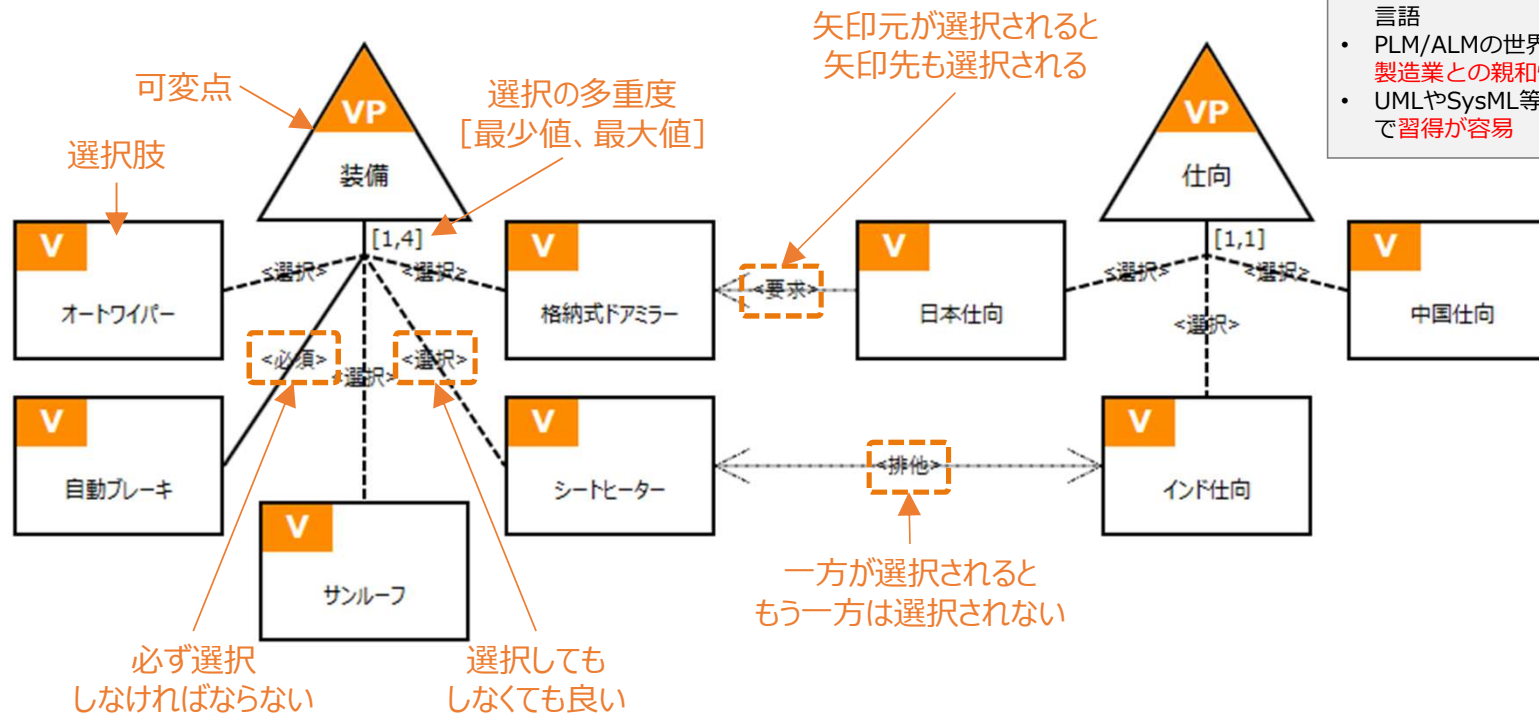
- ・ Clingo : 世界中の研究者/開発者によって長年にわたって検証されているオープンソースASPソルバシステム
現在も開発/検証が継続されており、信頼性と性能が向上し続けている
- ・ OVM : ASPと非常に親和性の高いモデリング言語（制約の考え方が同じ/後述）
を用いることで高い信頼性を確保している

Clingo5
・ 2016/09/30 ... v5.0.0
...
・ 2021/04/21 ... v5.5.0
・ 2021/11/16 ... v5.5.1
※<https://github.com/potassco/clingo/tags>



1. OVMモデル

OVMの記法



OVM : Orthogonal Variability Model
(ISO 26550)

- “組み合わせ”と“制約”に特化したモデリング言語
- PLM/ALMの世界で提唱されたものであり、**製造業との親和性が高い**
- UMLやSysML等に比べ、仕様が非常に単純で**習得が容易**

「可変点」には、“必須”と“必須ではない”ものがある

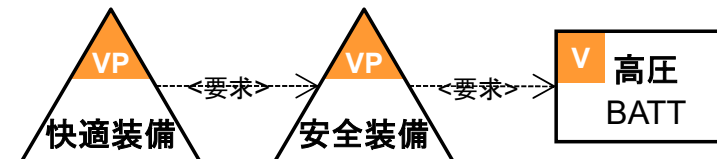
必須：
どれかが必ず選択される



必須ではない：
どれも選択されないことがある



<要求>と<排他>は、「可変点」同士、「選択肢」同士、「可変点」と「選択肢」間、のいずれでも引くことができる

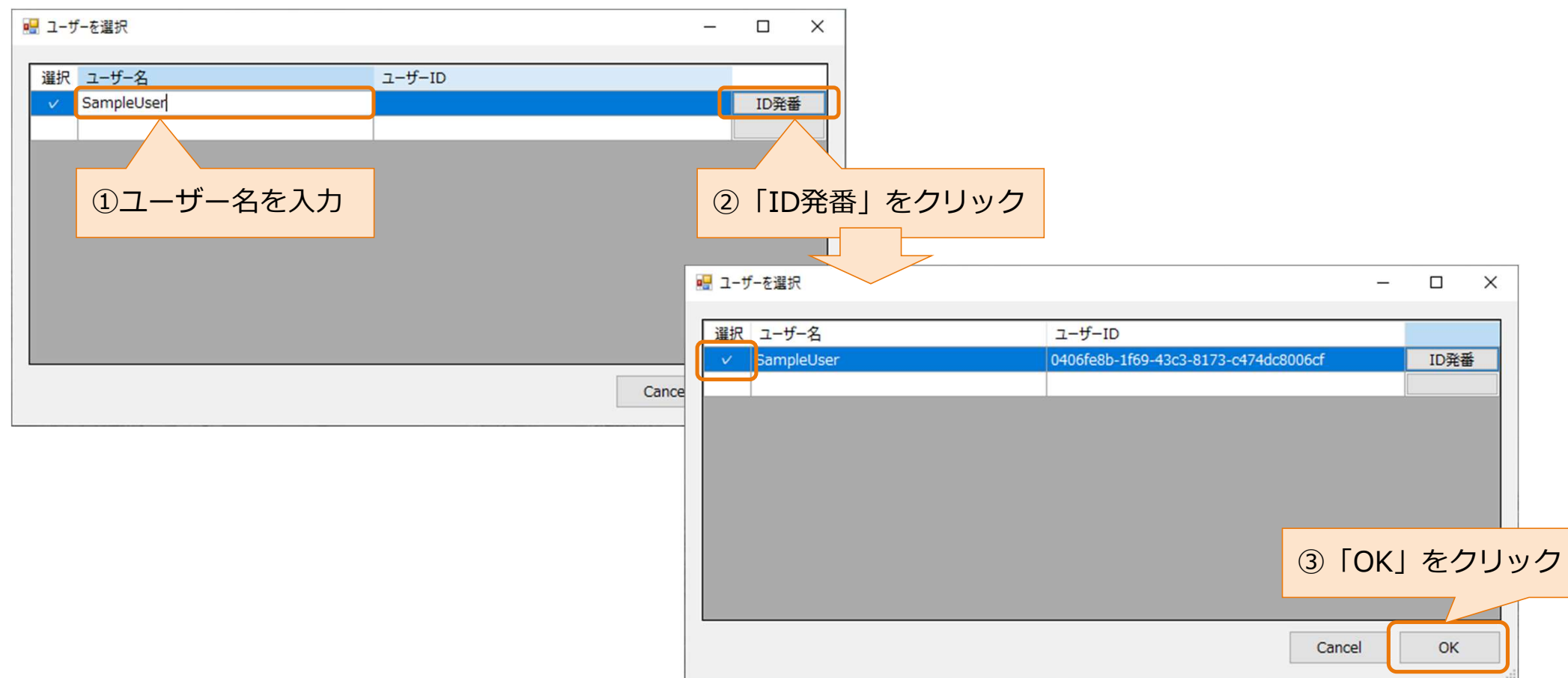


2. OVMモデラーの起動

ユーザーを選択する、ユーザーを登録する

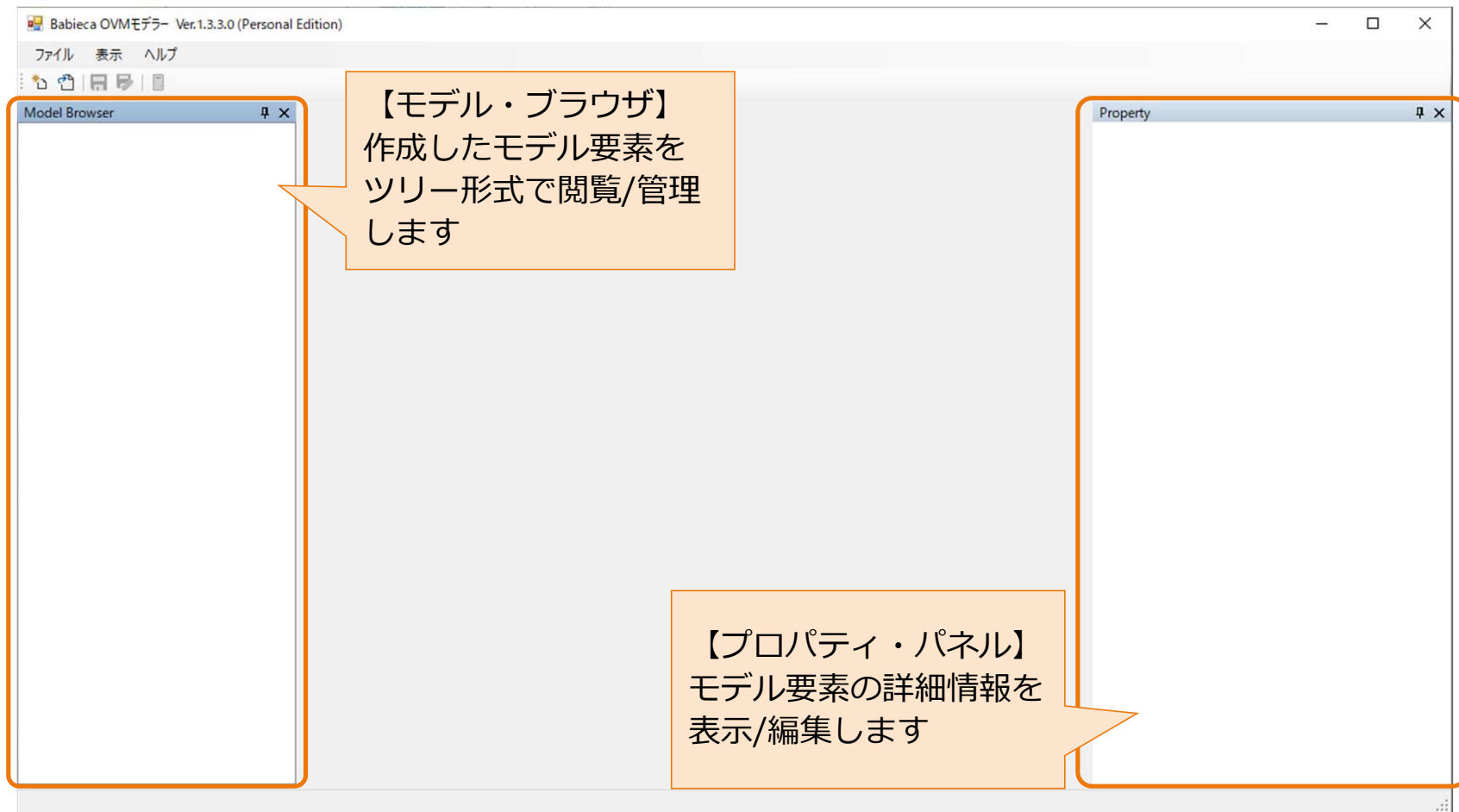
ユーザーを選択、または新たにユーザーを登録します

(新たにユーザーを登録する場合)



OVMモデラーの初期画面

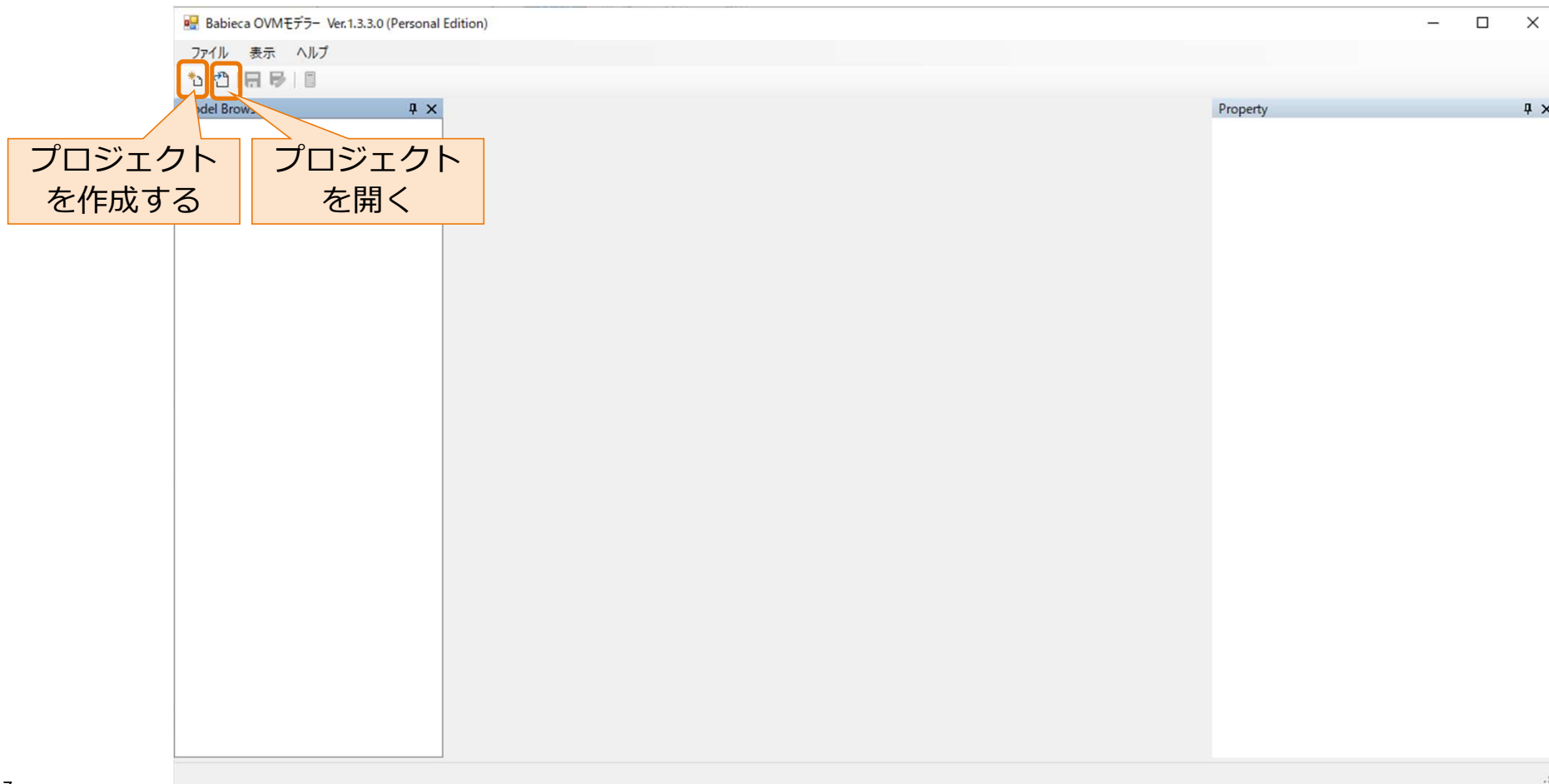
ユーザーを選択し「OK」をクリックすると、OVMモデラーが起動します



3. プロジェクト、フォルダの作成

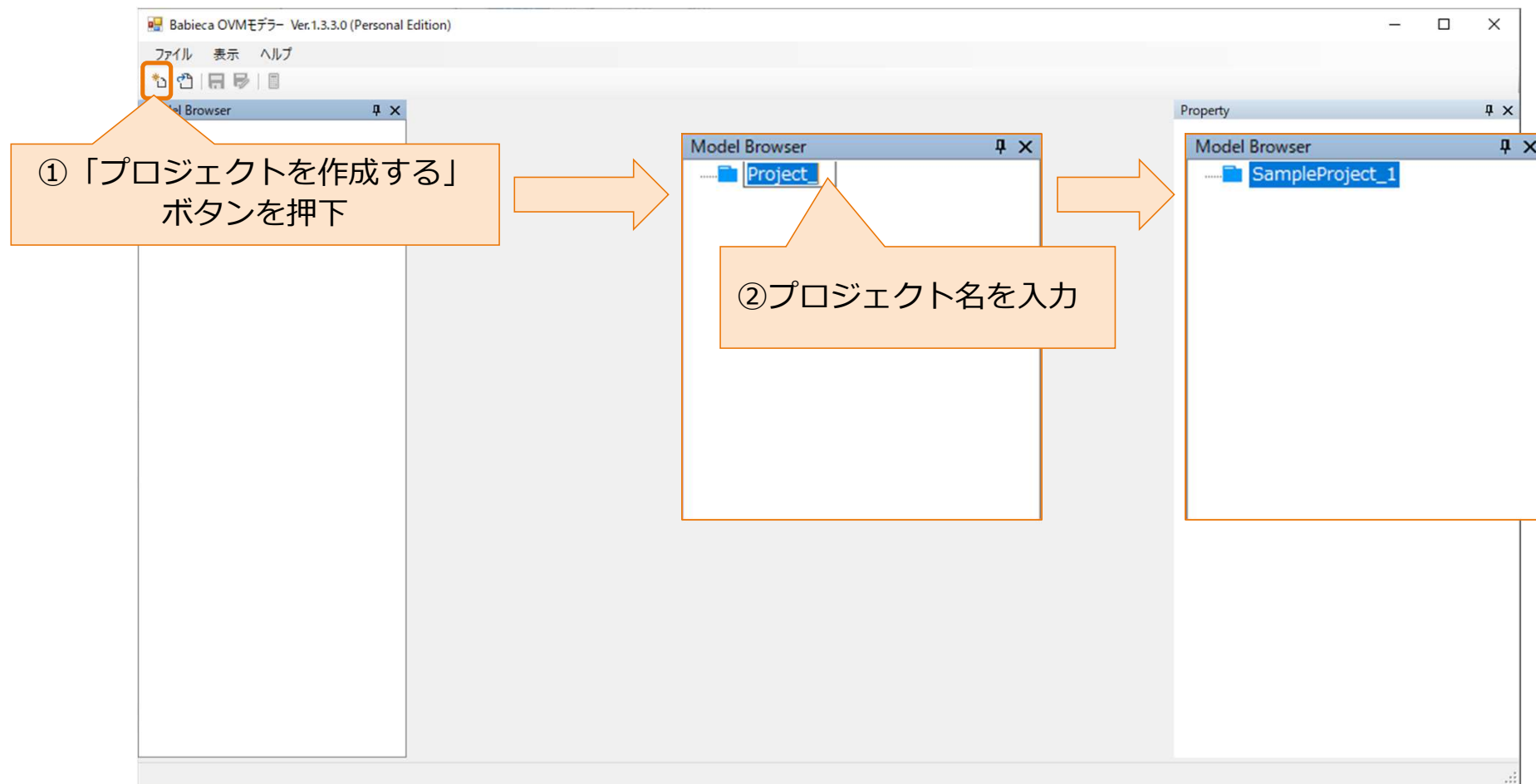
プロジェクトを開く、プロジェクトを作成する

「プロジェクトを開く」ボタン、または「プロジェクトを作成する」ボタンを押下します



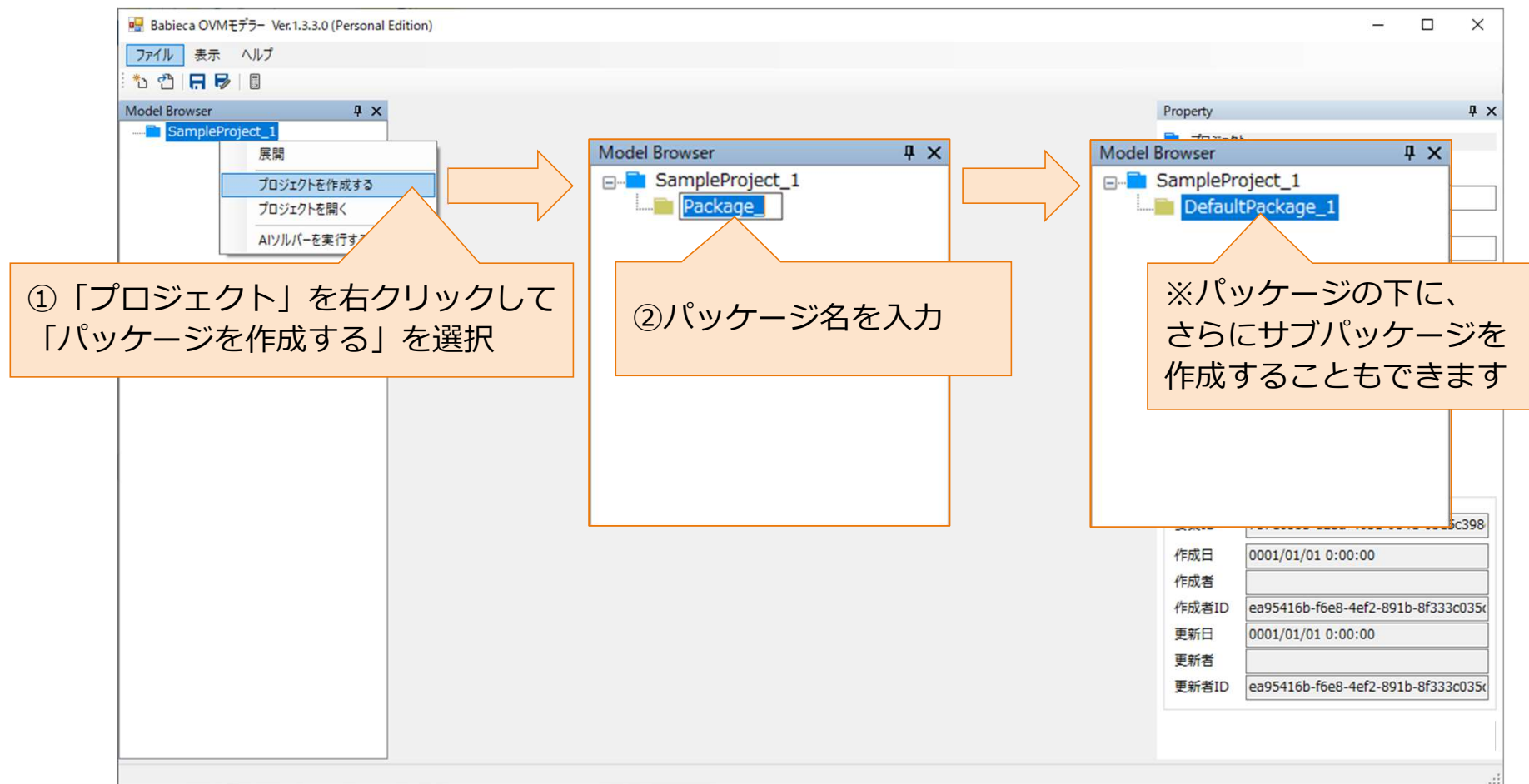
プロジェクトを作成する

「プロジェクトを作成する」ボタンを押下します



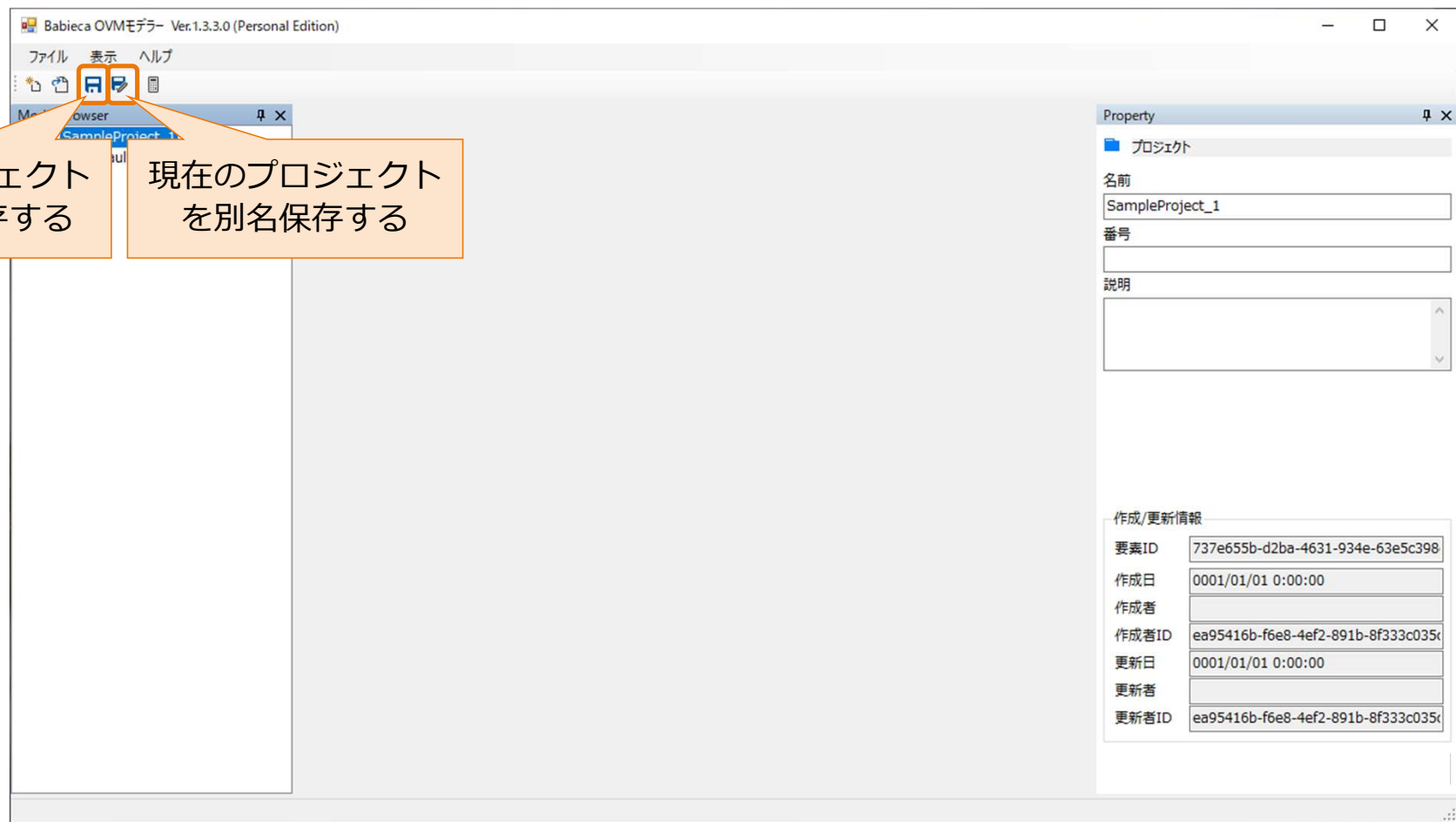
パッケージを作成する

「プロジェクト」を右クリックして「パッケージを作成する」を選択します



プロジェクトを保存する

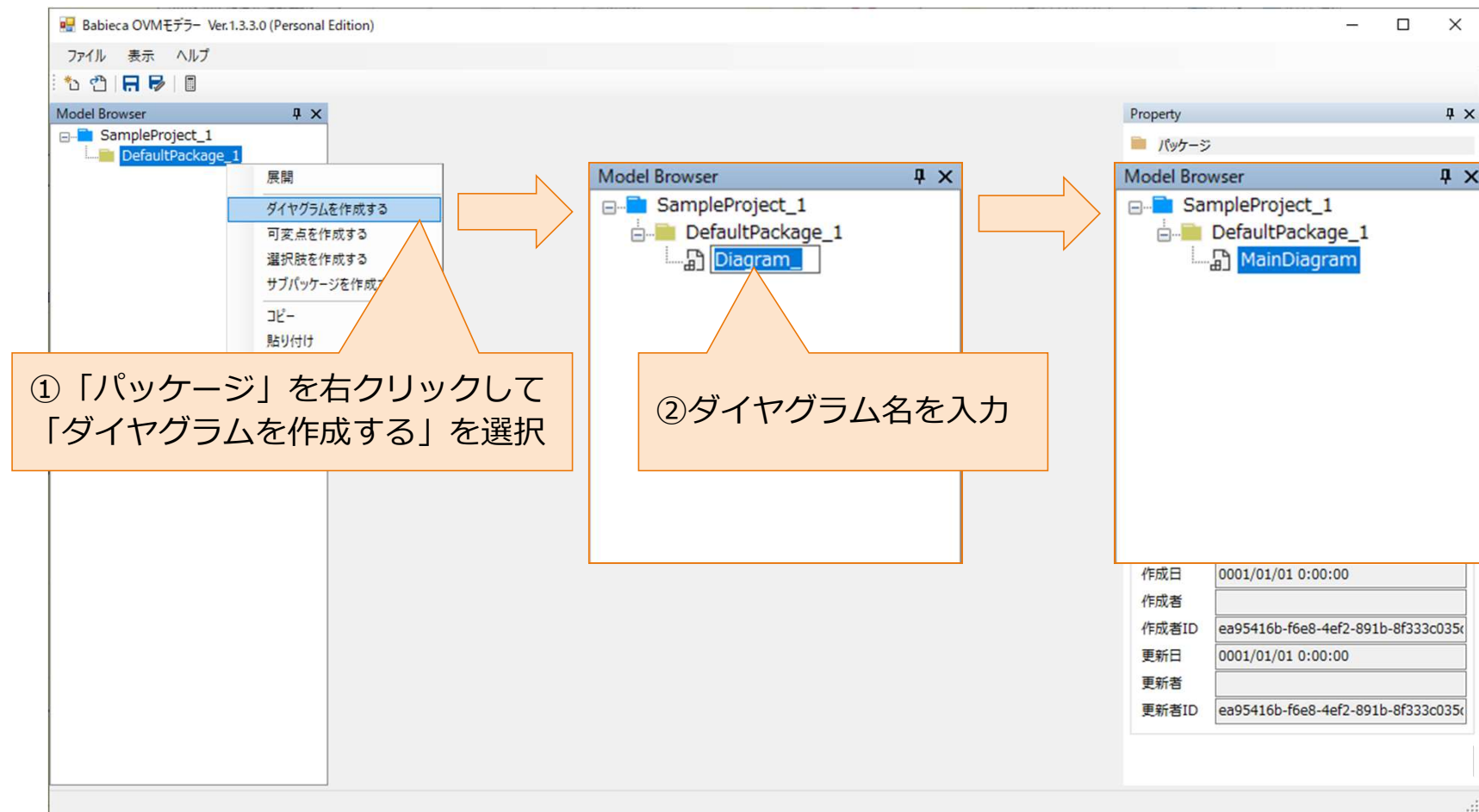
「上書き保存する」ボタン、または「名前を付けて保存する」ボタンを押下します。



4-1. OVMモデリング（その1）

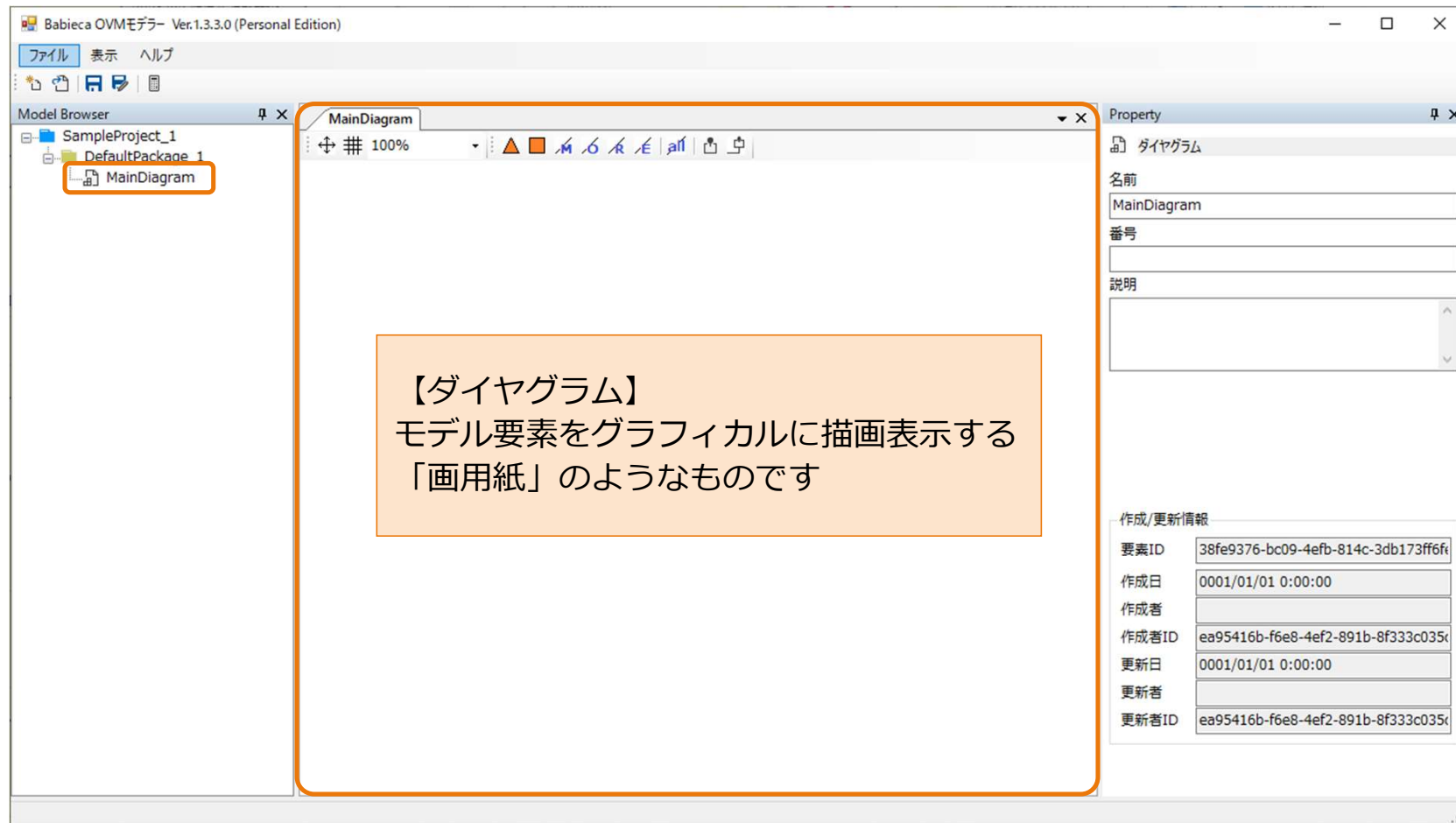
ダイアグラムを作成する

「パッケージ」を右クリックして「ダイアグラムを作成する」を選択します



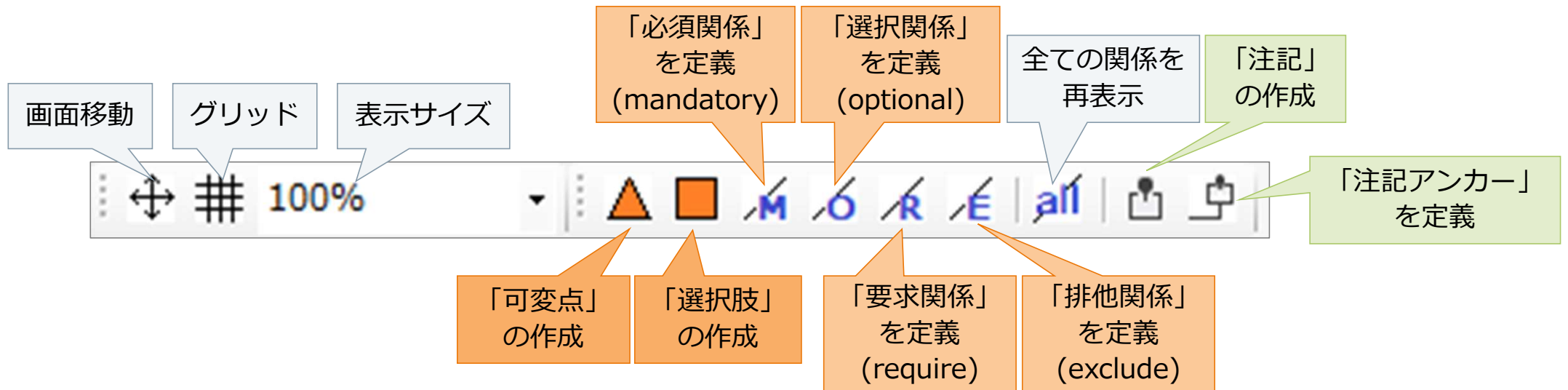
ダイアグラムを開く

作成した「ダイアグラム」をダブルクリックすると、「ダイアグラム」が開きます



ダイアグラムの操作

「ダイアグラム」のツールバーを用いてモデル要素を作成します



「可変点」を作成する

「可変点の作成」アイコンボタンを押下した後、ダイアグラム上の任意の場所をクリックすると、可変点が作成/配置されます

The screenshot illustrates the process of creating a variable point (VP) in the Babieca OVM Modeller. The interface includes a Model Browser on the left, a MainDiagram workspace in the center, and a properties panel on the right.

① 「可変点の作成」 ボタンを押下

An arrow points from the 'MainDiagram' toolbar to the 'VP' icon, which is a triangle with the letters 'VP' inside.

② ダイアグラム上の任意の場所をクリック

An arrow points from the 'VP' icon to a small triangle on the grid, labeled 'VP_ [1,1]'.

③ 可変点の名前を入力

An arrow points from the 'VP_ [1,1]' label to a larger triangle on the grid, labeled 'VP 菓子 [1,1]'.

The properties panel on the right shows the following information:

- ☒ 必須にする
- 選択肢の最小選択数: 1
- 選択肢の最大選択数: 1
- 作成/更新情報
- 要素ID: c8ee74e8-54ac-4101-84a0-b9bb74a8
- 作成日: 0001/01/01 0:00:00
- 作成者:
- 作成者ID: ea95416b-f6e8-4ef2-891b-8f333c035c
- 更新日: 0001/01/01 0:00:00
- 更新者:
- 更新者ID: 00000000-0000-0000-0000-00000000

「選択肢」を作成する

「選択肢の作成」アイコンボタンを押下した後、ダイアグラム上の任意の場所をクリックすると、選択肢が作成/配置されます

①「選択肢の作成」ボタンを押下

②ダイアグラム上の任意の場所をクリック

③選択肢の名前を入力

MainDiagram

菓子

菓子

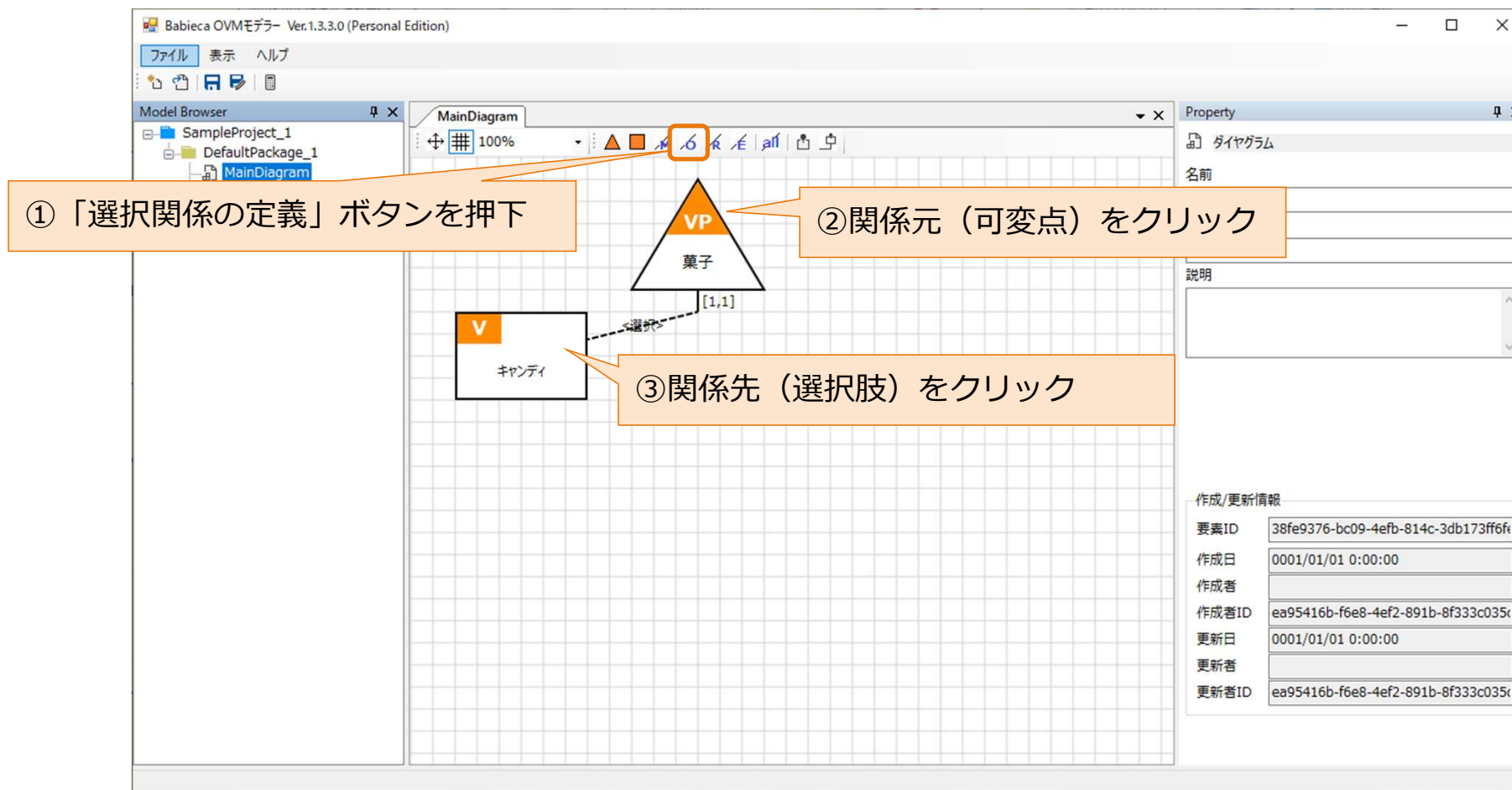
キャンディ

作成/更新情報

要素ID	0df65264-458d-48fc-9ed4-6af4371e4t
作成日	0001/01/01 0:00:00
作成者	
作成者ID	ea95416b-f6e8-4ef2-891b-8f333c035c
更新日	0001/01/01 0:00:00
更新者	
更新者ID	00000000-0000-0000-0000-00000000

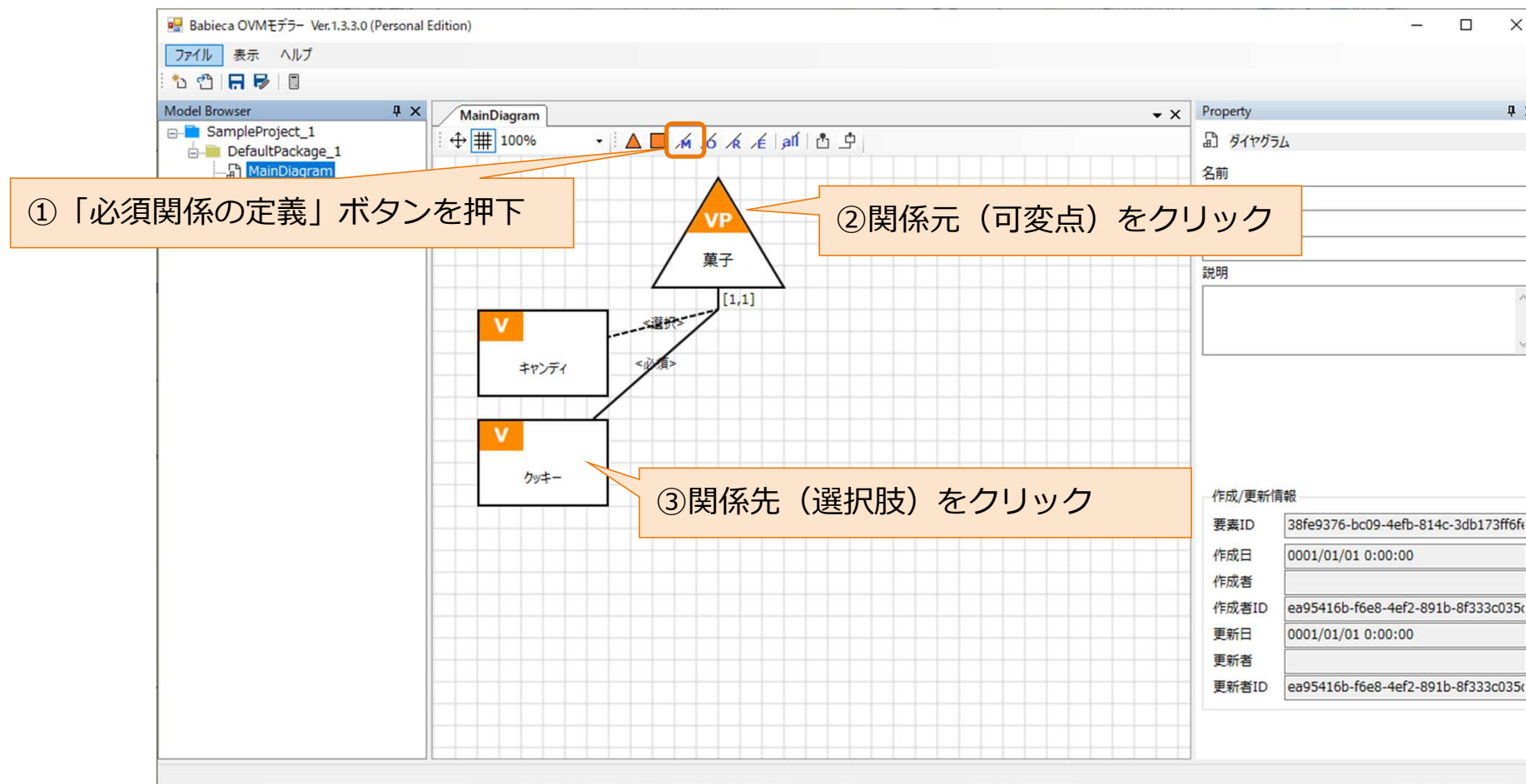
「可変点～選択肢」間の関係を定義する（“選択”関係）

「選択関係の定義」ボタンを押下し、関係元、関係先の順にクリックすると、「選択関係」が定義/表示されます



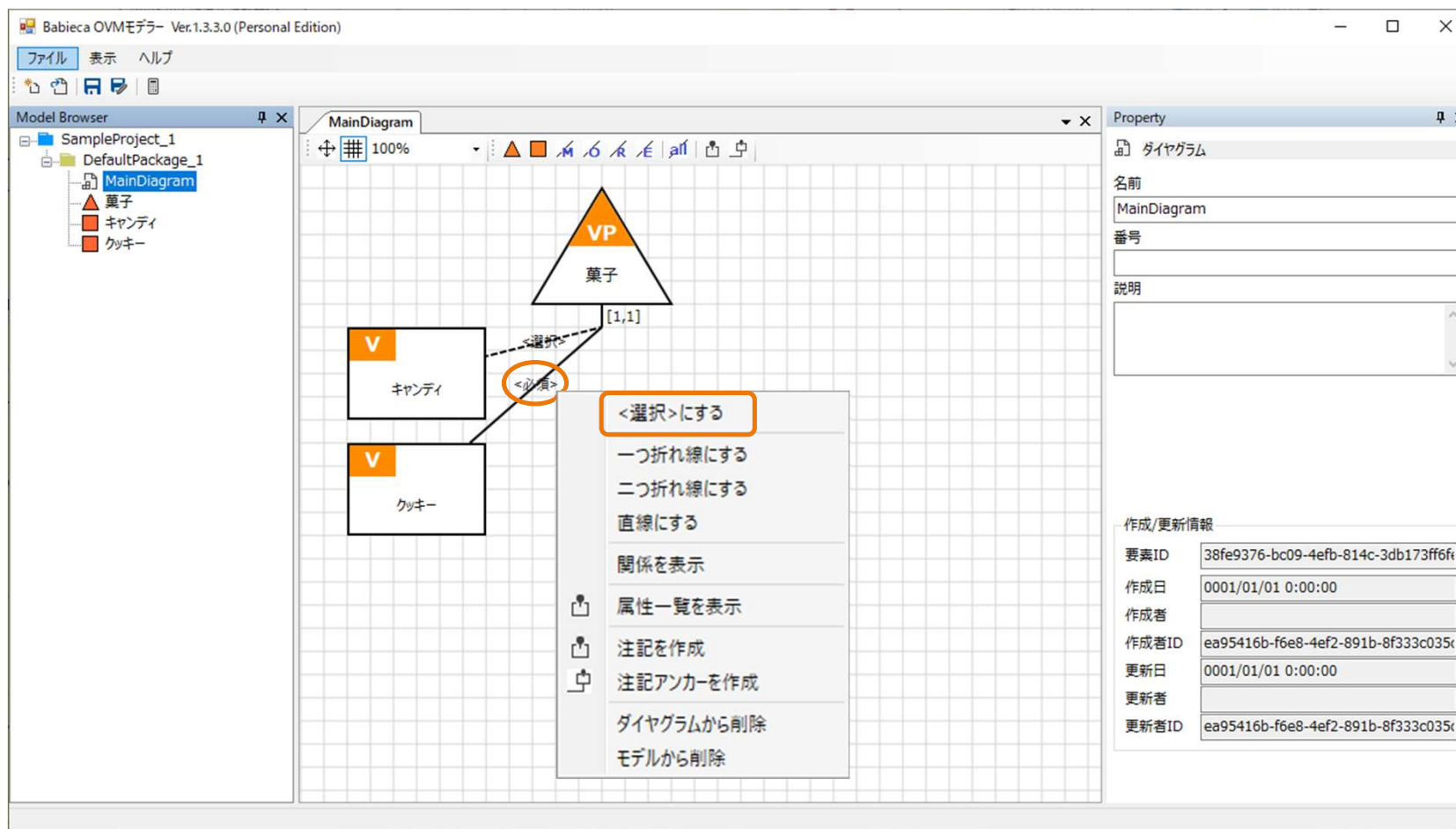
「可変点～選択肢」間の関係を定義する（“必須”関係）

「必須関係の定義」ボタンを押下し、関係元、関係先の順にクリックすると、「必須関係」が定義/表示されます

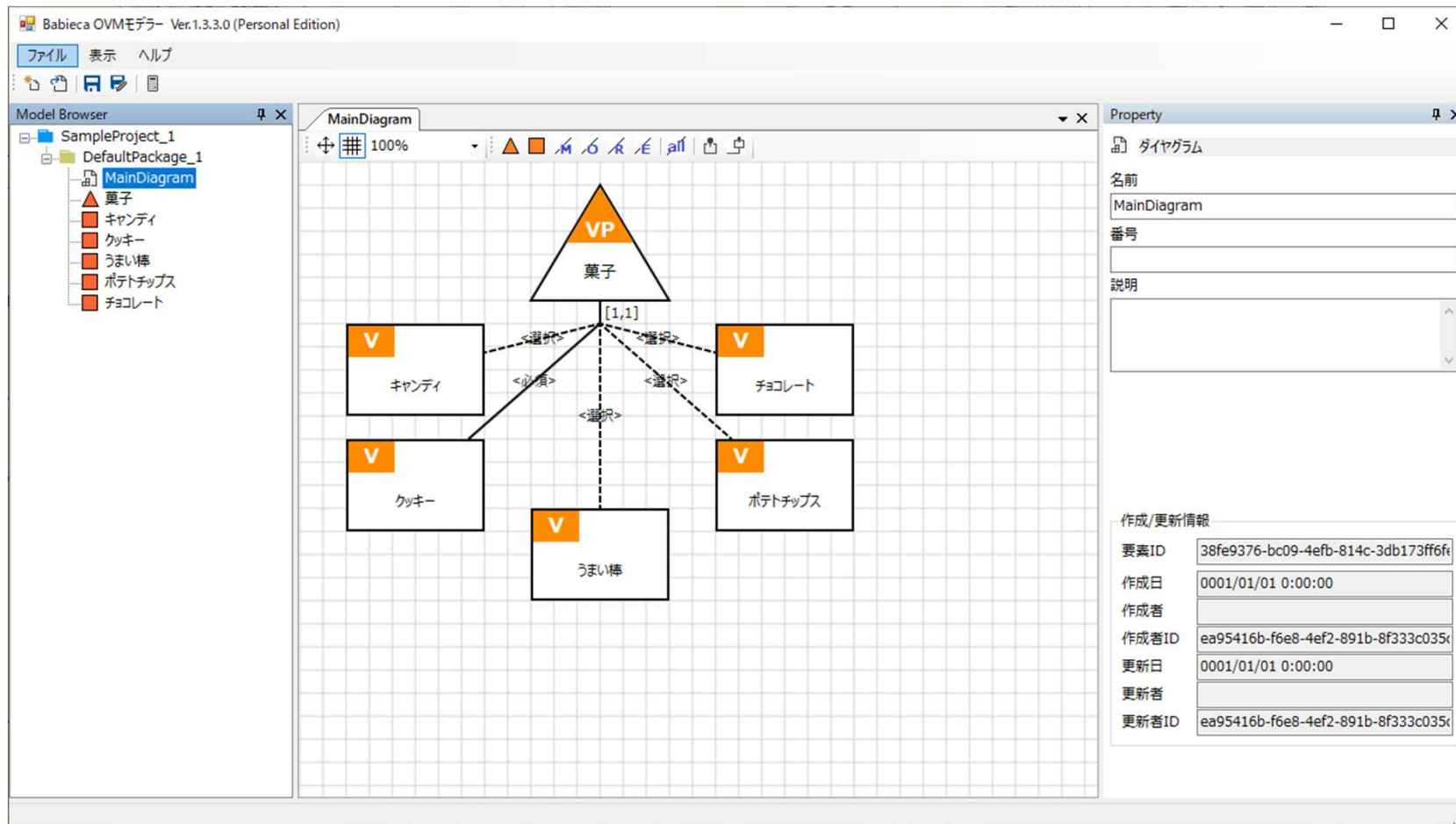


「可変点～選択肢」間の関係を変更する（“必須”⇔“選択”関係）

定義した「必須関係」「選択関係」を右クリックすることで、
関係の種類を変更することができます（“必須”⇔“選択”）



同様の手順を繰り返し、可変点/選択枝/必須関係/選択関係を作成していきます



選択肢の最大選択数を定義する

「プロパティパネル」で「選択肢の最小/最大選択数」を入力し「保存」ボタンを押下します

The screenshot shows the Babieca OVM Modeller interface. On the left, the 'Model Browser' lists '菓子' as a variable point. The main diagram shows a central '菓子' node with five options: 'キャンディ', 'クッキー', 'うまい棒', 'チョコレート', and 'ポテトチップス'. Callouts provide instructions:

- ①「可変点」を選択 (Select the variable point)
- ②「選択肢の最大選択数」を入力 (Enter the maximum selection count for the option)
- ④ダイアグラムに「最大選択数」が反映されます (The maximum selection count is reflected in the diagram)

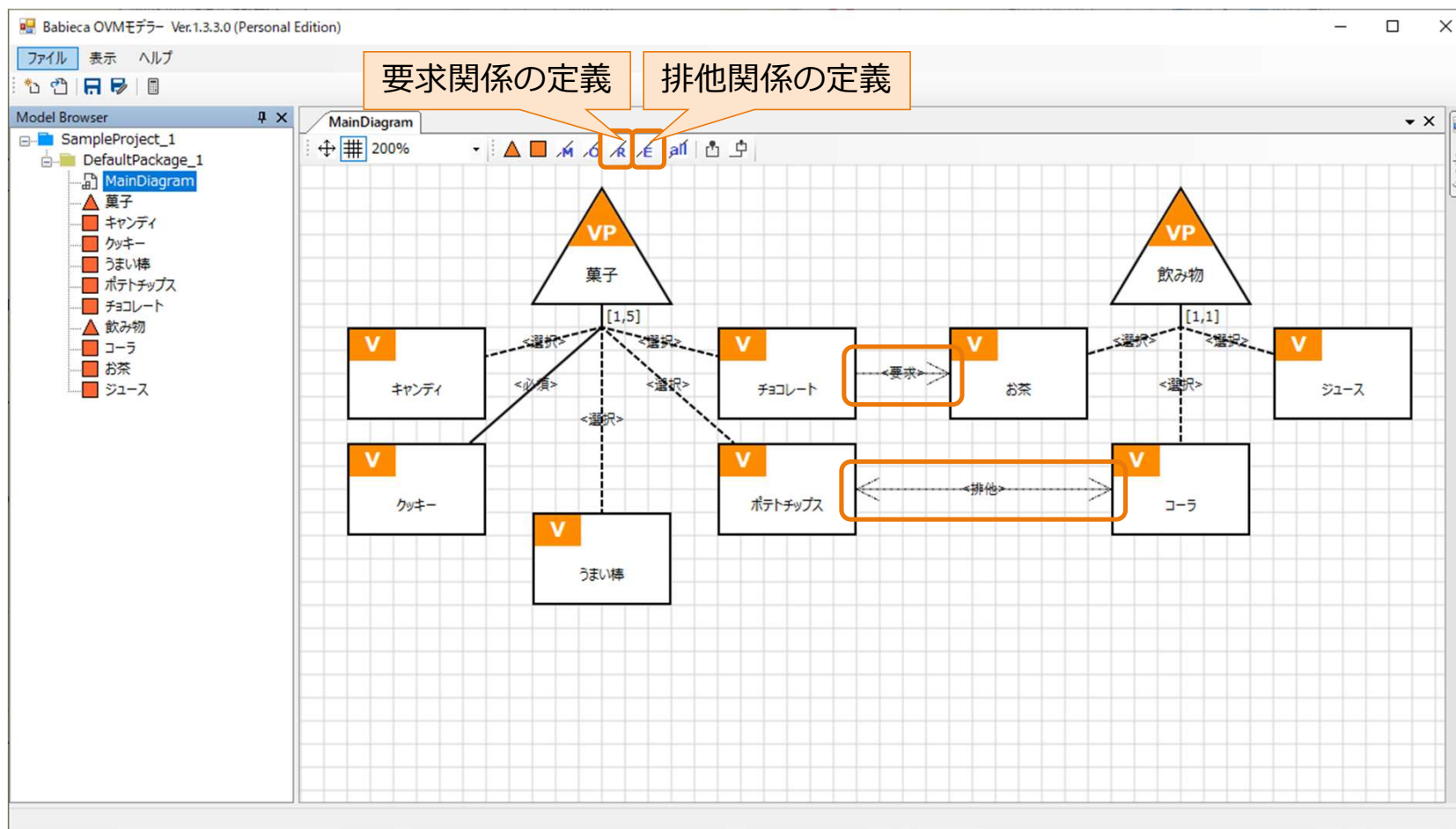
The 'Property' panel on the right shows the settings for the selected variable point '菓子'. The '可変点の設定' (Variable Point Settings) section includes a checkbox for '必須にする' (Make mandatory) and two input fields: '選択肢の最小選択数' (Minimum selection count) set to 1, and '選択肢の最大選択数' (Maximum selection count) set to 5.

プロパティパネルの入力は、下記のタイミングで確定/反映されます。

- ・ Enterキーを押下
- ・ ダイアグラムやモデルブラウザをクリック

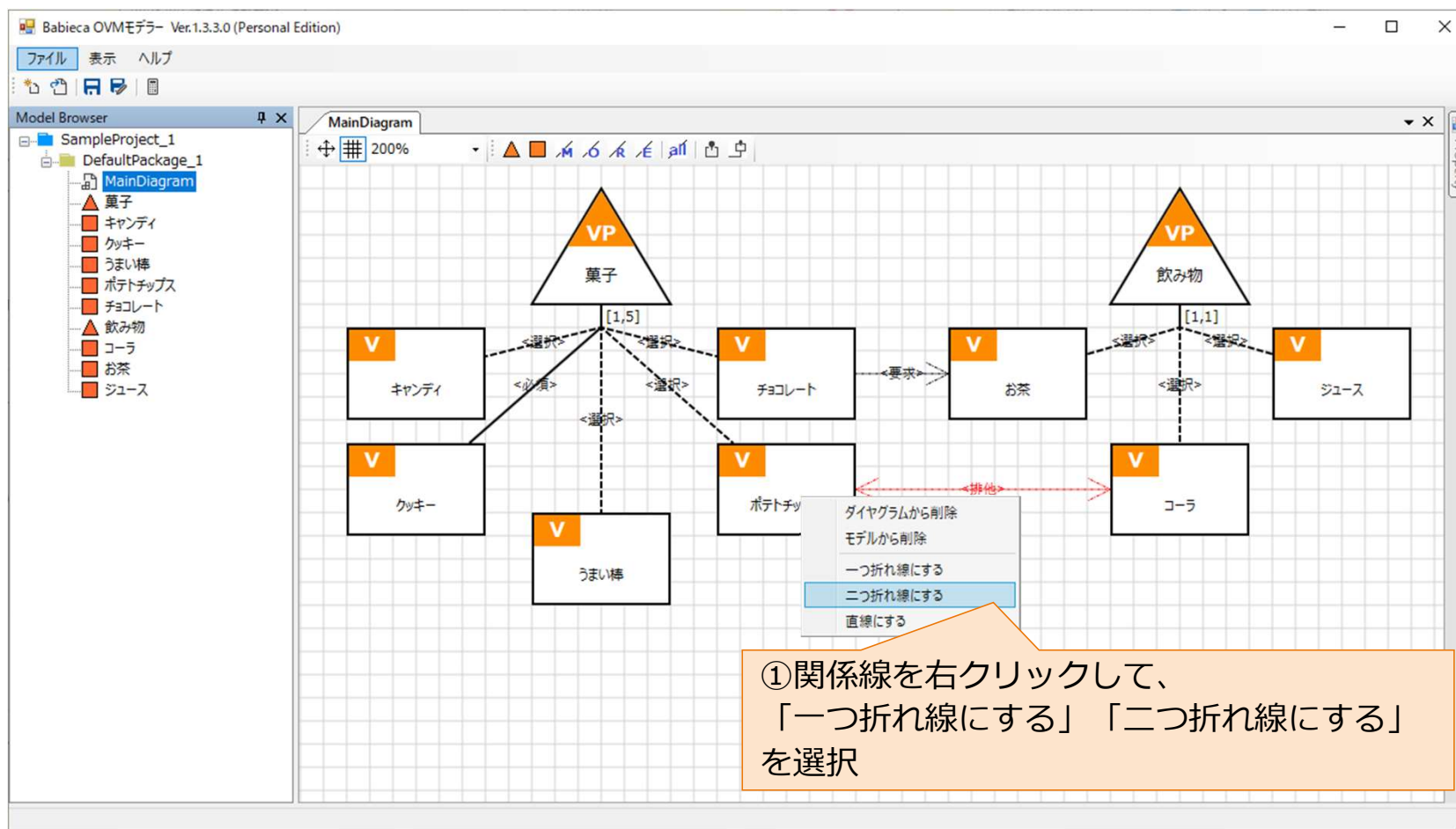
「要求関係」「排他関係」を定義する

「要求関係」と「排他関係」は、可変点-可変点、可変点-選択肢、選択肢-選択肢、いずれにも定義可能です



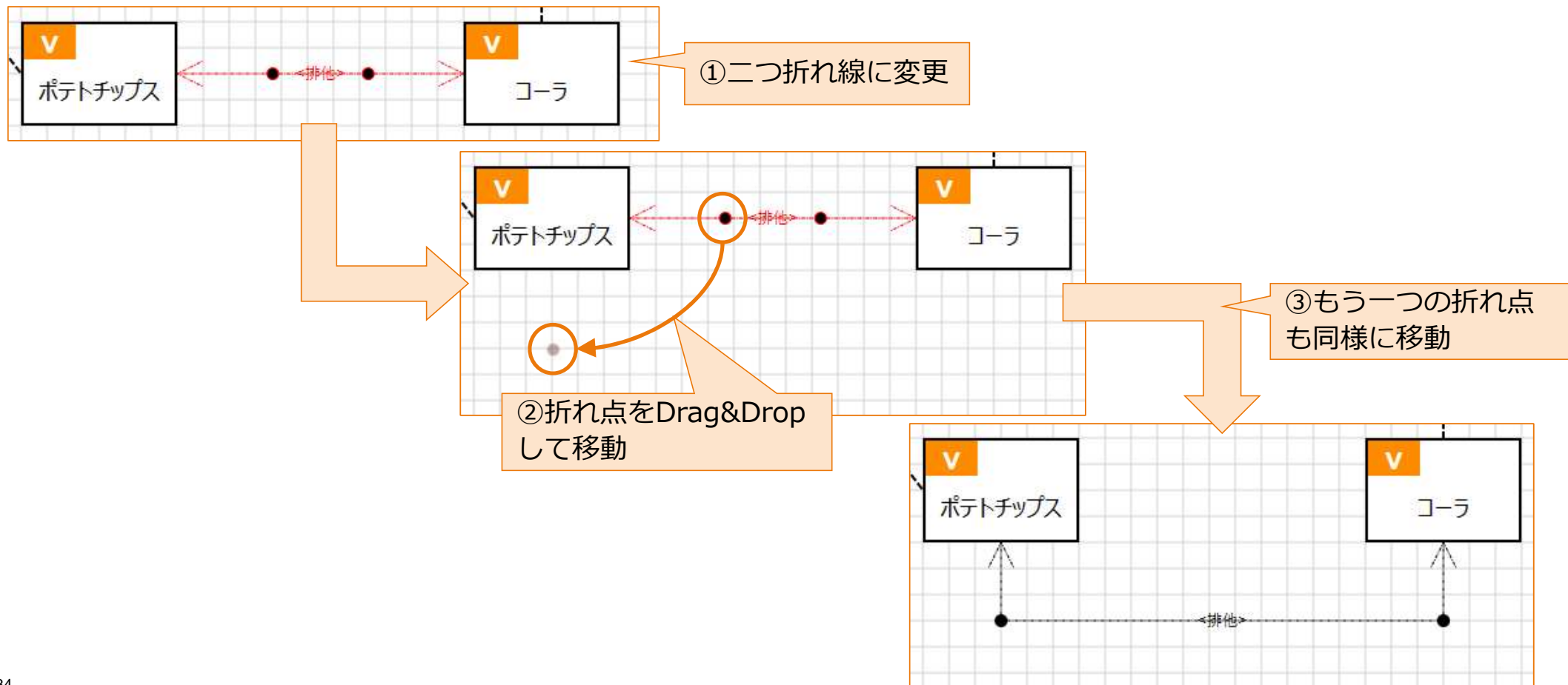
「関係」線のレイアウトを変更する

デフォルトでは関係線は直線ですが、
一つ折れ線/二つ折れ線にしてレイアウトすることが出来ます



「関係」線のレイアウトを変更する

デフォルトでは関係線は直線ですが、
一つ折れ線/二つ折れ線にしてレイアウトすることが出来ます



解の数を表示

画面左下の「電卓アイコン」をクリックすると、そのモデルの解の数が表示されます

ASPソルバーを「探索時間を5秒以内」「探索数を100個以内」として実行

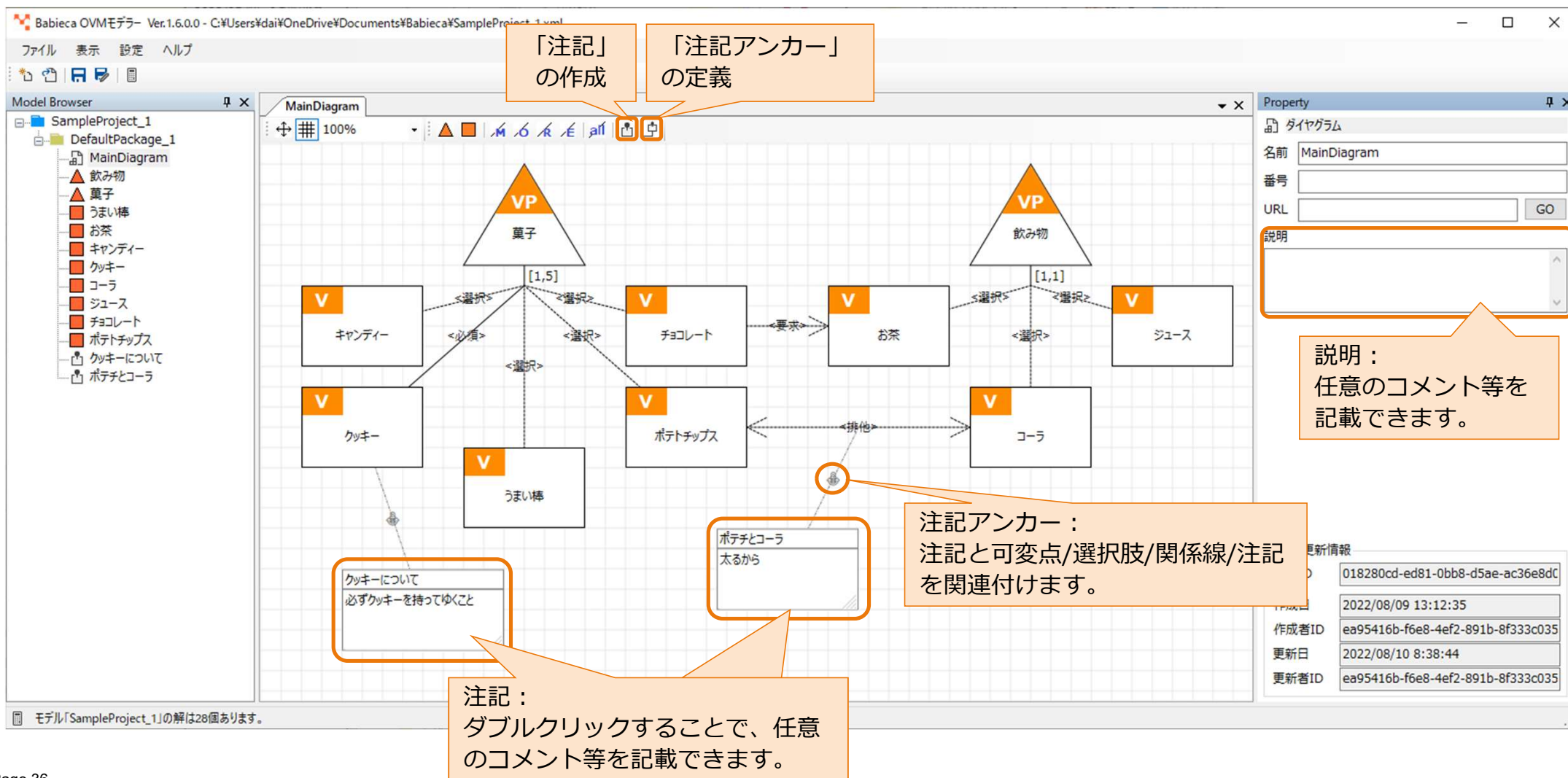
- ・5秒以内にASPソルバーの探索が完了した場合
解が見つかった場合
 - ・モデル「XXX」の解はxxx個あります。
 - ・モデル「XXX」の解は100個以上あります。解が見つからなかった場合
 - ・モデル「XXX」の解はありません。
- ・5秒以内にASPソルバーの探索が完了しなかった場合（途中打ち切り）
解が見つかった場合
 - ・モデル「XXX」の解はxxx個以上あります。解が見つからなかった場合
 - ・解の探索を打ち切りました。モデル「XXX」の解の有無は不明です。

解が100個見つかった時点で探索打ち切り

モデルが巨大な場合等

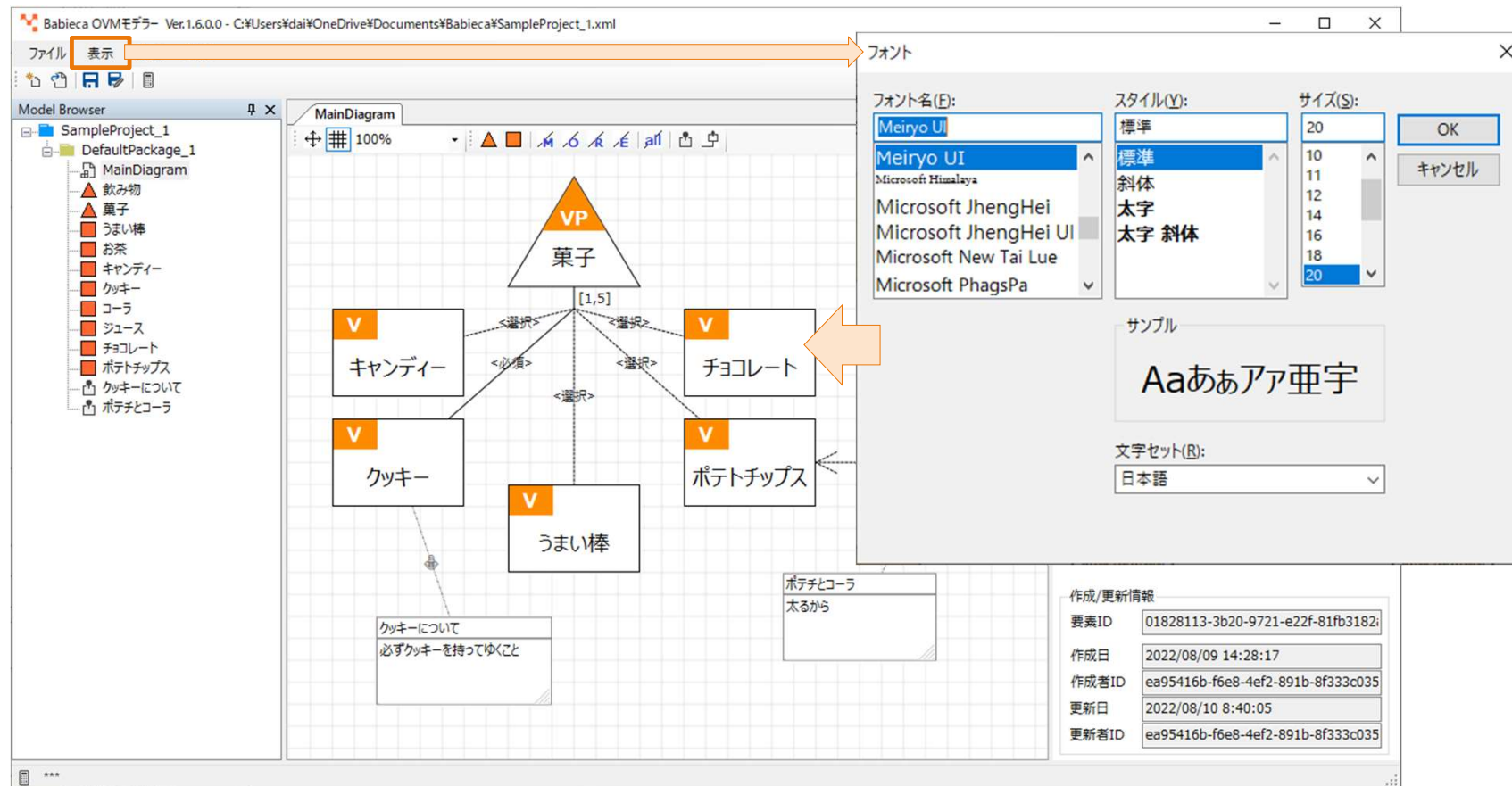
モデル「SampleProject_1」の解は28個あります。

モデル要素へのアノテーション記述



表示フォントの変更

メニュー「表示」→「フォント」でモデル要素の表示フォントを変更することが出来ます
(設定はモデルではなく、PC端末毎に保存/適用されます)

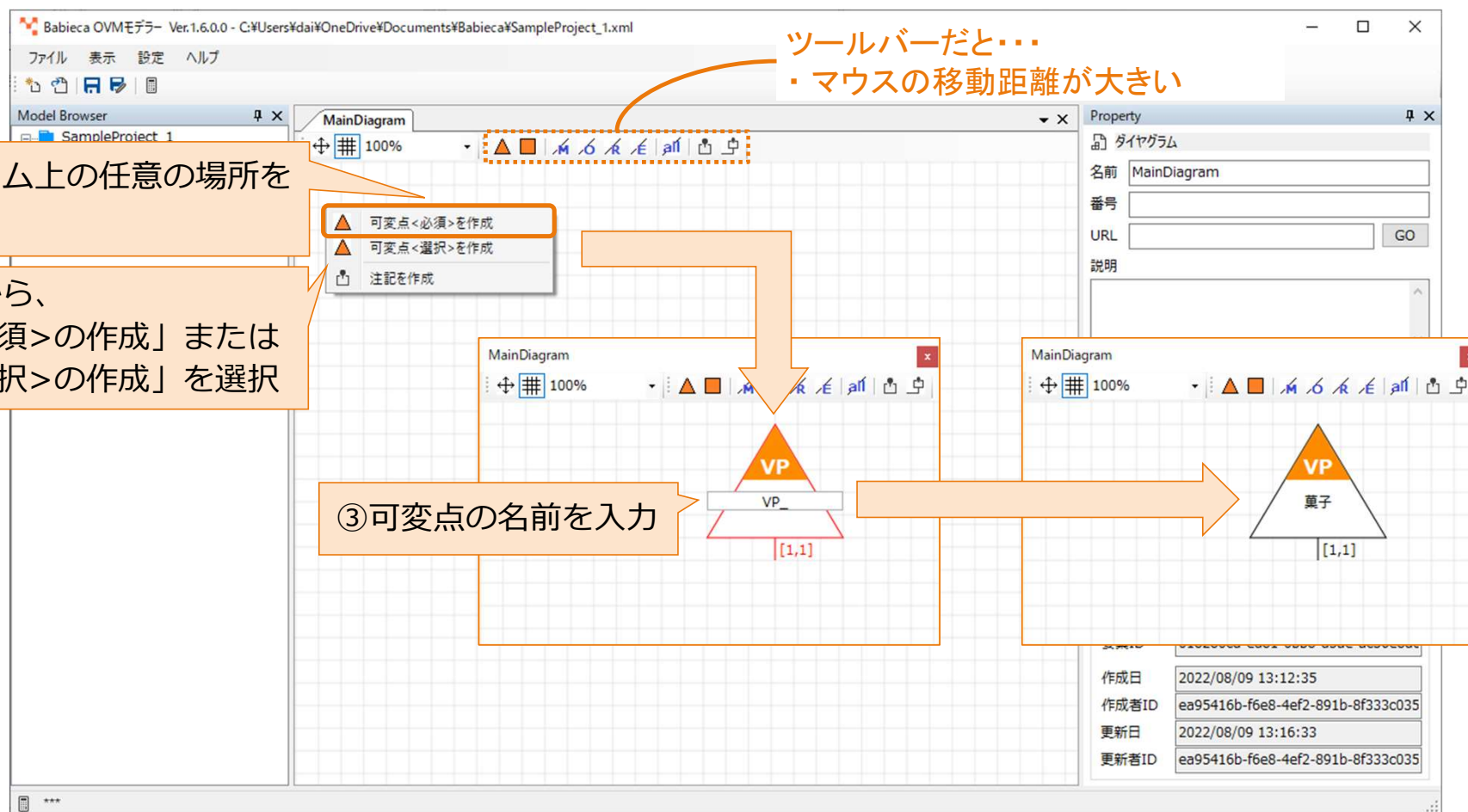


デフォルトは、
・Meiryo UI
・標準スタイル
・サイズ12
です

4-2. OVMモデリング（その2）

「可変点」を作成する

ツールバーの代わりに、**ダイアグラム上の任意の場所**を右クリックすることでも、可変点を作成/配置することができます



「選択肢」を作成する

ツールバーの代わりに、**可変点**を右クリックすることで、
選択肢と関係線を同時に作成/配置することができます

The screenshot illustrates the software interface for creating a selection branch. The main window shows a diagram with a variable point (VP) labeled '菓子' (Kashi). A context menu is open, showing options to create a selection branch. The menu options are:

- 選択肢<必須>を作成
- 選択肢<選択>を作成
- <必須>関係を作成
- <選択>関係を作成
- <要求>関係を作成
- <排他>関係を作成
- 関係を表示
- 注記を作成
- 注記アンカーを作成
- ダイアグラムから削除
- モデルから削除

The diagram shows the variable point (VP) connected to a selection branch (V) labeled 'キャンディー' (Candy). The relationship is labeled '<選択>' (Selection) with a cardinality of '[1,1]'. The selection branch is represented by a box with a 'V' icon and the name 'キャンディー'.

①可変点を右クリック

②メニューから、「選択肢<必須>の作成」または「選択肢<選択>の作成」を選択

③ダイアグラム上の任意の場所を左クリックすると、選択肢と関係線が同時に作成/配置されます

④選択肢の名を入力

ツールバーだと・・・
・マウスの移動距離が大きい

MainDiagram
100%

Property
可変点
名前 菓子
番号
URL
説明

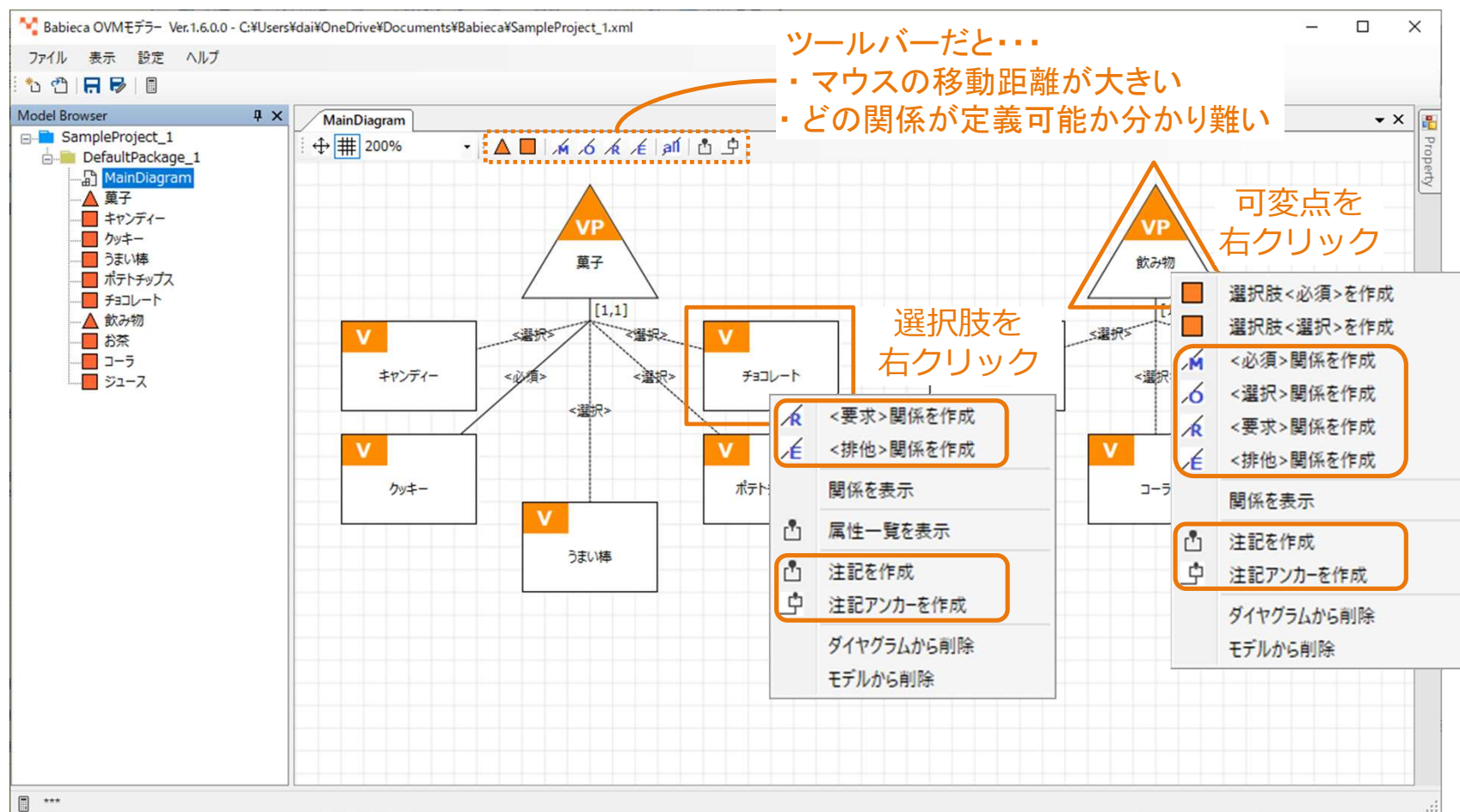
可変点の設定

MainDiagram
100%

更新日 2022/08/09 13:34:30
更新者ID ea95416b-f6e8-4ef2-891b-8f333c035

「関係」を定義する

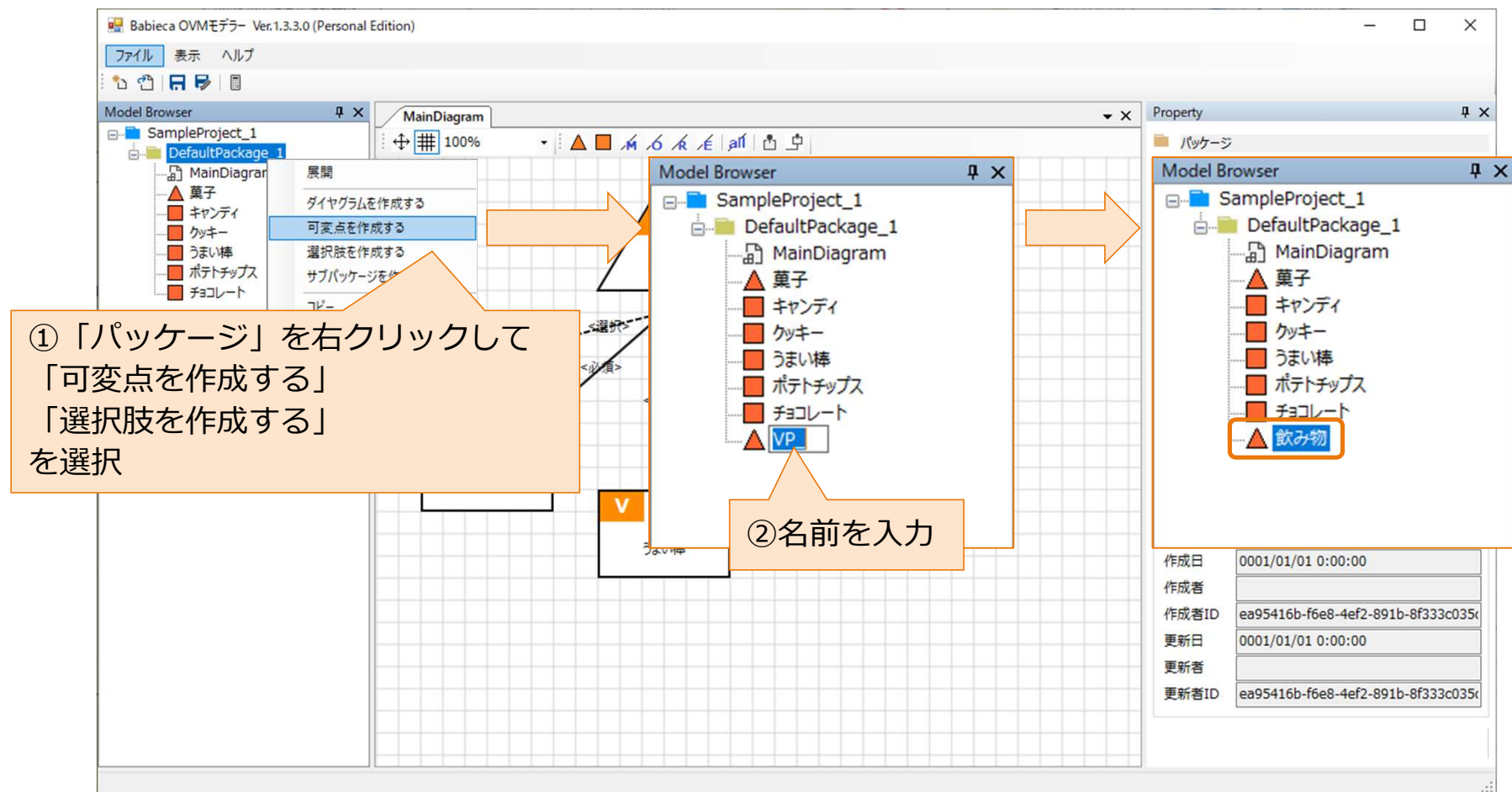
ツールバーの代わりに、可変点や選択枝を右クリックすることで、**定義可能な関係**を作成/配置することができます



4-3. OVMモデリング（その3）

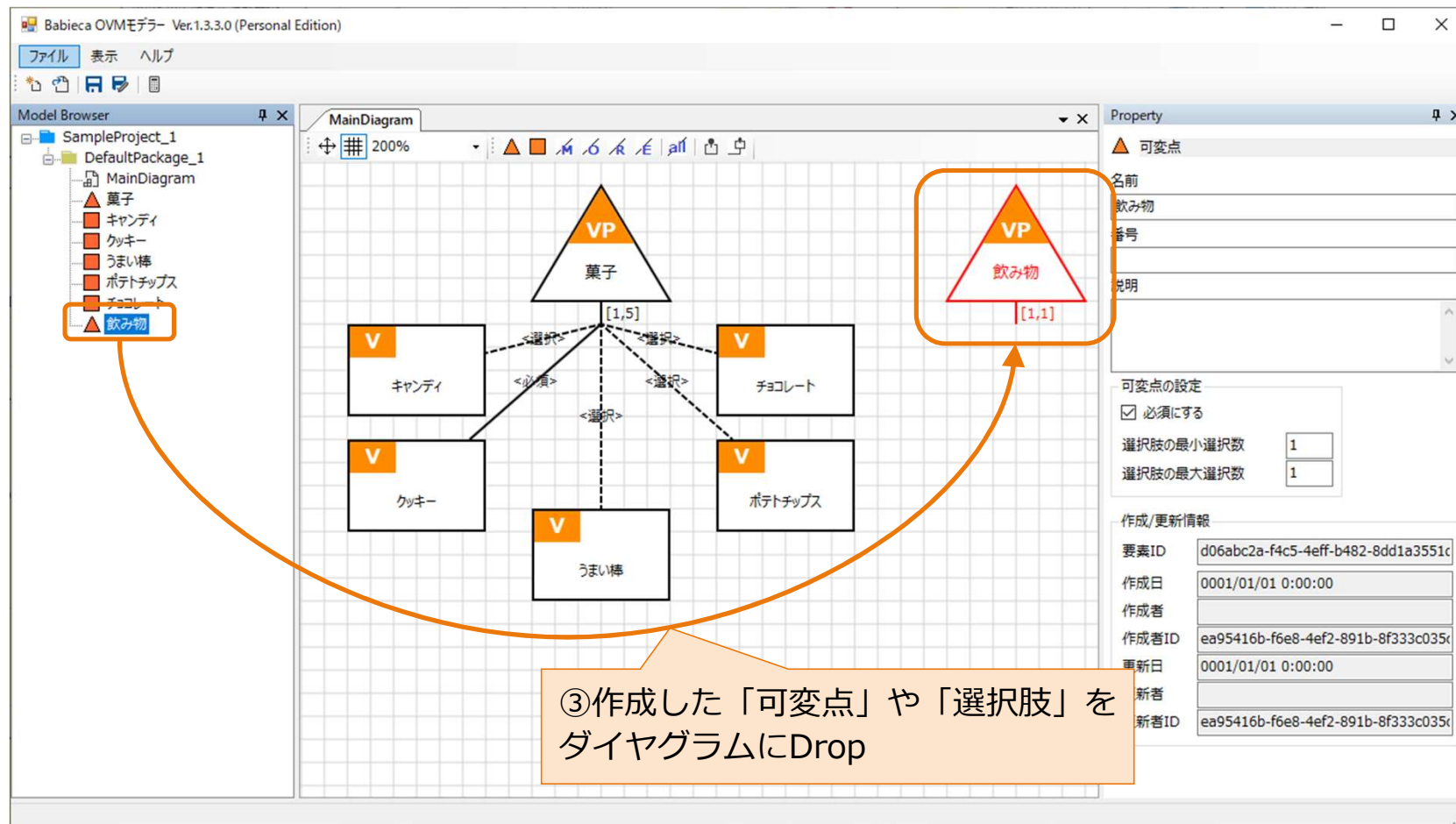
「可変点」「選択肢」を作成する

「モデル・ブラウザ」上で「可変点」や「選択肢」を作成し、それをダイアグラム上にDropする方法もあります（次項「モデルとビュー」参照）



「可変点」「選択肢」を作成する

「モデル・ブラウザ」上で「可変点」や「選択肢」を作成し、それをダイアグラム上にDropする方法もあります（次項「モデルとビュー」参照）



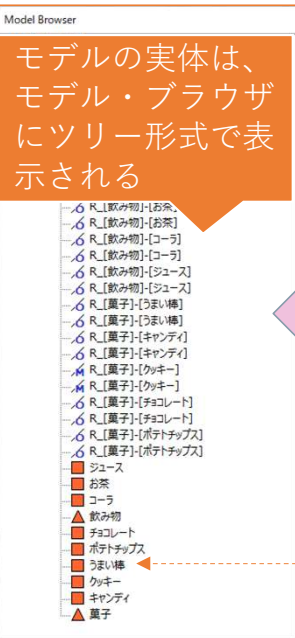
5. モデルとビュー

「モデル」と「ビュー」

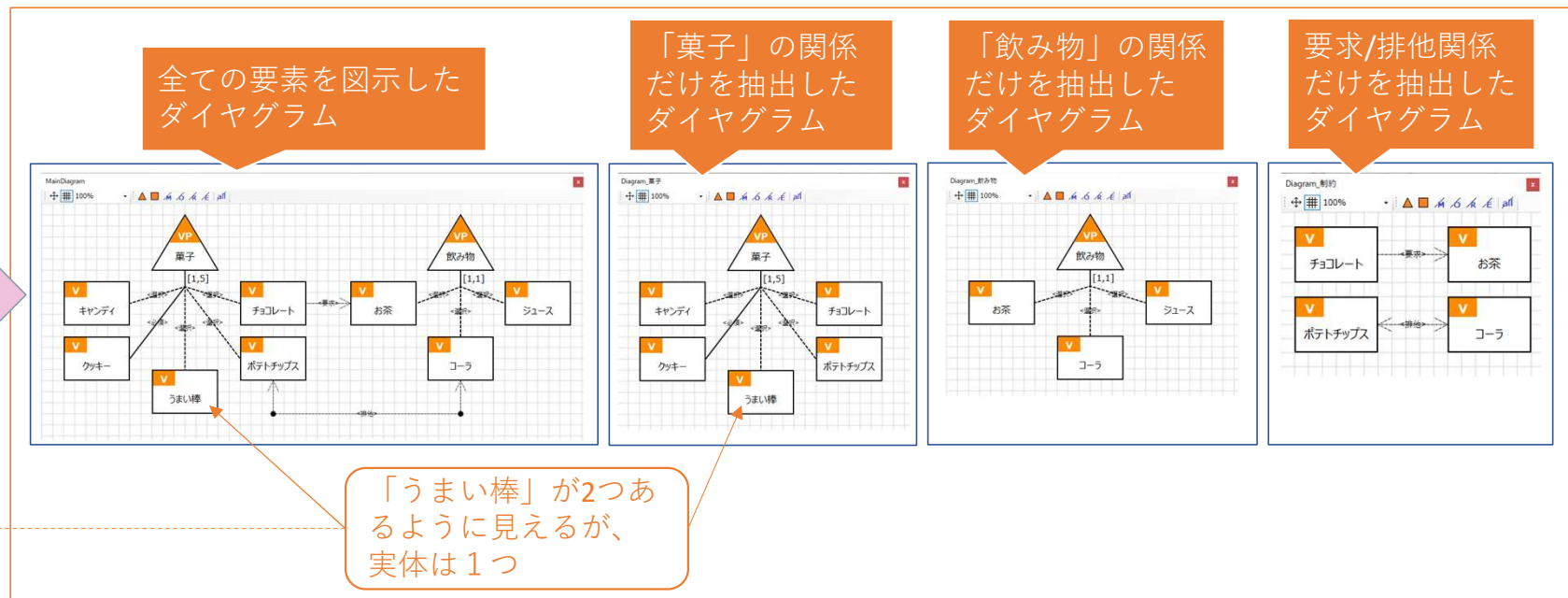
このモデラーは一般的なモデラー同様、「モデルとビューの概念」を採用しています

- “モデル”はモデルに定義された要素そのもの(実体)
→ モデル・ブラウザに表示される
- “ビュー”はモデルに定義された要素とその関係を“ある観点”で抽出し図示したもの（投影図）
→ ダイアグラムに表示される

“モデル”

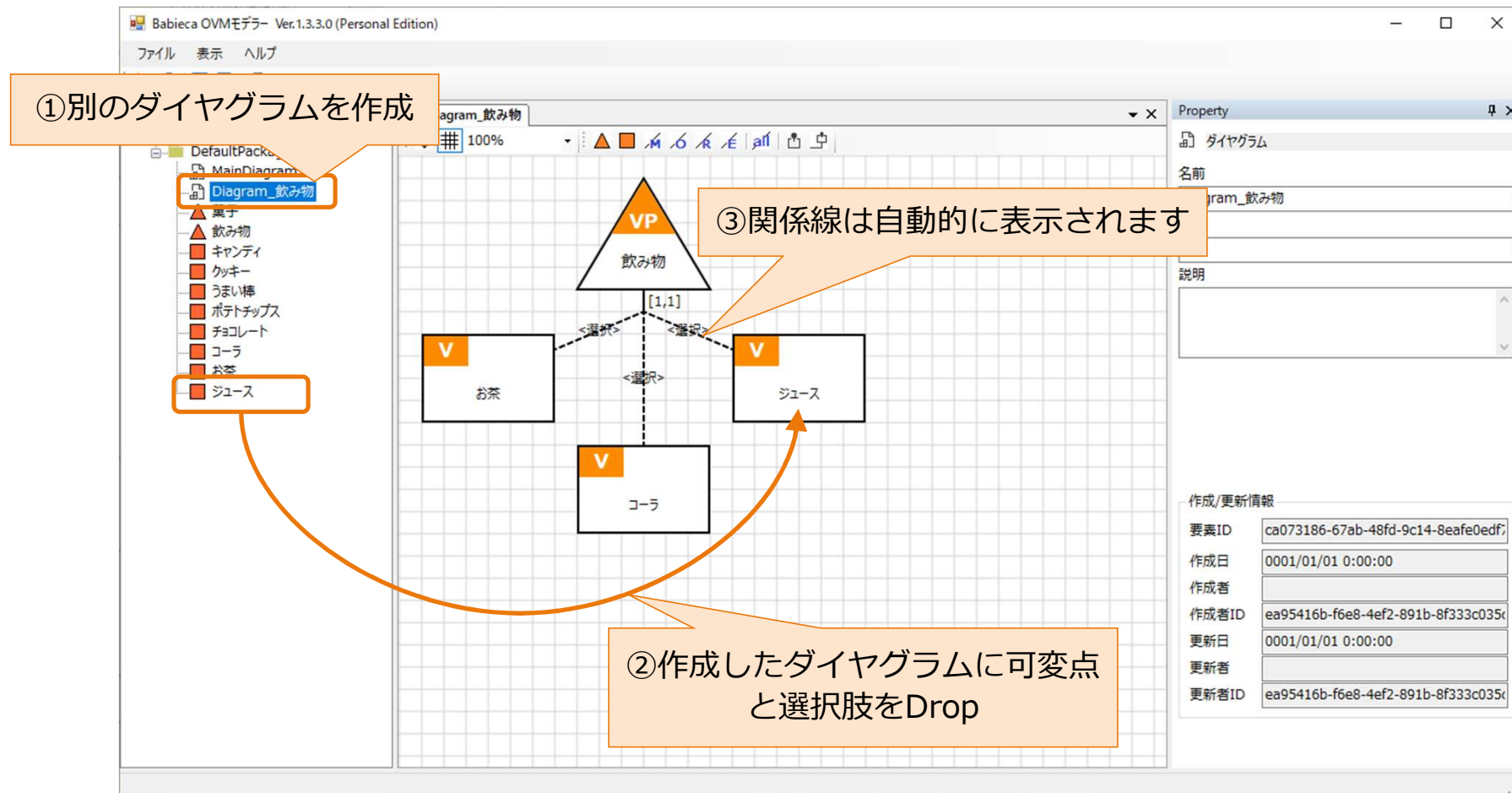


“ビュー”



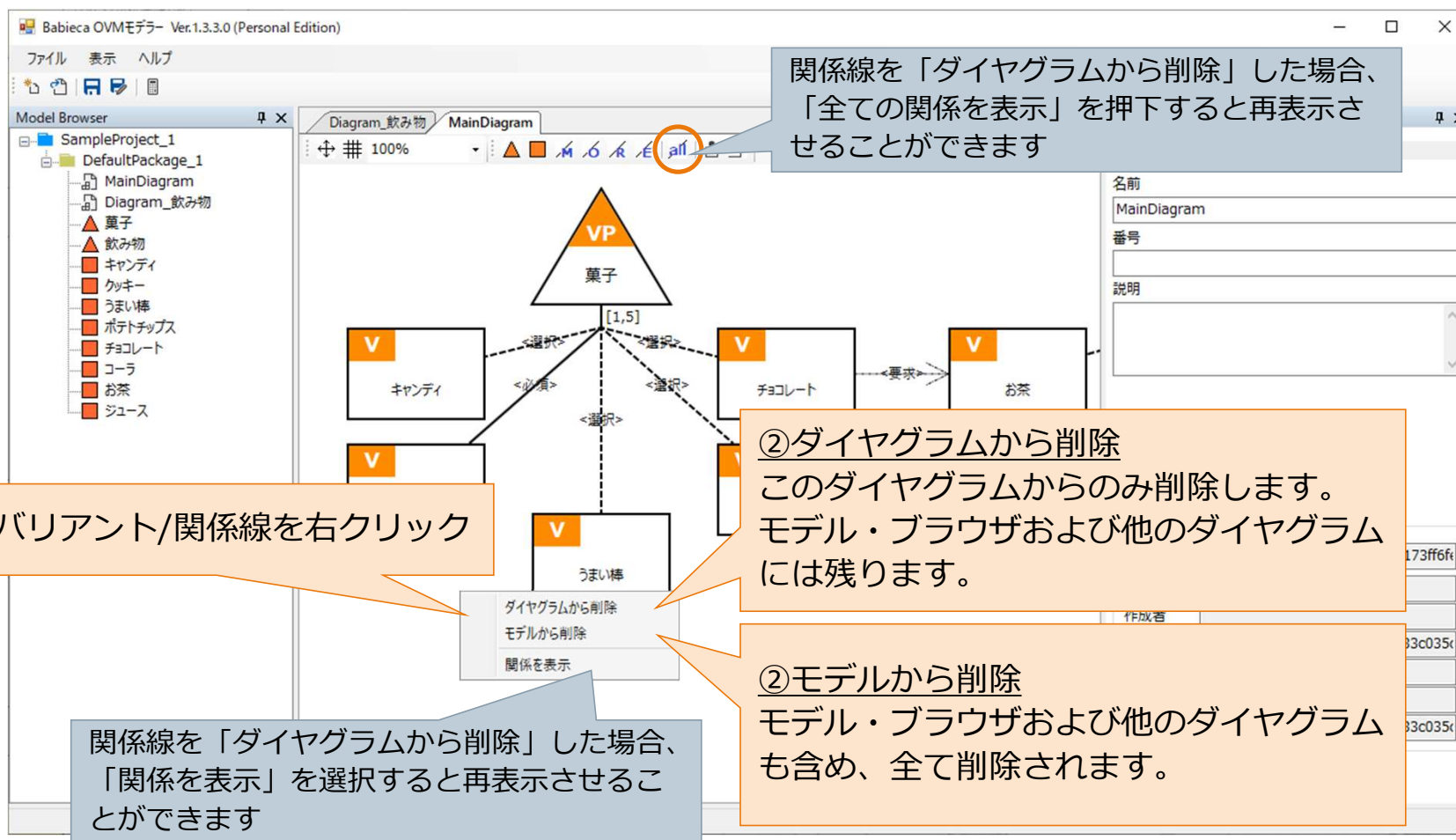
「モデル」と「ビュー」：一部分を抽出したダイアグラムを作成

一枚のダイアグラムに全ての可変点/選択肢を配置するだけではなく、
観点毎に個別のダイアグラムを作成することができます



モデル要素を削除する

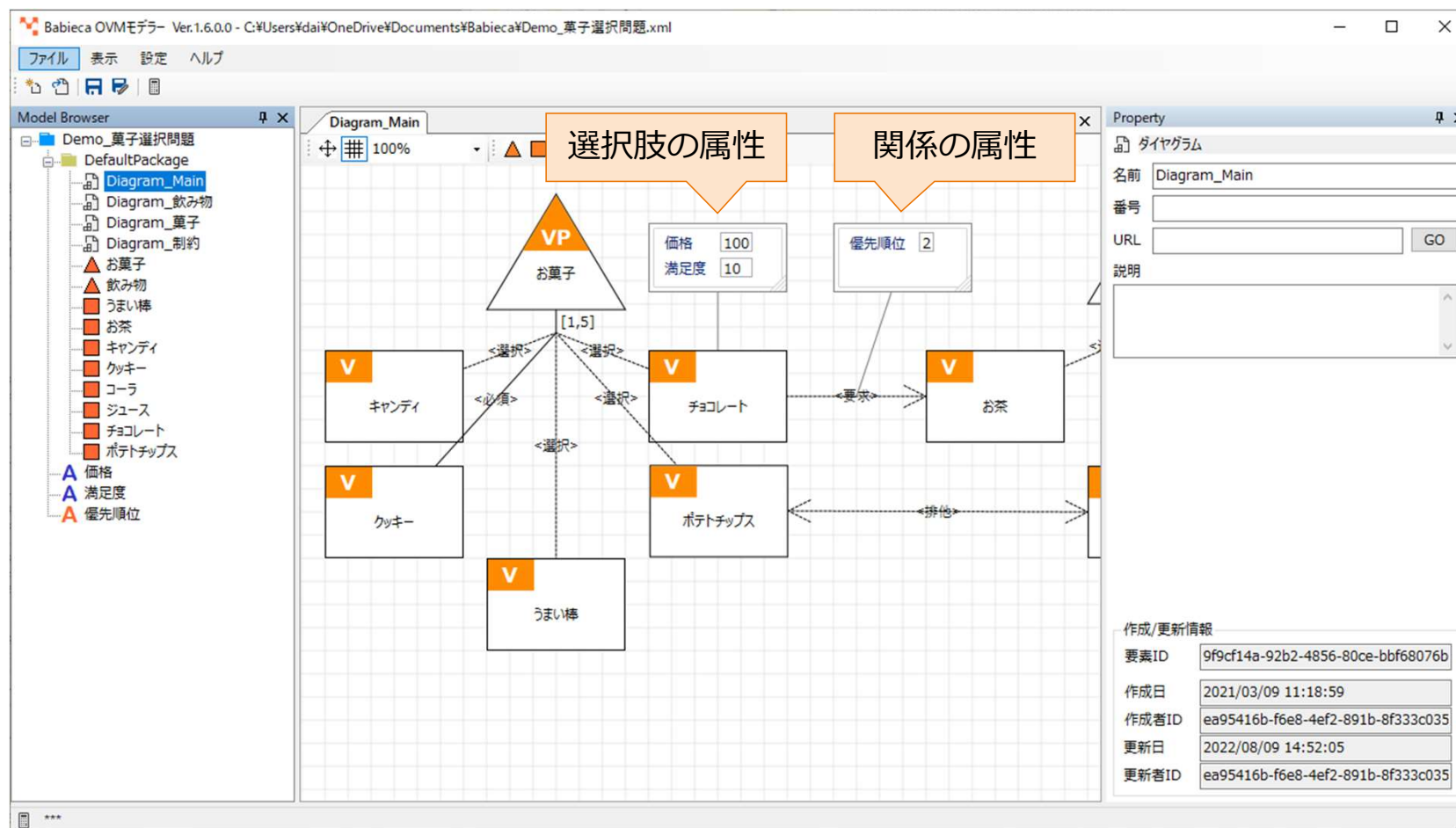
「削除」には、「ダイアグラムから削除」と「モデルから削除」の2通りがあります



6. 属性の作成

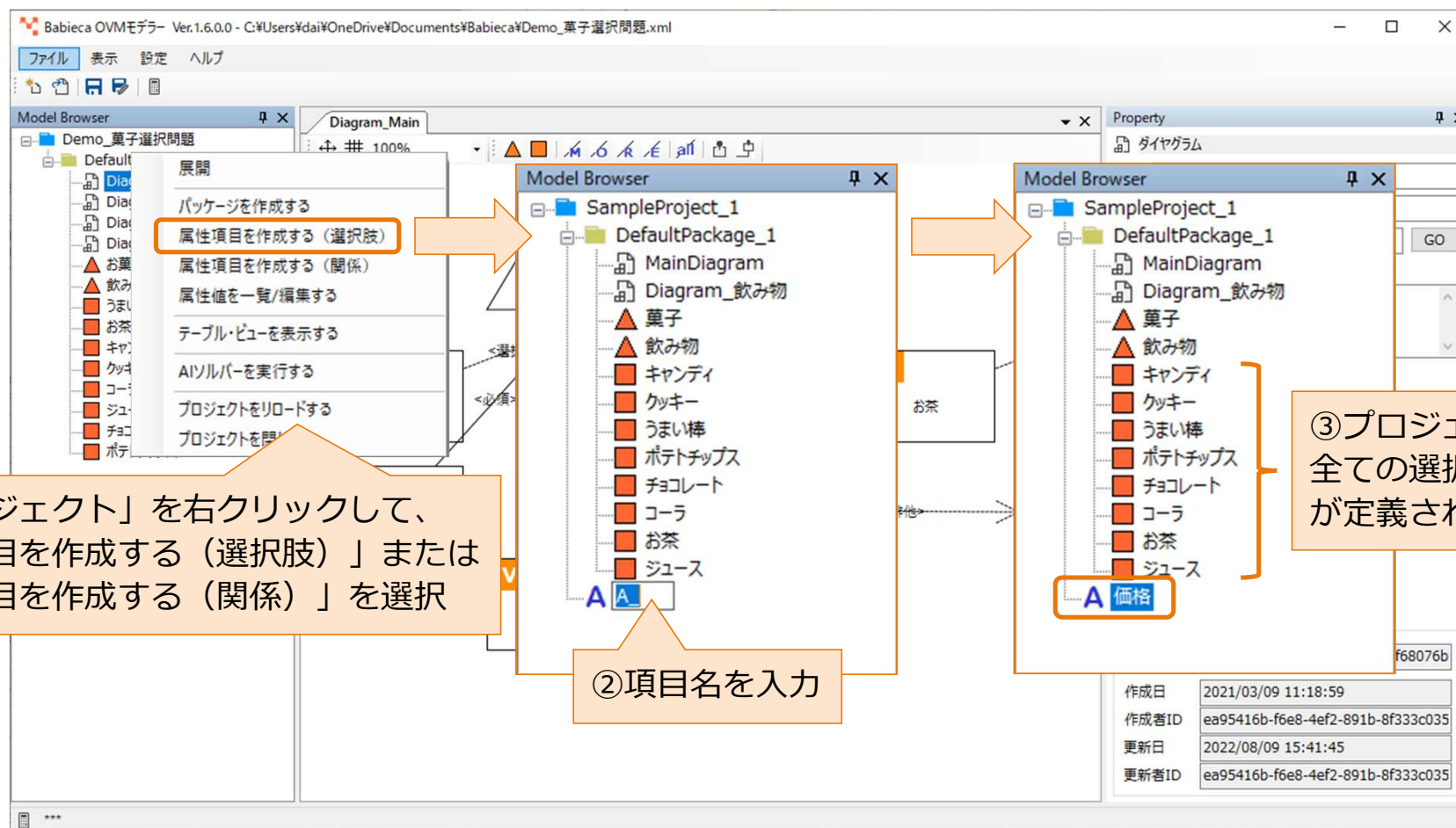
「属性」とは

「属性」とは、「属性項目」と「属性値」のペアからなるデータで、**選択肢**、または**関係**に対して定義することができます。



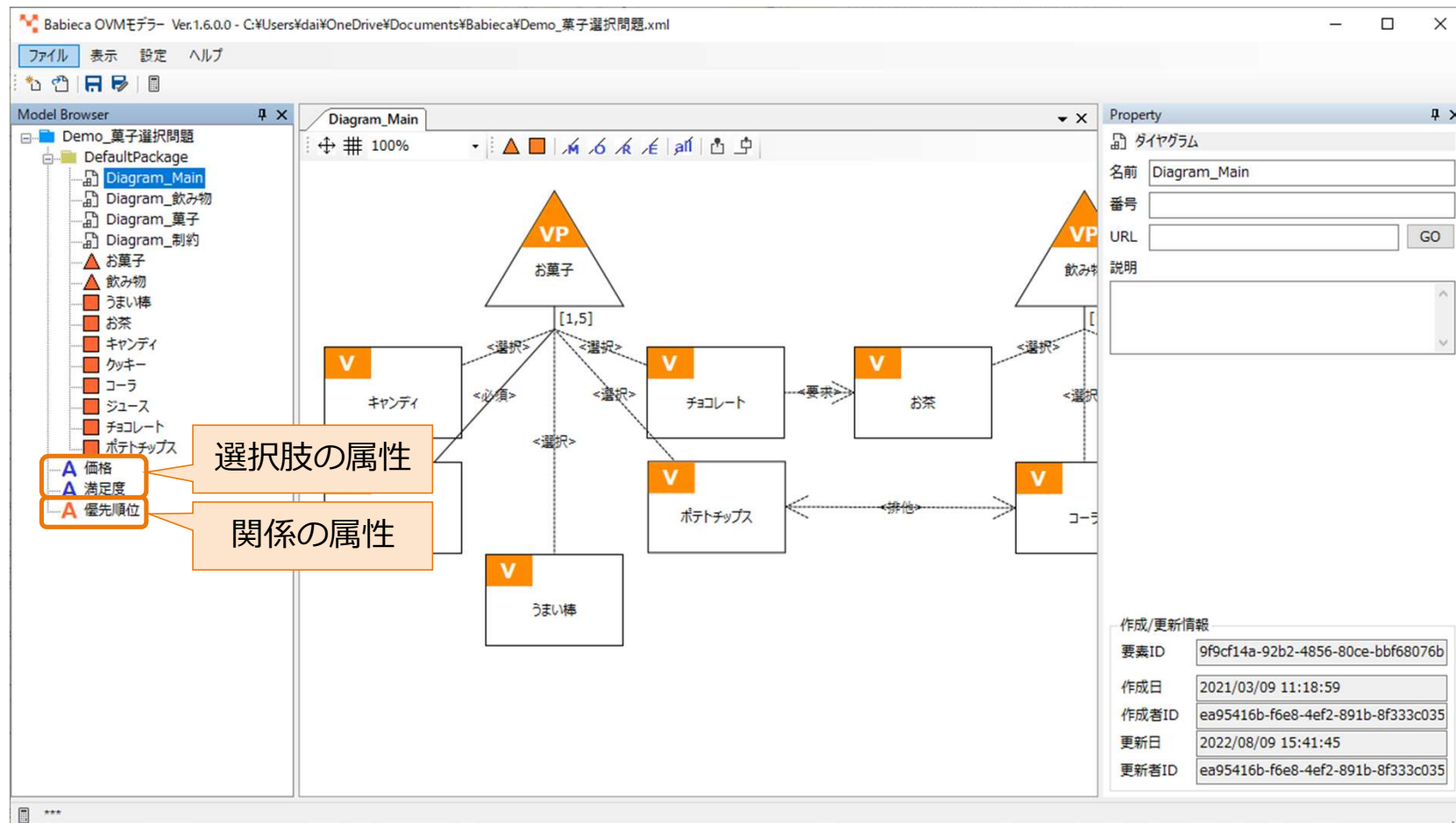
「属性項目」を定義する

「属性項目」は、プロジェクト毎に定義します



「属性項目」を定義する

「属性項目」は複数定義可能です



「属性値」を設定する

一括編集も可能です
(次頁)

「プロパティパネル」で選択肢または関係の「属性値」を入力します
※AIソルバが対応している属性値は**“整数”**のみです

①「選択肢」または「関係」を選択

The screenshot shows the Babieca OVM Modeller interface. The main window displays a diagram with a central node 'お菓子' (Candy) connected to several other nodes: 'キャンディ' (Candy), 'チョコレート' (Chocolate), 'お茶' (Tea), 'クッキー' (Cookie), 'ポテトチップス' (Potato Chips), and 'うまい棒' (Umai棒). The 'キャンディ' node is highlighted with a red border. The 'Property' panel on the right shows the selected node's properties: '名前' (Name) is 'キャンディ', '番号' (Number) is empty, 'URL' is empty, and '説明' (Description) is empty. Below the '説明' field is a table with two columns: '属性項目名' (Attribute Item Name) and '値' (Value). The table contains two rows: '価格' (Price) with value '20' and '満足度' (Satisfaction) with value '3'. The '価格' row is highlighted with a blue background. An orange callout box points to the table with the text '②「属性値」を入力' (Enter attribute value). Another orange callout box points to the 'キャンディ' node with the text '①「選択肢」または「関係」を選択' (Select choice or relationship). A grey callout box points to the table with the text '※AIソルバが対応している属性値は整数のみです' (Attribute values supported by the AI solver are integers only). A yellow callout box at the bottom right contains the text: 'プロパティパネルの入力は、下記のタイミングで確定/反映されます。' (Input in the property panel is confirmed/reflected at the following timing.) followed by a bulleted list: '・Enterキーを押下' (Press Enter key) and '・ダイアグラムやモデルブラウザをクリック' (Click diagram or model browser).

属性項目名	値
価格	20
満足度	3

プロパティパネルの入力は、下記のタイミングで確定/反映されます。

- ・Enterキーを押下
- ・ダイアグラムやモデルブラウザをクリック

「属性値」を設定する

属性値を一覧/一括編集することもできます

The screenshot shows the Babieca OVM Modeller interface. On the left, the 'Model Browser' shows a project named 'Demo_菓子選択問題'. A right-click context menu is open over the project, with the option '属性値を一覧/編集する' (View/Edit Attribute Values) highlighted. An orange arrow points from this menu item to a dialog box titled '属性値を一覧/編集する' (View/Edit Attribute Values).

The dialog box contains two tables:

選択肢の属性 (Attribute of Choice)

選択肢	価格	満足度
うまい棒	30	7
お茶	0	1
キャンディ	20	3
クッキー	80	4
コーラ	50	10
ジュース	30	5
チョコレート	100	10
ポテトチップス	120	8

関係の属性 (Attribute of Relationship)

関係	優先度
R_[菓子]-[うまい棒]	1
R_[菓子]-[キャンディ]	1
R_[菓子]-[クッキー]	1
R_[菓子]-[チョコレート]	1
R_[菓子]-[ポテトチップス]	1
R_[飲み物]-[お茶]	1
R_[飲み物]-[コーラ]	1
R_[飲み物]-[ジュース]	1
R_[チョコレート]-[お茶]	2
R_[ポテトチップス]-[コーラ]	3

At the bottom of the dialog, there is a '満足度' (Satisfaction) field with the value '8' and a metadata section:

要素ID	67aa3e07-b112-411e-8cc4-51e6630E
作成日	2021/03/09 11:18:59
作成者ID	ea95416b-f6e8-4ef2-891b-8f333c035
更新日	2022/08/09 15:47:17
更新者ID	ea95416b-f6e8-4ef2-891b-8f333c035

An orange callout box points to the context menu with the text: 「プロジェクト」を右クリックして「属性値を一覧/編集する」を選択 (Right-click the 'Project' and select 'View/Edit Attribute Values').

「属性」を表示する

選択枝や関係の属性をダイアグラム上に表示させることができます

The screenshot shows the Babieca OVM Modeller interface. The main diagram area displays a hierarchy of nodes. A context menu is open over the 'ポテトチップス' (Potato Chips) node, with the option '属性一覧を表示' (Show attribute list) highlighted. An orange callout box points to this menu item with the text: ①モデル要素を右クリックして「属性一覧を表示」を選択. Another orange callout box points to the attribute list for 'ポテトチップス' with the text: ②ダイアグラム上の任意の場所をクリック. The attribute list shows '価格' (Price) as 120 and '満足度' (Satisfaction) as 8. The right-hand 'Property' panel shows the details for the selected element 'ポテトチップス'.

①モデル要素を右クリックして「属性一覧を表示」を選択

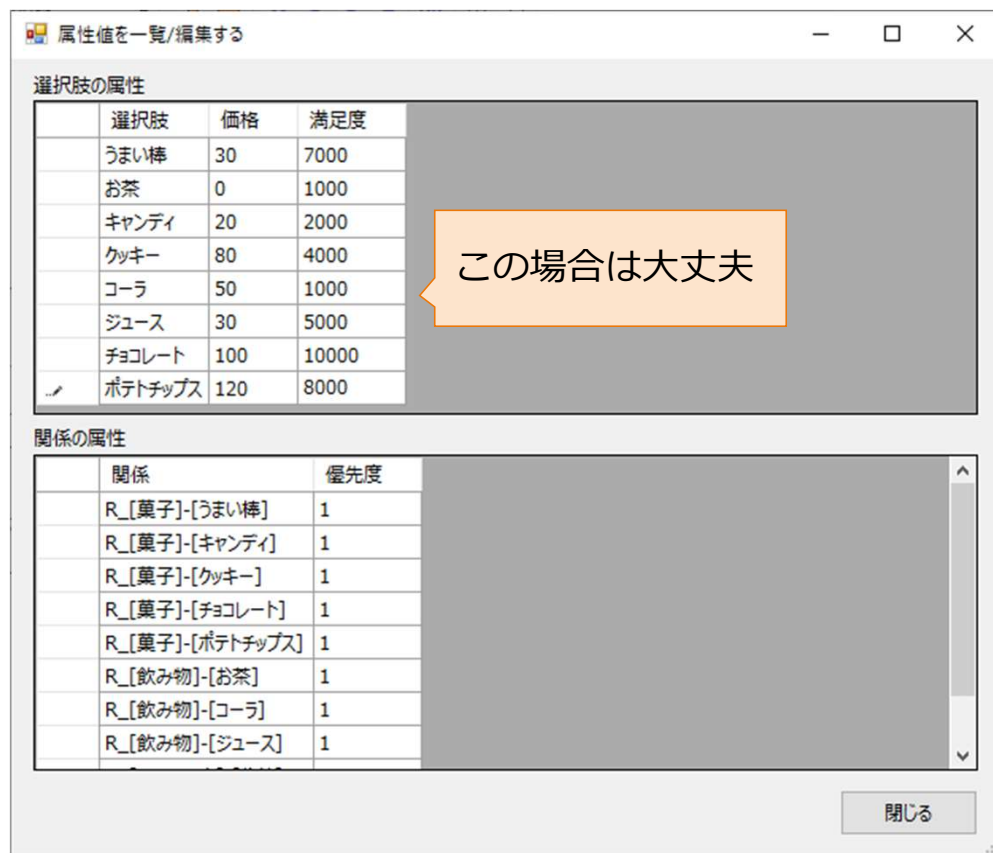
②ダイアグラム上の任意の場所をクリック

属性項目名	値
価格	120
満足度	8

作成日	2021/03/09 11:18:59
作成者ID	ea95416b-f6e8-4ef2-891b-8f333c035
更新日	2022/08/09 15:47:17
更新者ID	ea95416b-f6e8-4ef2-891b-8f333c035

※「属性値」についての補足

属性値の値の「桁数の幅」が非常に大きい場合（例:1桁と5桁の値が混在）、ASPソルバーの性質から演算速度が大幅に低下する場合があります。※現在対応中



属性値を一覧/編集する

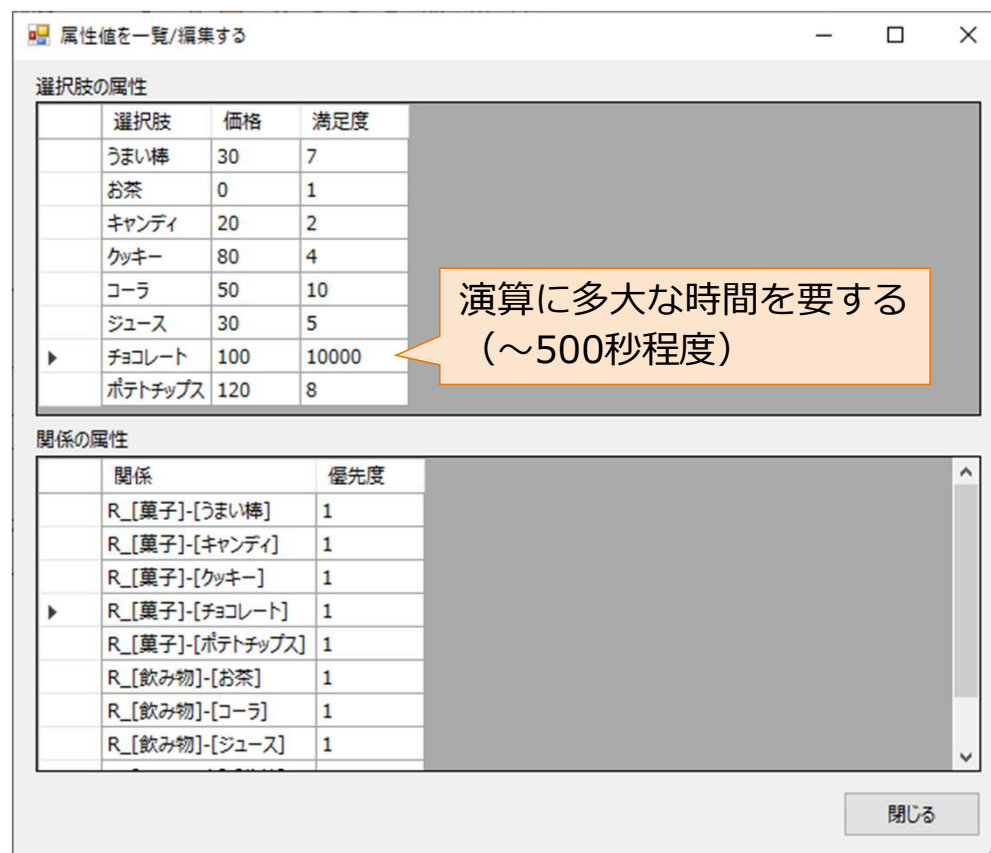
選択肢の属性

選択肢	価格	満足度
うまい棒	30	7000
お茶	0	1000
キャンディ	20	2000
クッキー	80	4000
コーラ	50	1000
ジュース	30	5000
チョコレート	100	10000
ポテトチップス	120	8000

関係の属性

関係	優先度
R_[菓子]-[うまい棒]	1
R_[菓子]-[キャンディ]	1
R_[菓子]-[クッキー]	1
R_[菓子]-[チョコレート]	1
R_[菓子]-[ポテトチップス]	1
R_[飲み物]-[お茶]	1
R_[飲み物]-[コーラ]	1
R_[飲み物]-[ジュース]	1

閉じる



属性値を一覧/編集する

選択肢の属性

選択肢	価格	満足度
うまい棒	30	7
お茶	0	1
キャンディ	20	2
クッキー	80	4
コーラ	50	10
ジュース	30	5
▶ チョコレート	100	10000
ポテトチップス	120	8

関係の属性

関係	優先度
R_[菓子]-[うまい棒]	1
R_[菓子]-[キャンディ]	1
R_[菓子]-[クッキー]	1
▶ R_[菓子]-[チョコレート]	1
R_[菓子]-[ポテトチップス]	1
R_[飲み物]-[お茶]	1
R_[飲み物]-[コーラ]	1
R_[飲み物]-[ジュース]	1

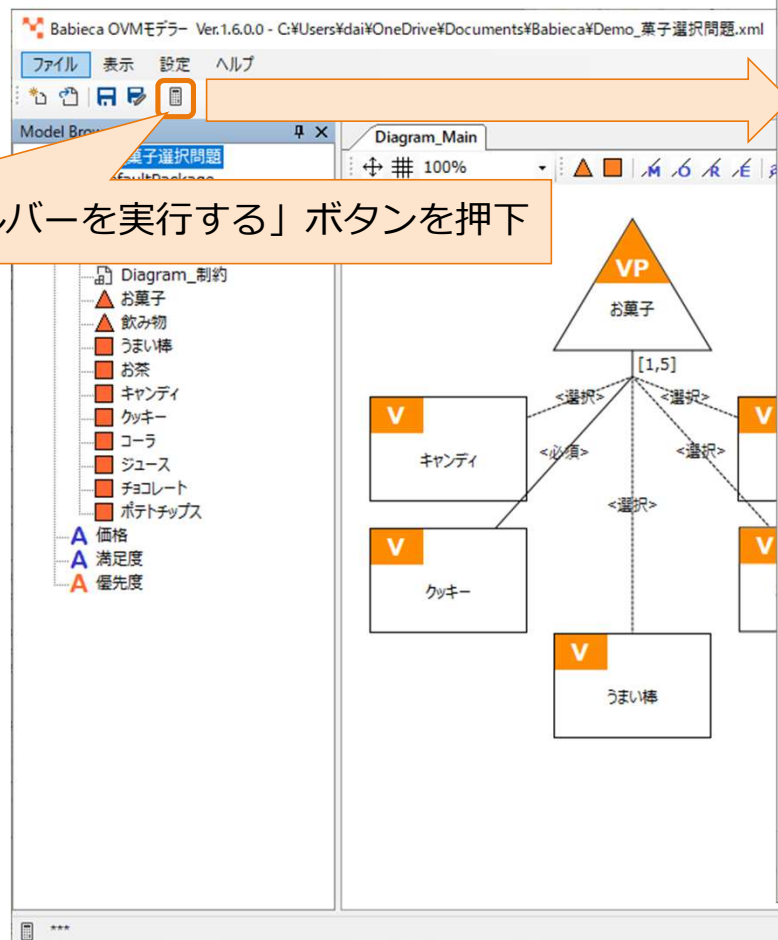
閉じる

7. AIソルバーの実行

AIソルバーを実行する

AIソルバーはプロジェクト単位で
実行させることができます

①「AIソルバーを実行する」ボタンを押下



The Solver (ソルバー) dialog box is shown with the 'Solver Settings and Execution' (ソルバーの設定と実行) tab selected. The 'Solver Type' (ソルバータイプ) is set to 'ASP Solver (clingo)'. The 'Time Out (sec/0 is unlimited)' (タイムアウト) is 30, and the 'Maximum Number of Solutions (0 is unlimited)' (※解の最大数) is 100. The 'Selection Branch Properties' (選択枝の属性) table lists variables W1 (Price) and W2 (Satisfaction). The 'Relationship Properties' (関係の属性) table lists variable W3 (Priority). The 'Constraints (Individual Values)' (制約 (個別値)) and 'Objective Function' (目的関数) sections are empty. The 'Excel Output' (Excel出力) section has 'Horizontal' (横書) selected. The 'Execute' (実行) button is highlighted.

変数	項目名
W1	価格
W2	満足度

変数	項目名
W3	優先度

AIソルバーを実行する・その2

AIソルバーはプロジェクト単位で
実行させることができます

Babieca OVMモデラー Ver.1.6.0.0 - C:\Users\dai\OneDrive\Documents\Babieca\Demo_菓子選択問題.xml

ファイル 表示 設定 ヘルプ

Model Browser

- DefaultPacka
 - Diagram
 - Diagram
 - Diagram
 - Diagram
 - お菓子
 - 飲み物
 - うまい棒
 - お茶
 - キャンディ

展開

- パッケージを作成する
- 属性項目を作成する
- テーブル・ビューを表示する
- AIソルバーを実行する**
- 貼り付け
- プロジェクトを開く

①「プロジェクト」を右クリックして
「AIソルバーを実行する」を選択

Diagram_Main

VP

[1,5]

選択>

選択>

選択>

選択>

クッキー

うまい棒

A 価格

A 満足度

A 優先度

ソルバー

ソルバーの設定と実行 高度な設定 (OVM) 高度な設定 (BOM)

ソルバータイプ ASPソルバー (clingo) ※タイムアウト (秒/0は無制限) 30 ※解の最大数 (0は無制限) 100

選択肢の属性

変数	項目名
W1	価格
W2	満足度

制約 (個別値)

変数	項目名	最小値	最大値
*			

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	最小値	最大値
*		

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	タイプ
*	

関係の属性

変数	項目名
W3	優先度

制約 (個別値)

変数	項目名	最小値	最大値
*			

Excel出力: ☒ 横書 ☐ 縦書

閉じる 実行

AIソルバーの設定①実行設定

ソルバー

ソルバーの設定と実行 高度な設定 (OVM) 高度な設定 (BOM)

ソルバータイプ
ASPソルバー (clingo) 【ソルバータイプ】

※タイムアウト (秒/0は無制限) 30
※解の最大数 (0は無制限) 100

選択肢の属性

変数	項目名
W1	価格
W2	満足度

制約 (個別値)

変数	項目名	最小値	最大値
*			

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+,*,/,^)が使用可能です。

式	最小値	最大値
*		

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+,*,/,^)が使用可能です。

式	タイプ
*	

関係の属性

変数	項目名
W3	優先度

制約 (個別値)

変数	項目名	最小値	最大値
*			

Excel出力: ☒ 横書 ☐ 縦書

閉じる 実行

【タイムアウト】 解探索時間の上限を指定します。
Cloudのソルバーサービスを利用する場合は20秒です。

【解の最大数】 解列挙数の上限を指定します。
無制限に探索させることも可能ですが、
結果ウィンドウの表示上限は100個まで、
Excel出力の表示上限は16381個までです。

結果ウィンドウ

実行結果

結果 標準出力

船の数 308718

可変点	バリエーション	船1	船2	船3	船4	船5	船6	船7	船8
カラー	カラー指定	X	X	X	X	X	X	X	X
カラー	サブライヤ指定	X	X	X	X	X	X	X	X
カラー	規格	O	O	O	O	O	O	O	O
コネクタ種類	耐熱コネクタ	X	X	X	X	X	X	X	X
コネクタ種類	耐水コネクタ	O	O	O	O	O	O	O	O
コネクタ種類	耐熱コネクタ	X	X	X	X	X	X	X	X
シールド	シールド有	X	X	X	X	X	X	X	X
シールド	シールド無	O	O	O	O	O	O	O	O
ハーネス構成	コネクタ	O	O	O	O	O	O	O	O
ハーネス構成	固定具	O	O	O	O	O	O	O	O
ハーネス構成	電線	O	O	O	O	O	O	O	O

Excel

Excel

ファイル ホーム 挿入 ページレイアウト 公式 データ 校閲 表示 開発 ヘルプ 操作アシスト

A1

可変点	バリエーション	制約/目的関数	船1	船2	船3	船4	船5	船6	船7	船8
1	カラー	カラー指定	X	X	X	X	X	X	X	X
2	カラー	サブライヤ指定	X	X	X	X	X	X	X	X
3	カラー	規格	O	O	O	O	O	O	O	O
4	コネクタ種類	耐熱コネクタ	X	X	X	X	X	X	X	X
5	コネクタ種類	耐水コネクタ	O	O	O	O	O	O	O	O
6	コネクタ種類	耐熱コネクタ	X	X	X	X	X	X	X	X
7	シールド	シールド有	X	X	X	X	X	X	X	X
8	シールド	シールド無	O	O	O	O	O	O	O	O
9	ハーネス構成	コネクタ	O	O	O	O	O	O	O	O
10	ハーネス構成	固定具	O	O	O	O	O	O	O	O
11	ハーネス構成	電線	O	O	O	O	O	O	O	O
12	ハーネス性能	保護具	O	O	O	O	O	O	O	O
13	ハーネス性能	印加電圧	X	X	X	X	X	X	X	X
14	ハーネス性能	周波数特性	O	O	O	O	O	O	O	O

【Excel出力】
解をExcelに出力します。
横書（解を横に並べる）と縦書（解を縦に並べる）が選べます。

AIソルバーの設定②制約設定（個別値）

選択肢や関係が持つ個々の属性値に制約を定義することができます

【選択肢の制約（個別値）】

属性値が指定値以上/以下の「選択肢」のみ有効とする

- ・左記例では「価格が0円～100円の菓子だけを選ぶことができる」ことになります。

【関係の制約（個別値）】

属性値が指定値以上/以下の「関係」のみ有効とする

- ・左記例では「優先度が0,1,2の関係」だけが有効となり、それ以外（3以上）の関係は無視されます。

ソルバー

ソルバーの設定と実行 高度な設定 (OVM) 高度な設定 (BOM)

ソルバータイプ ASPソルバー (clingo) ※タイムアウト (秒/0は無制限) 30 ※解の最大数 (0は無制限) 100

選択肢の属性

変数	項目名
W1	価格
W2	満足度

【変数】属性を「英数字」で置き換えています

制約 (個別値)

変数	項目名	最小値	最大値
W1	価格	0	100

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	最小値	最大値
W1	200	300

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	タイプ
W2	最大化

関係の属性

変数	項目名
W3	優先度

制約 (個別値)

変数	項目名	最小値	最大値
W3	優先度	0	2

Excel出力: ☒ 横書 ☐ 縦書

閉じる 実行

AIソルバーの設定③制約設定（合計値）

選択枝の属性の**合計値**に制約を定義することができます

The screenshot shows the 'AI Solver' window with the 'Constraint Settings' tab active. It is configured for 'ASP Solver (clingo)' with a 30-second timeout and a maximum of 100 solutions. The 'Selection Option Attributes' section on the left lists variables W1 (Price) and W2 (Satisfaction). The 'Constraints (Total Value)' table on the right defines a constraint for W1 (Price) with a minimum value of 200 and a maximum value of 300. The 'Objective Function' section shows W2 (Satisfaction) set to be maximized. The 'Relationship Attributes' section at the bottom lists variable W3 (Priority).

変数	項目名
W1	価格
W2	満足度

変数	項目名	最小値	最大値
W1	価格	0	100
制約（合計値）... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。			
式	最小値	最大値	
W1	200	300	

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	タイプ
W2	最大化

変数	項目名
W3	優先度

変数	項目名	最小値	最大値
W3	優先度	0	2

Excel出力: ☒ 横書 ☐ 縦書

【選択枝の制約（合計値）】

選択枝の属性値の合計が指定値以上/以下の組み合わせを求める

- ・ 左記例では「価格の合計が200円～300円」になるような組み合わせを求めることになります。

式には、四則演算が使用可能です
(例)

- ・ 「価格」の合計 → "W1"
- ・ 「価格」あたりの「満足度」 → "W2/W1"

AIソルバーは「実数」をサポートしておりませんので、小数点以下の値は全て切り捨てられます。
10を乗ずる(*10)等、適宜調整してください。

AIソルバーの設定④制約設定（目的関数）

選択肢の属性の合計値に目的関数（最大/最小）を定義することができます

The screenshot shows the 'AI Solver' window with the '高度な設定 (BOM)' tab selected. The 'Solver Type' is set to 'ASP Solver (clingo)'. The 'Timeout' is 30 seconds and the 'Maximum Number of Solutions' is 100.

選択肢の属性 (Alternative Attributes):

変数 (Variable)	項目名 (Item Name)
W1	価格 (Price)
W2	満足度 (Satisfaction)

制約 (個別値) (Constraints (Individual Values)):

変数 (Variable)	項目名 (Item Name)	最小値 (Min Value)	最大値 (Max Value)
W1	価格	0	100

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式 (Formula)	最小値 (Min Value)	最大値 (Max Value)
W1	200	300

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式 (Formula)	タイプ (Type)
W2	最大化 (Maximize)

関係の属性 (Relationship Attributes):

変数 (Variable)	項目名 (Item Name)
W3	優先度 (Priority)

制約 (個別値) (Constraints (Individual Values)):

変数 (Variable)	項目名 (Item Name)	最小値 (Min Value)	最大値 (Max Value)
W3	優先度	0	2

Excel出力: ☒ 横書 ☐ 縦書 [閉じる] [実行]

【目的関数の優先度】

目的関数が複数ある場合、
上にあるものほど優先度:高とします

【選択肢の目的関数】

選択肢の属性値の合計が最大/最小となるような組み合わせを求める

- ・ 左記例では「満足度の合計が最大」になるような組み合わせを求める」こととなります。

AIソルバーの実行例①

AIソルバーのインターフェース。変数として「価格」(W1)と「満足度」(W2)が設定されている。制約と目的関数の設定欄も表示されている。右下の「実行」ボタンが押されている状態。

可能な組み合わせは？



制約および目的関数が何も設定されていない場合、
「全ての組み合わせ」が列挙されます



実行結果

結果標準出力

解の数28

可変点	選択肢	番号	属性	解1	解2	解3	解4	解5	解6	解7	解8	解9	解10	解11	解12	解13	解14	解15	解16	解17	解18	解19	解20	解21	解22	解23	解24	解25	解26	解27	解28
[可変点]飲み物	[選択肢]お茶			×	×	×	×	○	○	○	○	○	×	×	×	×	×	×	×	○	○	○	○	○	○	○	○	×	○	○	
[可変点]飲み物	[選択肢]コーラ			×	○	×	○	×	×	×	×	×	×	×	×	×	○	×	×	×	×	×	×	×	×	×	×	○	×	×	
[可変点]飲み物	[選択肢]ジュース			○	×	○	×	×	×	×	×	×	○	○	○	○	×	○	○	×	×	×	×	×	×	×	×	×	×	×	
[可変点]お菓子	[選択肢]クッキー			○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
[可変点]お菓子	[選択肢]うまい棒			×	×	×	×	×	○	×	○	×	×	×	○	○	○	○	○	×	○	×	○	○	×	○	×	○	○	○	○
[可変点]お菓子	[選択肢]キャンディ			×	×	○	○	×	×	○	○	×	×	○	×	○	×	×	○	×	×	×	×	×	○	○	○	○	○	○	○
[可変点]お菓子	[選択肢]チョコレート			×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	○	○	○	×	○	○	○	○	×	×	○	×
[可変点]お菓子	[選択肢]ポテトチップス			×	×	×	×	○	○	○	○	×	○	○	○	○	×	×	×	×	×	○	×	○	×	×	○	×	×	○	×
属性値の合計: 価格				110	130	130	150	200	230	220	250	80	230	250	260	280	160	140	160	180	210	300	110	330	200	230	320	100	180	350	130
属性値の合計: 満足度				9	14	12	17	13	20	16	23	5	17	20	24	27	21	16	19	15	22	23	12	30	18	25	26	8	24	33	15

属性値の合計

AIソルバーの実行例②

価格の合計が200円以上300円以下となる組み合わせは？

目的関数が何も設定されていない場合、
「制約を満たす全ての組み合わせ」が列挙されます

AIソルバーの画面。制約（合計値）... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	最小値	最大値
W1	200	300

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

変数 項目名

変数	項目名
W1	価格
W2	満足度

制約 (個別値)

変数	項目名	最小値	最大値
*			

関係の属性

変数	項目名	制約 (個別値)
*		

Excel出力: ☐ 横書 ☒ 縦書

実行結果

結果 標準出力

解の数 12

可変点	選択枝	番号	属性	解1	解2	解3	解4	解5	解6	解7	解8	解9	解10	解11	解12
[可変点]飲み物	[選択枝]お茶			○	○	○	○	×	○	×	×	○	○	○	×
[可変点]飲み物	[選択枝]コーラ			×	×	×	×	×	×	×	×	×	×	×	×
[可変点]飲み物	[選択枝]ジュース			×	×	×	×	○	×	○	○	×	×	×	○
[可変点]お菓子	[選択枝]クッキー			○	○	○	○	○	○	○	○	○	○	○	○
[可変点]お菓子	[選択枝]うまい棒			○	×	○	×	×	×	○	×	×	○	○	○
[可変点]お菓子	[選択枝]キャンディ			○	○	×	×	×	×	×	○	○	×	○	○
[可変点]お菓子	[選択枝]チョコレート			○	○	○	○	×	×	×	×	×	×	×	×
[可変点]お菓子	[選択枝]ポテトチップス			×	×	×	○	○	○	○	○	○	○	○	○
制約式の値: 価格				230	200	210	300	230	200	260	250	220	230	250	280
属性値の合計: 価格				230	200	210	300	230	200	260	250	220	230	250	280
属性値の合計: 満足度				25	18	22	23	17	13	24	20	16	20	23	27

制約式の値

属性値の合計

制約式の値: 価格

属性値の合計: 価格

属性値の合計: 満足度

AIソルバーの実行例③

ソルバー

ソルバーの設定と実行 高度な設定 (OVM) 高度な設定 (BOM)

ソルバータイプ ASPソルバー (clingo) ※タイムアウト (秒/0は無制限) 30 ※解の最大数 (0は無制限) 100

選択枝の属性

変数	項目名
W1	価格
W2	満足度

制約 (個別値)

変数	項目名	最小値	最大値
*			

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	最小値	最大値
W1	200	300

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	タイプ
W2	最大化

関係の属性

変数	項目名

制約 (個別値)

変数	項目名	最小値	最大値
*			

Excel出力: ☐ 横書 ☒ 縦書

閉じる 実行

価格の合計が200円以上300円以下で、かつ「満足度」が最大となる組み合わせは？

実行結果

結果 標準出力

解の数 1

可変点	選択枝	番号	属性	解1
[可変点]飲み物	[選択枝]お茶			×
[可変点]飲み物	[選択枝]コーラ			×
[可変点]飲み物	[選択枝]ジュース			○
[可変点]お菓子	[選択枝]クッキー			○
[可変点]お菓子	[選択枝]うまい棒			○
[可変点]お菓子	[選択枝]キャンディ			○
[可変点]お菓子	[選択枝]チョコレート			×
[可変点]お菓子	[選択枝]ポテトチップス			○

制約式の値	280
目的関数の値	27
属性値の合計: 価格	280
属性値の合計: 満足度	27

制約式の値

目的関数の値

属性値の合計

制約式の値: 価格

目的関数の値: 満足度

属性値の合計: 価格

属性値の合計: 満足度

AIソルバーの実行例④

ソルバー

ソルバーの設定と実行 高度な設定 (OVM) 高度な設定 (BOM)

ソルバータイプ ASPソルバー (clingo) ※タイムアウト (秒/0は無制限) 30 ※解の最大数 (0は無制限) 100

選択肢の属性

変数	項目名
W1	価格
W2	満足度

制約 (個別値)

変数	項目名	最小値	最大値
*			

制約 (合計値) ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	最小値	最大値
W1	200	300

目的関数 ... 変数(W1,W2,...)を用いて定義してください。四則演算(+, -, *, /)が使用可能です。

式	タイプ
W2 * 100 / W1	最大化

AIソルバーは「実数」をサポートしていない (小数点以下の値は切り捨てられる) ため、100を乗じる

関係の属性

変数	項目名	最小値	最大値
*			

Excel出力: ☐ 横書 ☒ 縦書 閉じる 実行

価格の合計が200円以上300円以下で、かつ「価格あたりの満足度」が最大となる組み合わせは？

実行結果

結果 標準出力

解の数 1

可変点	選択肢	番号	属性	解1
[可変点]飲み物	[選択肢]お茶			×
[可変点]飲み物	[選択肢]コーラ			×
[可変点]飲み物	[選択肢]ジュース			○
[可変点]お菓子	[選択肢]クッキー			○
[可変点]お菓子	[選択肢]うまい棒			○
[可変点]お菓子	[選択肢]キャンディ			○
[可変点]お菓子	[選択肢]チョコレート			×
[可変点]お菓子	[選択肢]ポテトチップス			○
制約式の値: 価格				280
目的関数の値: 満足度 * 100 / 価格				65
属性値の合計: 価格				280
属性値の合計: 満足度				27

制約式の値

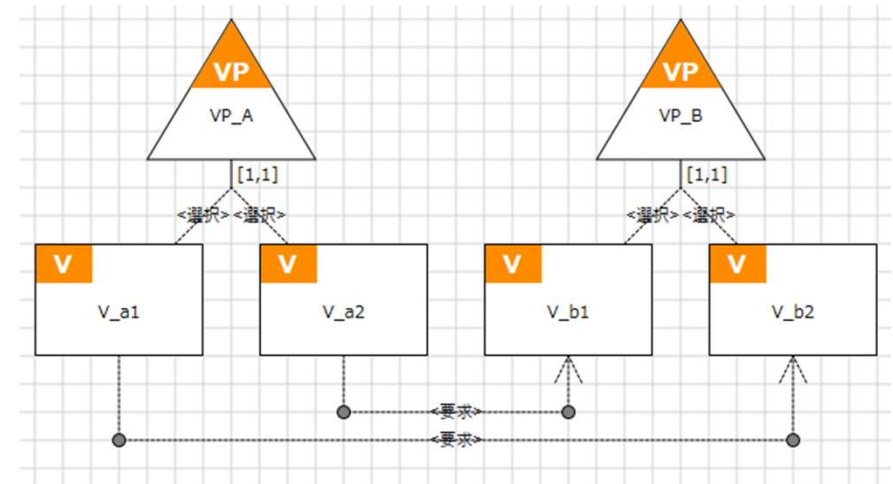
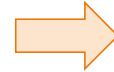
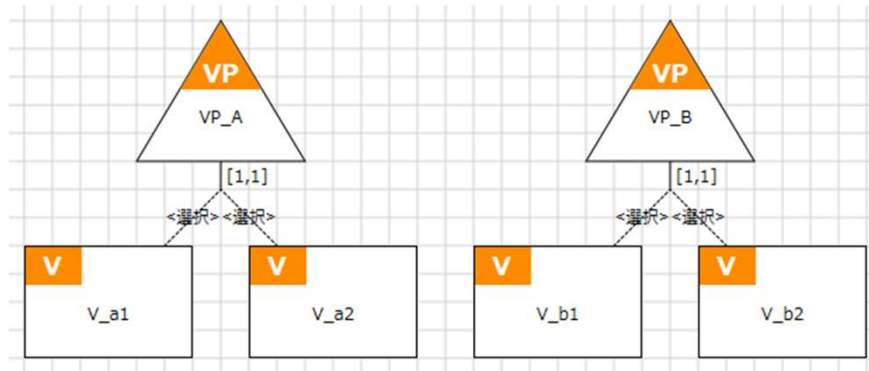
目的関数の値

属性値の合計

(参考) OVMモデリングの定石

OVMモデリングの基本アプローチ： “組み合わせ”を“制約”で絞り込んでゆく

<要求>や<排他>

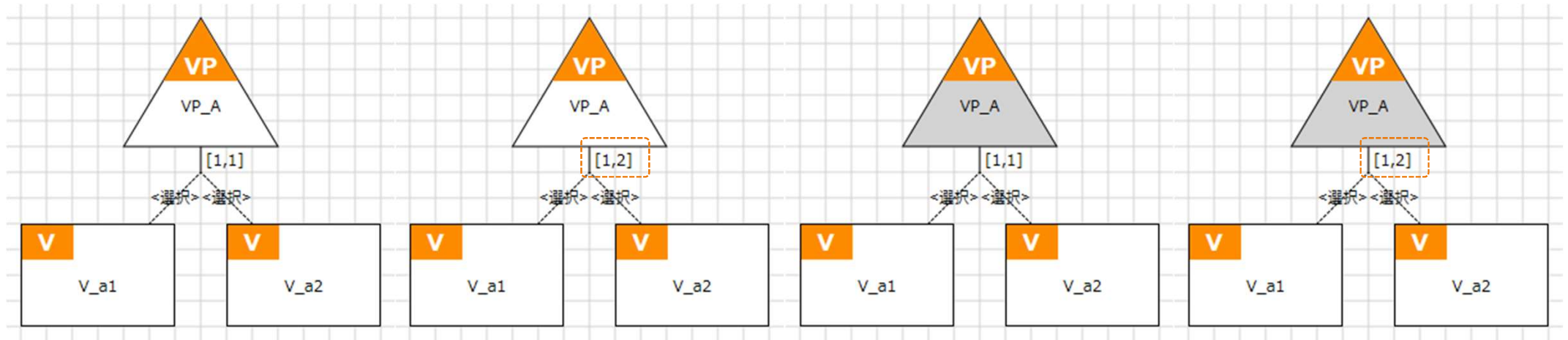


	解1	解2	解3	解4
V_a1	○	×	○	×
V_a2	×	○	×	○
V_b1	×	○	○	×
V_b2	○	×	×	○

	解1	解2		
V_a1	○	×	○	×
V_a2	×	○	×	○
V_b1	×	○	○	×
V_b2	○	×	×	○

“組み合わせ” の定義①

- ・ 可変点の多重度
- ・ 必須な可変点 と 必須でない可変点（その1）



	解1	解2		
V_a1	○	×		
V_a2	×	○		

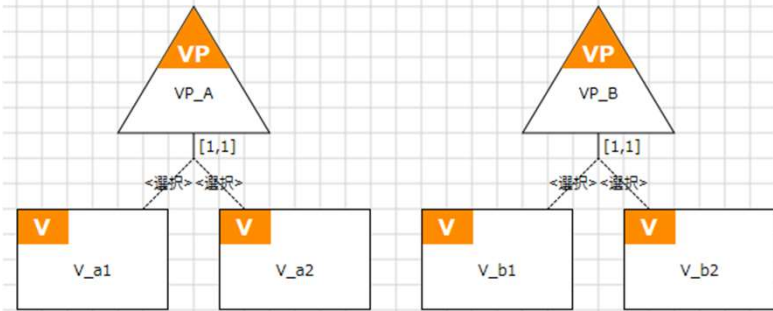
	解1	解2	解3	
V_a1	○	×	○	
V_a2	×	○	○	

	解1	解2	解3	
V_a1	○	×	×	
V_a2	×	○	×	

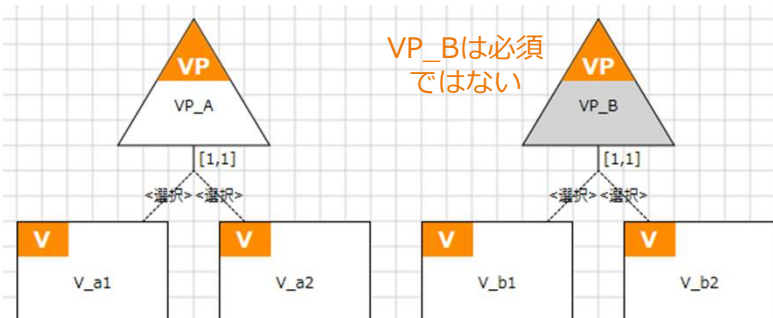
	解1	解2	解3	解4
V_a1	○	×	○	×
V_a2	×	○	○	×

“組み合わせ” の定義②

- ・ 必須な可変点 と 必須でない可変点（その2）

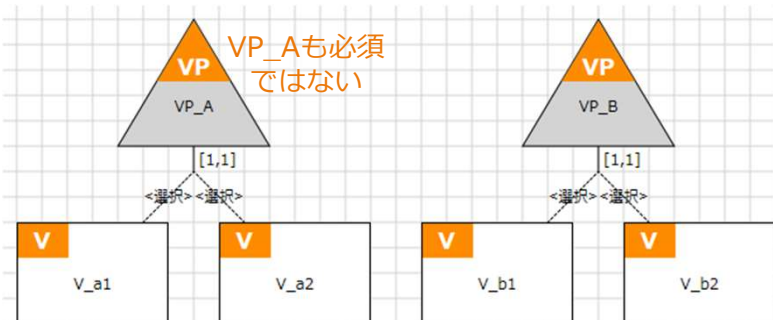


	解1	解2	解3	解4					
V_a1	○	○	×	×					
V_a2	×	×	○	○					
V_b1	○	×	○	×					
V_b2	×	○	×	○					



	解1	解2	解3	解4	解5	解6			
V_a1	○	○	×	×	○	×			
V_a2	×	×	○	○	×	○			
V_b1	○	×	○	×	×	×			
V_b2	×	○	×	○	×	×			

VP_B(V_b1/V_b2)が
選択されない、という
組み合わせが追加



	解1	解2	解3	解4	解5	解6	解7	解8	解9
V_a1	○	○	×	×	○	×	×	×	×
V_a2	×	×	○	○	×	○	×	×	×
V_b1	○	×	○	×	×	×	○	×	×
V_b2	×	○	×	○	×	×	×	○	×

VP_A(V_a1/V_a2)が
選択されない、という
組み合わせが追加

“制約” の定義①

・ 制約の考え方について

OVMモデラーが装備するソルバーは、基本的に「解に含まれない条件」を制約として解の探索を行う。
つまり、

ある「条件に合致する解」を探索して解とするのではなく、

ある「条件に合致しない解」を排除し、残ったものを解とする。

("for ALL"、若しくは、"ブラックリスト方式")

従って、制約を増やす程、解空間が小さくなり、探索が高速になる。

例：BはAを要求する

→ Aが無くてBが有る (A:×, B:○)、という解は無い



#	A	B	解
1	○	○	○
2	○	×	○
3	×	○	×
4	×	×	○

←この解が制約によって排除される

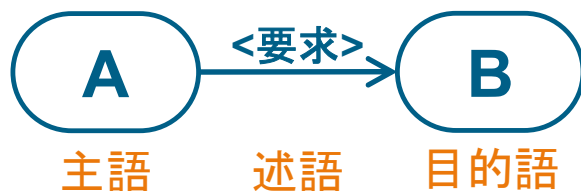
“制約” の定義①

・ 制約の方向について

※トリプルのRDFグラフ表現に準拠

- ・ トリプル：関係情報を表現するための要素。主語、述語、目的語の3つ。
- ・ RDF：Resource Description Framework (W3C)

“AはBを要求する”



矢印元が主語、矢印先が目的語

“AとBは排他である”



“AはBを排除する”



“BはAを排除する”



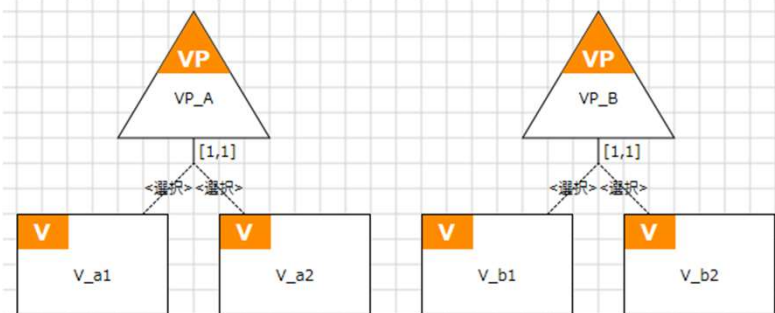
AND

“制約” の定義②

・ 選択肢 に対する<要求>

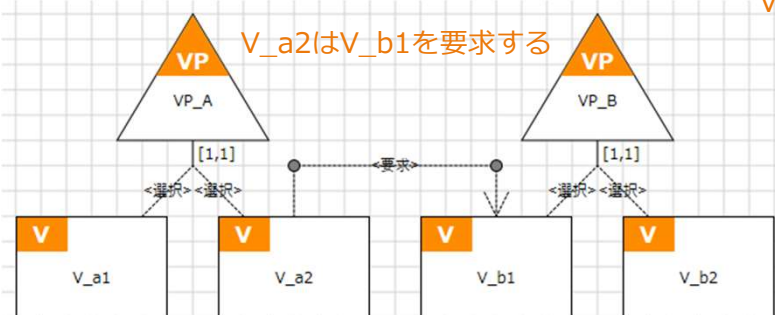


解を追加するのではなく、
解を**排除**してゆく



	解1	解2	解3	解4
V_a1	○	×	○	×
V_a2	×	○	×	○
V_b1	×	○	○	×
V_b2	○	×	×	○

V_a2が選択されなくても
V_b1が選択されることはある

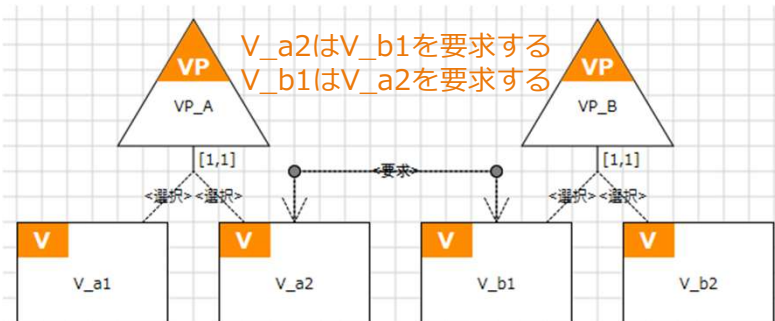


	解1	解2	解3	
V_a1	○	×	○	×
V_a2	×	○	×	○
V_b1	×	○	○	×
V_b2	○	×	×	○

V_a2はV_b1を要求する

↓
V_a2が選択されたらV_b1も選択される

↓
V_a2が○（選択されている）で
V_b1が×（選択されていない）、
という組み合わせは排除



	解1	解2		
V_a1	○	×	○	×
V_a2	×	○	×	○
V_b1	×	○	○	×
V_b2	○	×	×	○

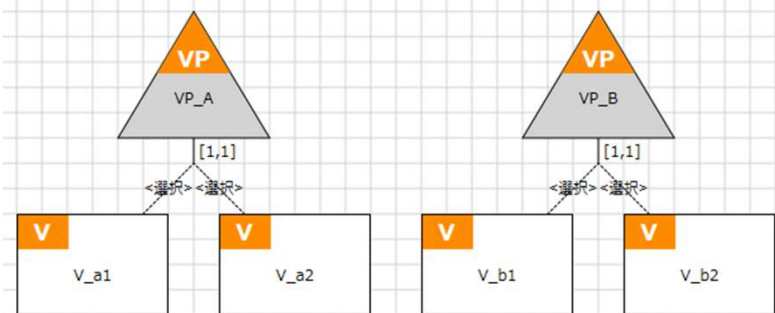
V_a2はV_b1を要求する
V_b1はV_a2を要求する

↓
V_a2が選択されたらV_b1も選択される
V_b1が選択されたらV_a2も選択される

↓
V_a2が○でV_b1が×、
V_b1が○でV_a2が×
という組み合わせは排除

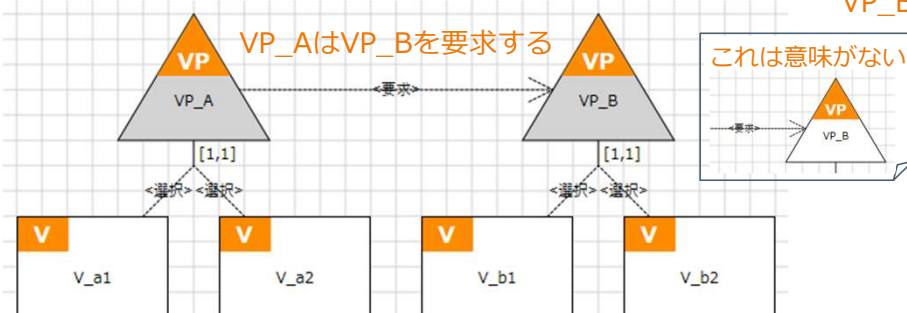
“制約” の定義③

・ 可変点 に対する<要求>



	解1	解2	解3	解4	解5	解6	解7	解8	解9
V_a1	○	○	×	×	×	×	×	○	×
V_a2	×	×	×	×	○	○	×	×	○
V_b1	○	×	○	×	○	×	×	×	×
V_b2	×	○	×	○	×	○	×	×	×

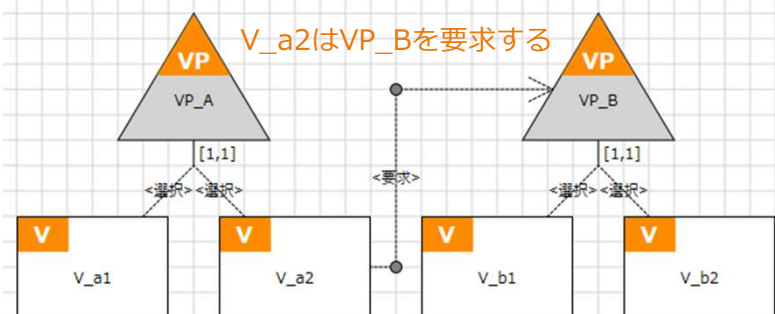
VP_Aが選択されなくても
VP_Bが選択されることはある



	解1	解2	解3	解4	解5	解6	解7		
V_a1	○	○	×	×	×	×	×	○	×
V_a2	×	×	×	×	○	○	×	×	○
V_b1	○	×	○	×	○	×	×	×	×
V_b2	×	○	×	○	×	○	×	×	×

VP_A(V_a1/V_a2)が
選択されているのに
VP_Bが(V_b1/V_b2)が
選択されていない、
という組み合わせを排除

V_a2が選択されなくても
VP_Bが選択されることはある

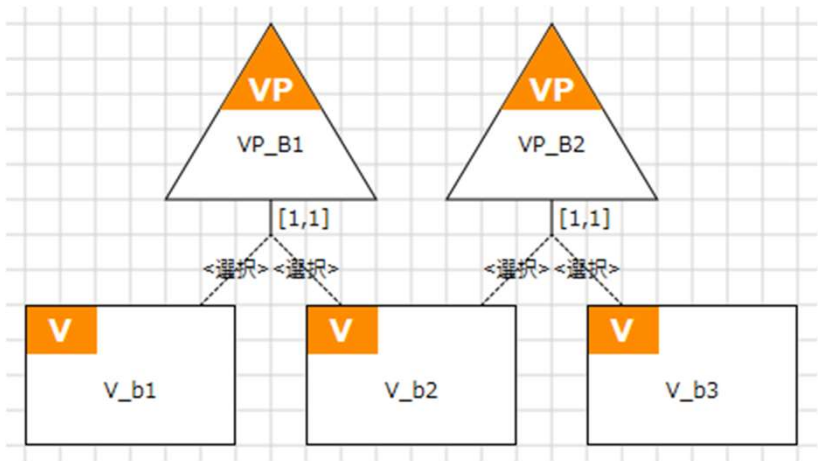


	解1	解2	解3	解4	解5	解6	解7	解8	
V_a1	○	○	×	×	×	×	×	○	×
V_a2	×	×	×	×	○	○	×	×	○
V_b1	○	×	○	×	○	×	×	×	×
V_b2	×	○	×	○	×	○	×	×	×

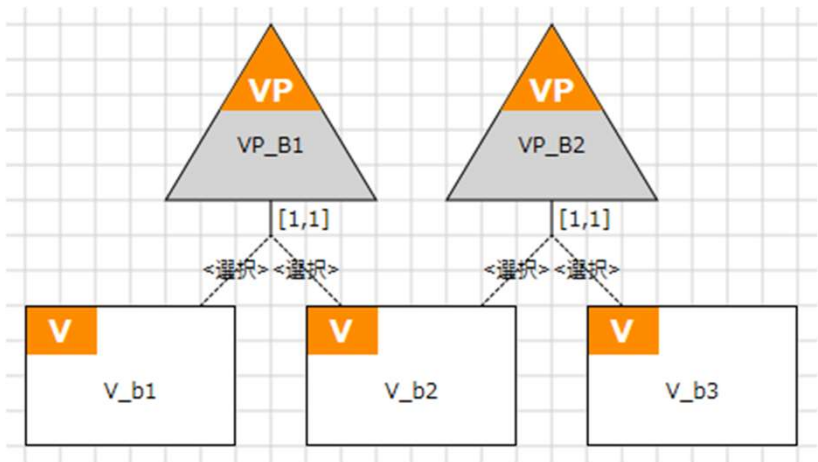
V_a2が
選択されているのに
VP_Bが(V_b1/V_b2)が
選択されていない、とい
う組み合わせを排除

モデリングの定石①

- ・ 選択肢を共有する可変点



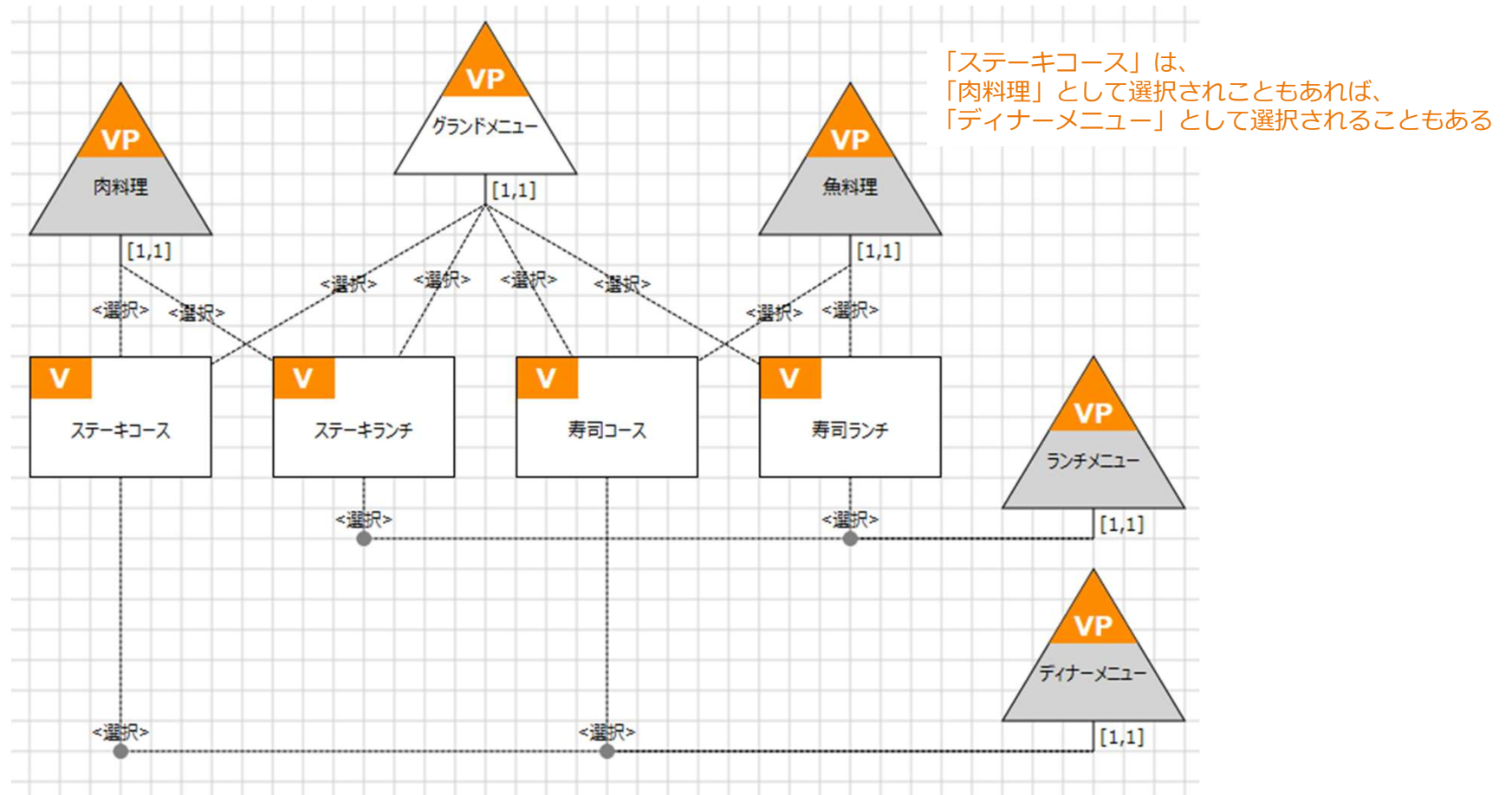
		解1	解2			
VP_B1	V_b1	○	×			
	V_b2	×	○			
VP_B2	V_b2	×	○			
	V_b3	○	×			



		解1	解2	解3	解4	解5	解6	解7
VP_B1	V_b1	○	×	×	×	○	×	×
	V_b2	×	○	○	×	×	×	×
VP_B2	V_b2	×	○	×	○	×	×	×
	V_b3	○	×	×	×	×	○	×

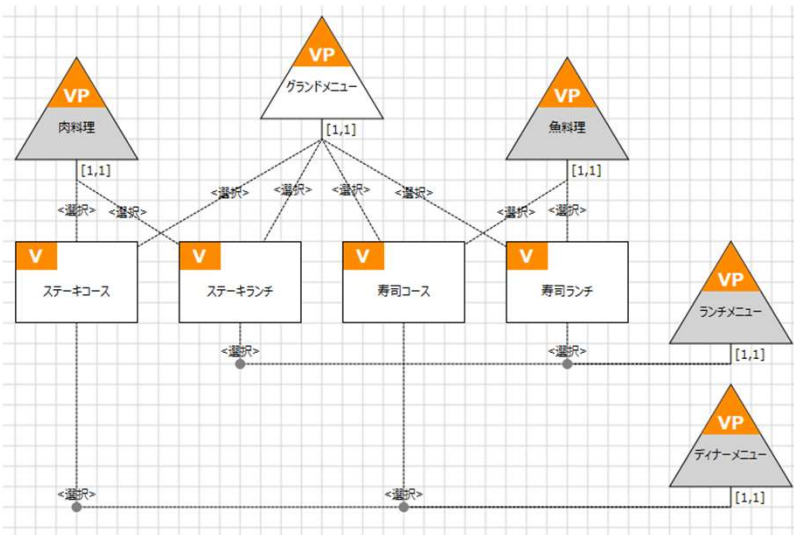
選択肢を共有する可変点のモデル例

OVMでは、選択肢に対して、複数の観点で可変点を設定/選択を行うことができます

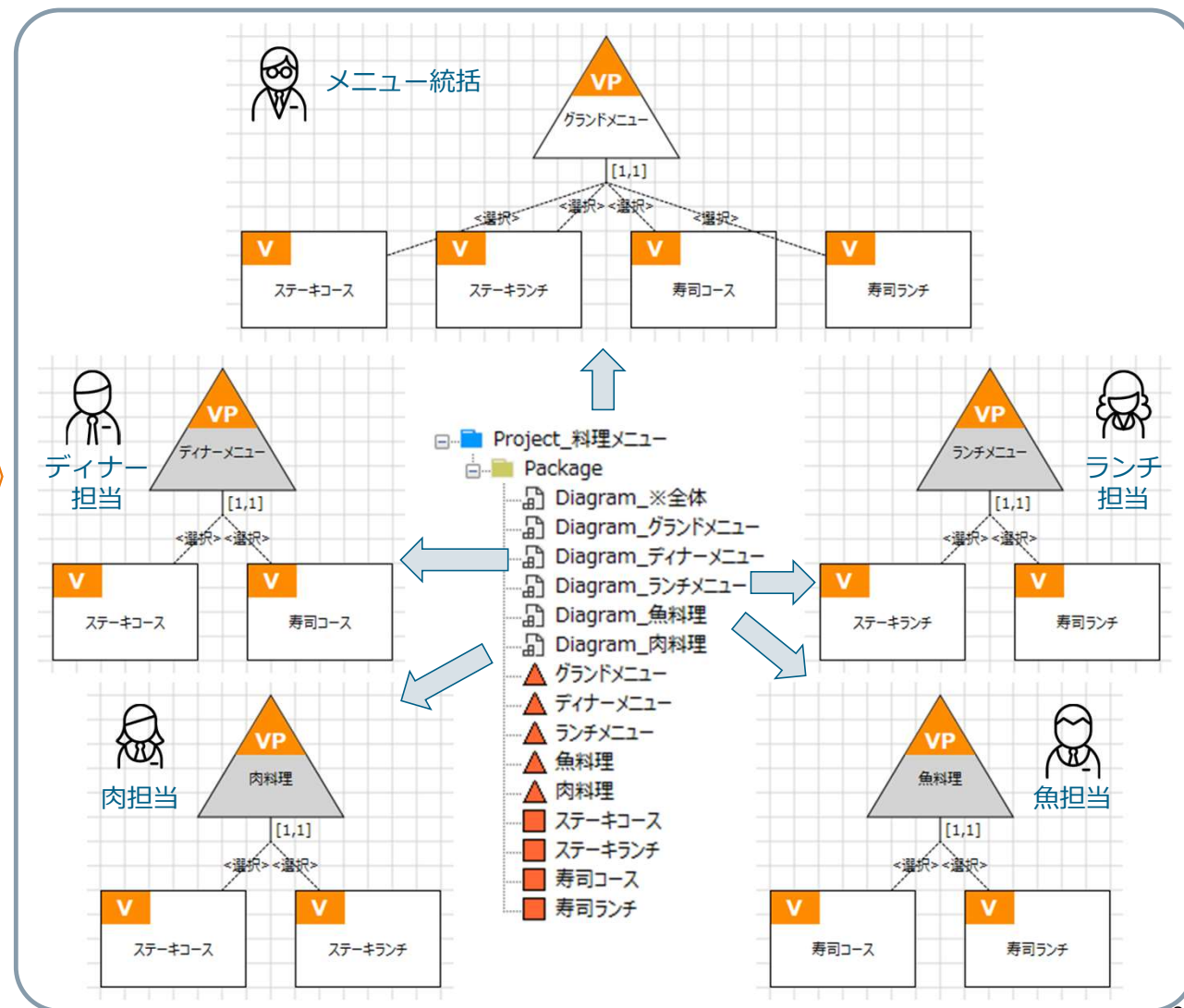


モデリングの定石②ダイアグラムは観点毎に分けて書く

一つのダイアグラムに全部書こうとすると
関係が錯綜して分かり難くなる



一つのダイアグラムに
10~20要素くらいが目安

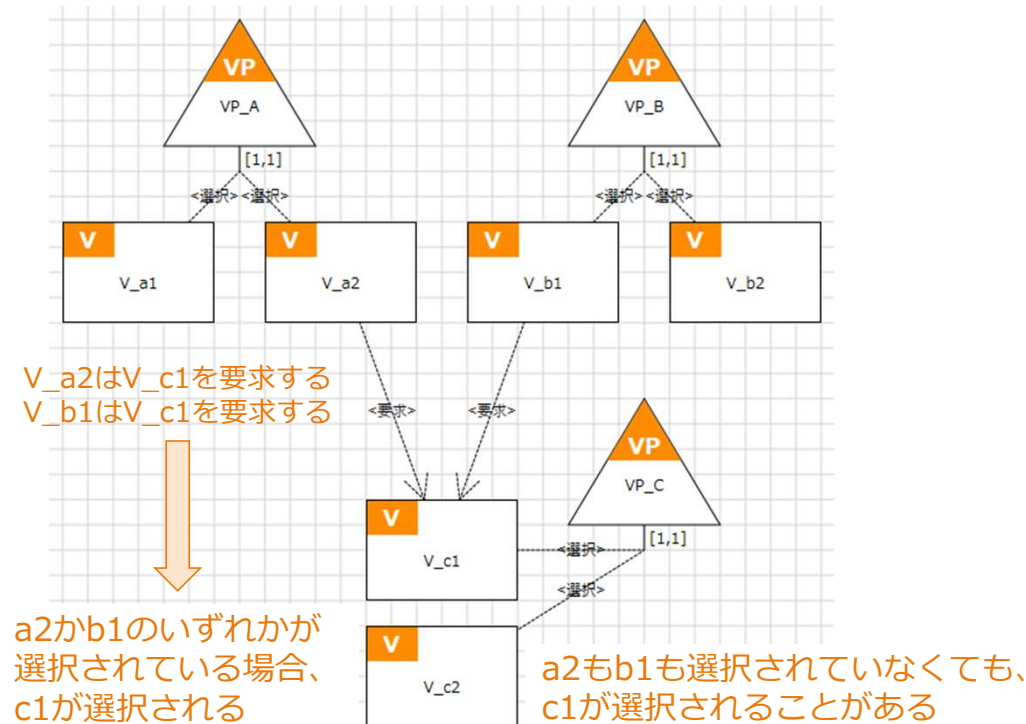


モデリングの定石③

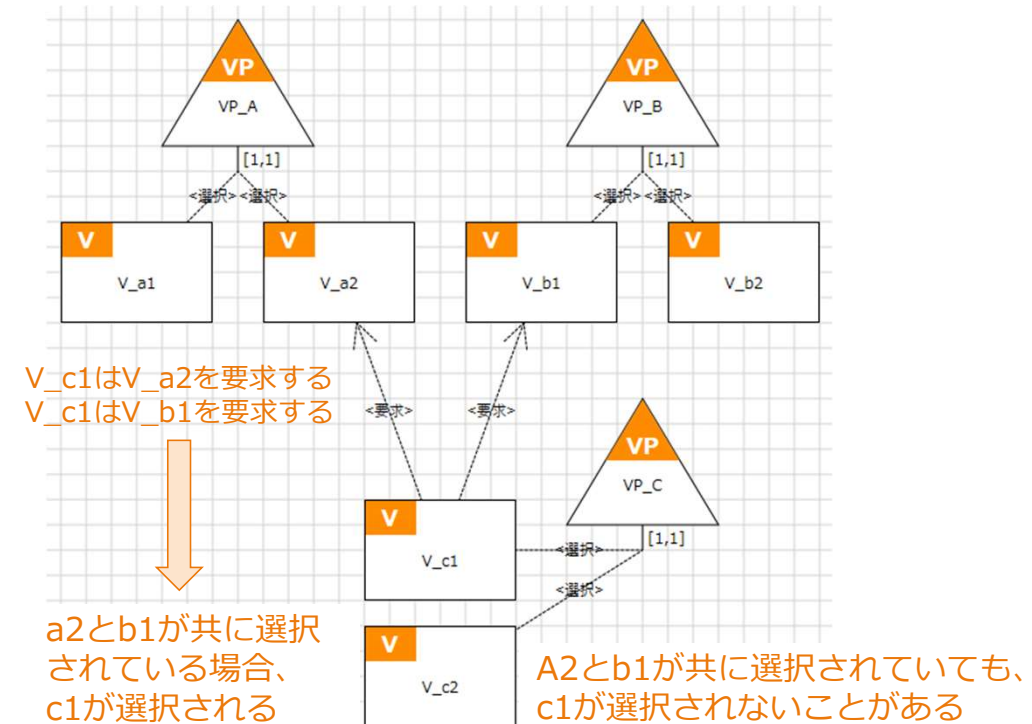
・ 複合条件 … 実は難しい



様々な記述法が考えられるが
定石たりうる記述方法はない

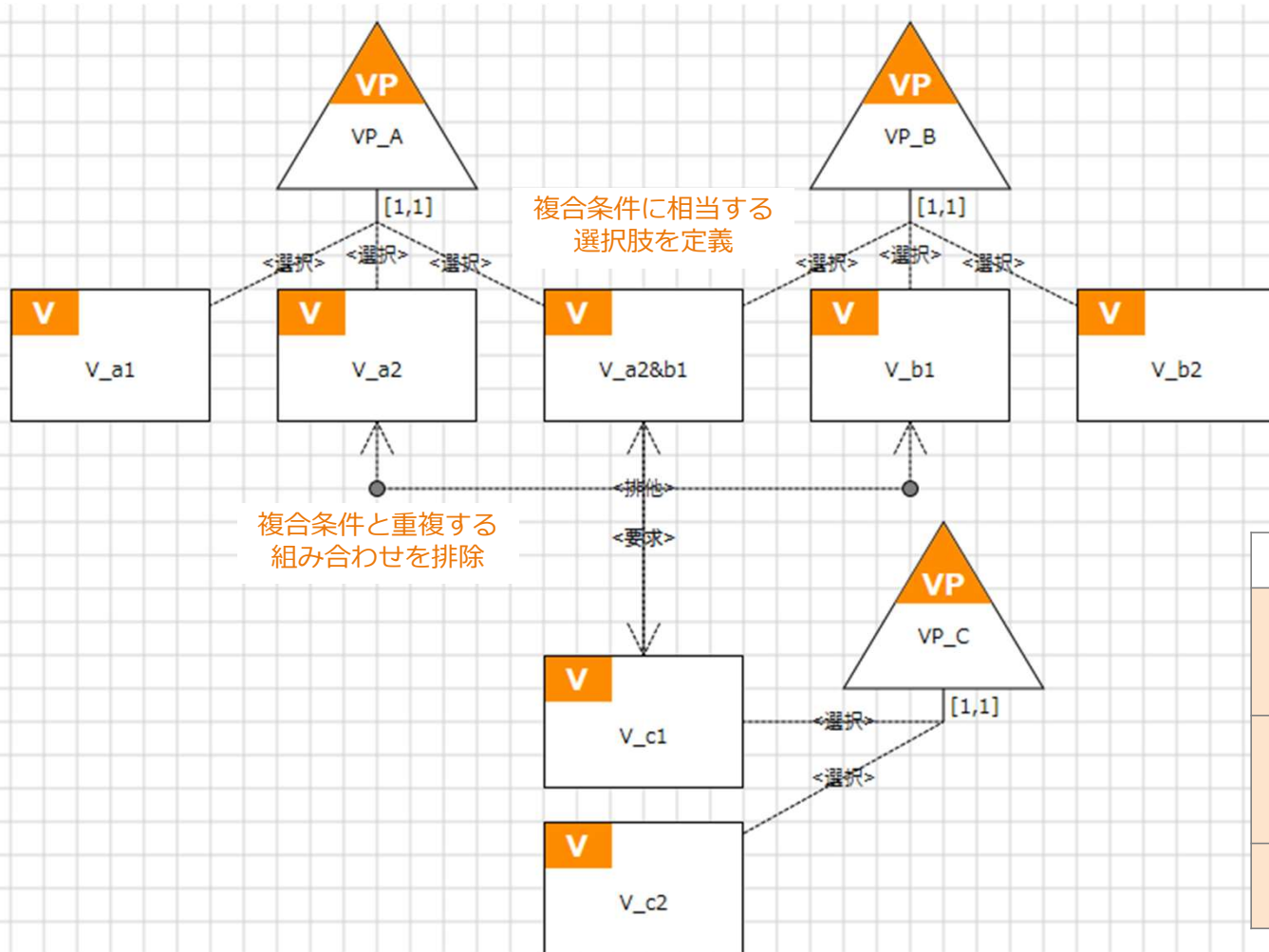


		解1	解2	解3	解4	解5
VP_A	V_a1	○	○	○	×	×
	V_a2	×	×	×	○	○
VP_B	V_b1	○	×	×	○	×
	V_b2	×	○	○	×	○
VP_C	V_c1	○	×	○	○	○
	V_c2	×	○	×	×	×



		解1	解2	解3	解4	解5
VP_A	V_a1	○	○	×	×	×
	V_a2	×	×	○	○	○
VP_B	V_b1	○	×	○	○	×
	V_b2	×	○	×	×	○
VP_C	V_c1	×	×	○	×	×
	V_c2	○	○	×	○	○

(参考) 複合条件のモデル記述例



【注意/欠点】
V_a2&b1はV_a2やV_b1に比して
抽象的かつMECEではないため、
モデルが冗長になり保守性を損なう。

		解1	解2	解3	解4
VP_A	V_a1	○	○	×	×
	V_a2	×	×	○	×
	V_a2&b1	×	×	×	○
VP_B	V_b1	○	×	×	×
	V_b2	×	○	○	×
VP_C	V_c1	×	×	×	○
	V_c2	○	○	○	×

END