

ものづくり・IT 融合化推進技術の研究開発

MZ Platform デバッガ操作説明書

= *Debugger Manual* =

＝目次＝

1. 概要	3
2. 用途	3
3. 用語説明	3
4. デバッガの操作方法	4
4-1.ブレークポイント	4
4-1-1.ブレークポイントの設定	4
4-1-2.ブレークポイントの解除	5
4-2.デバッガの起動	6
4-3.デバッグ画面の説明	7
4-4.開始／再起動	8
4-5.デバッグ終了	9
4-6.再開	10
4-7.ステップイン	11
4-7-1.イベントの発生しないメソッド	11
4-7-2.イベントの発生するメソッド	12
4-7-3.複合コンポーネント	13
4-8.ステップオーバー	14
4-9.ステップアウト	15
4-10.メソッド最後のステップ実行	16
4-11.メソッドコールスタックパネル	17
4-12.メソッド展開	18
4-13.引数と属性パネル	19
4-14.メソッド処理結果リストパネル	20
4-15.イベントキュー画面	21
5. デバッガの制限事項	22

1. 概要

アプリケーションビルダーでは、作成したアプリケーションの不具合(バグ)の発見や修正を支援する機能を提供しています。

アプリケーションに不具合があり、意図しない動作や異常終了が起きた場合、不具合の位置とその原因を調べる必要があります。動作しているアプリケーションの状態を調べたり、挙動に介入したりする機能を総称してデバッグ(「バグを取り除くもの」という意味)と呼びます。

デバッグには、アプリケーションの実行を特定の位置で一時停止する「ブレークポイント」機能や、一つずつ動作を確認しながらメソッド起動を実行する「ステップ実行」機能、実行中の引数や戻り値などの状態を見る「トレース」機能が備えられています。

2. 用途

- ・アプリケーションの実行を指定位置で一時停止する
- ・メソッド起動を一つずつ実行する
- ・実行中の変数の値を参照する

3. 用語説明

1) ブレークポイント

ブレークポイントとは、アプリケーション内の処理の途中につける「しるし」のことで、アプリケーションの実行を一時停止する「場所」を指定するために使用されます。ブレークポイントはメソッド起動に対して設定することができ、設定された場合にはそのメソッド起動が実行される直前で一時停止します。

2) ステップオーバー

ステップオーバーは、一時停止している位置のメソッド起動が完了した直後まで、実行を進めて停止することを指します。

3) ステップイン

ステップインは、一時停止している位置のメソッド起動を実行する際に、その処理中でさらに他のメソッド起動が呼び出されている場合に、その呼び出されるメソッド起動の直前で停止することを指します。このような状況は、メソッド起動によってイベントが発生する場合などに生じます。一方、一時停止している位置のメソッド起動を実行する際に、その処理中で他のメソッド起動が呼び出されていない場合には、ステップオーバーと同様にそのメソッド起動が完了した直後まで、実行を進めて停止します。

4) ステップアウト

ステップアウトは、ステップインで述べたメソッド起動中のメソッド起動が生じている状況で、停止位置のメソッド起動の呼び出し元であるメソッド起動が完了した直後まで、実行を進めて停止することを指します。

5) 再開

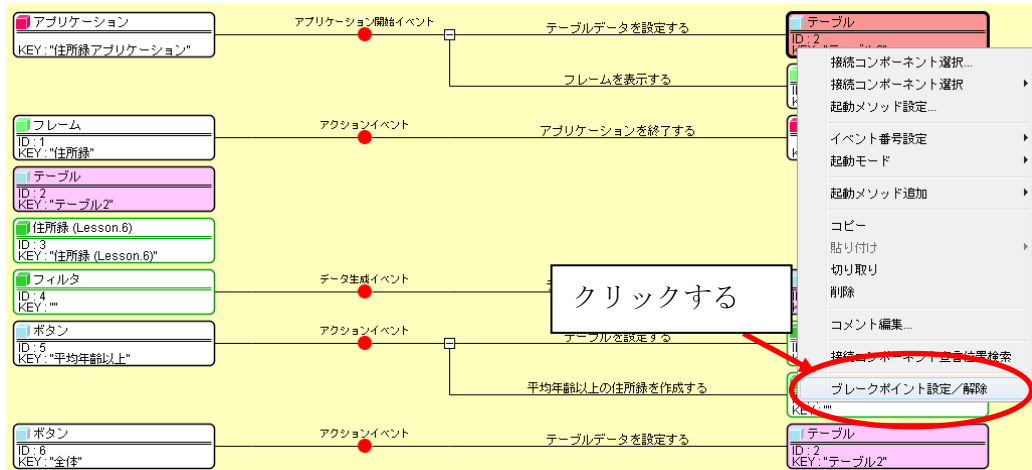
一時停止しているアプリケーションの実行を再開することを指します。再開後の処理にブレークポイントが設定されている場合は次のブレークポイントで一時停止します。

4. デバッガの操作方法

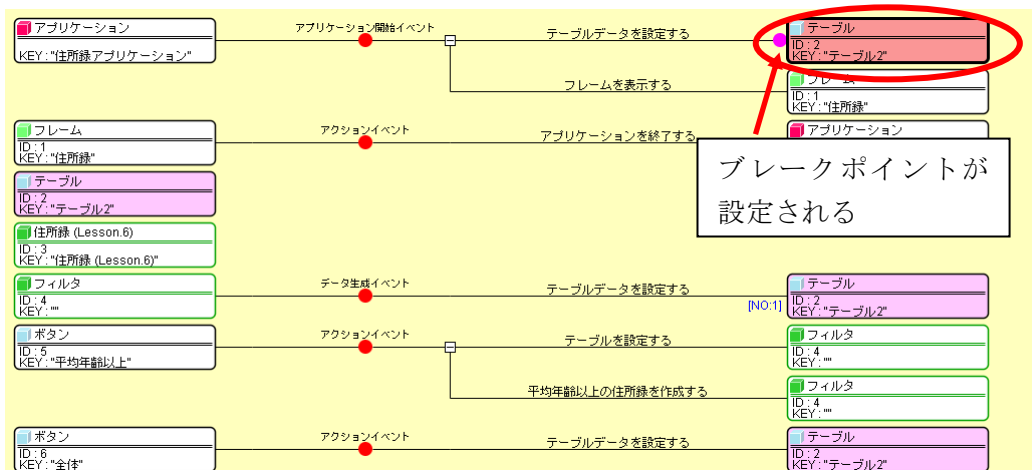
4-1. ブレークポイント

4-1-1. ブレークポイントの設定

- ①処理を一時停止したいメソッド起動の接続先コンポーネント上でマウスを右クリックして、ブレークポイント設定/解除メニューを選択する
- ②接続コンポーネントの横に●が表示される



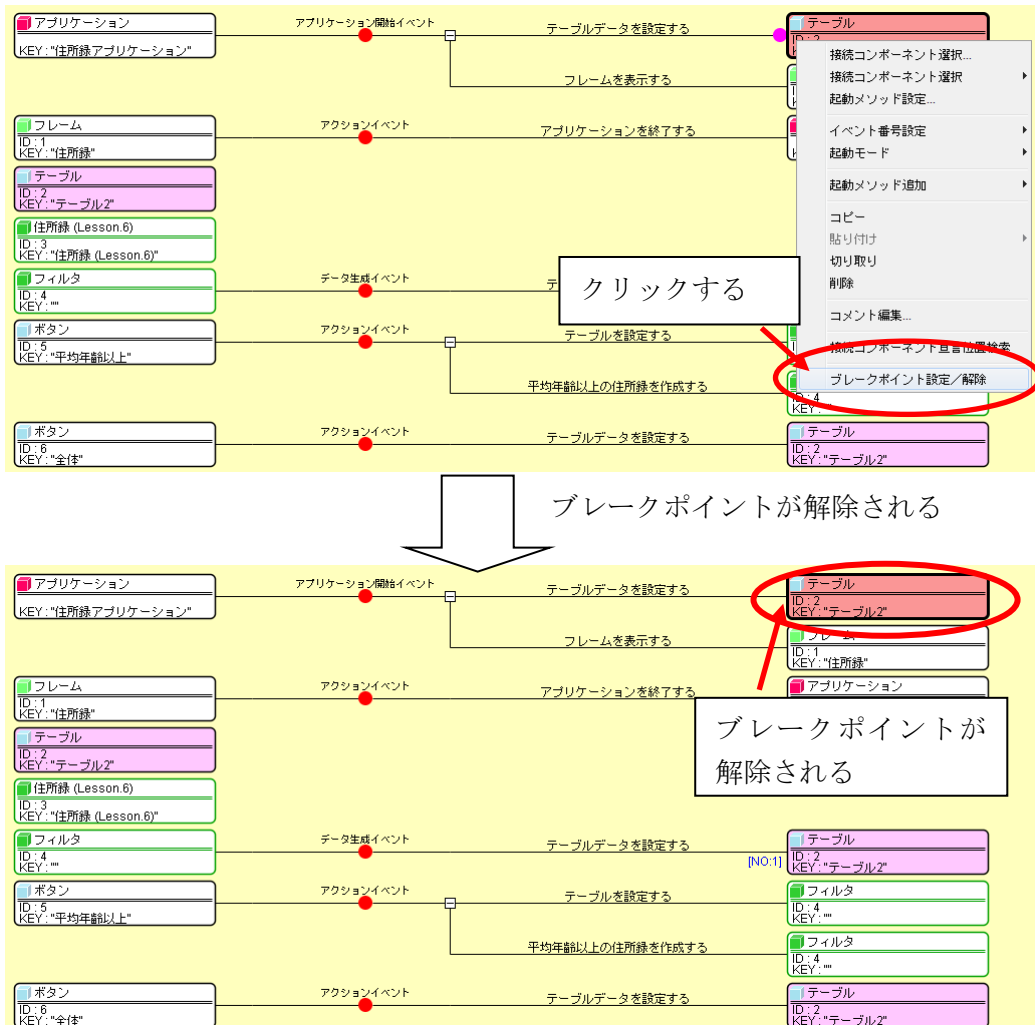
↓
ブレークポイントが
設定される



※ブレークポイントを設定した状態でアプリケーションを保存した場合、ブレークポイントの情報も同様に保存されます。

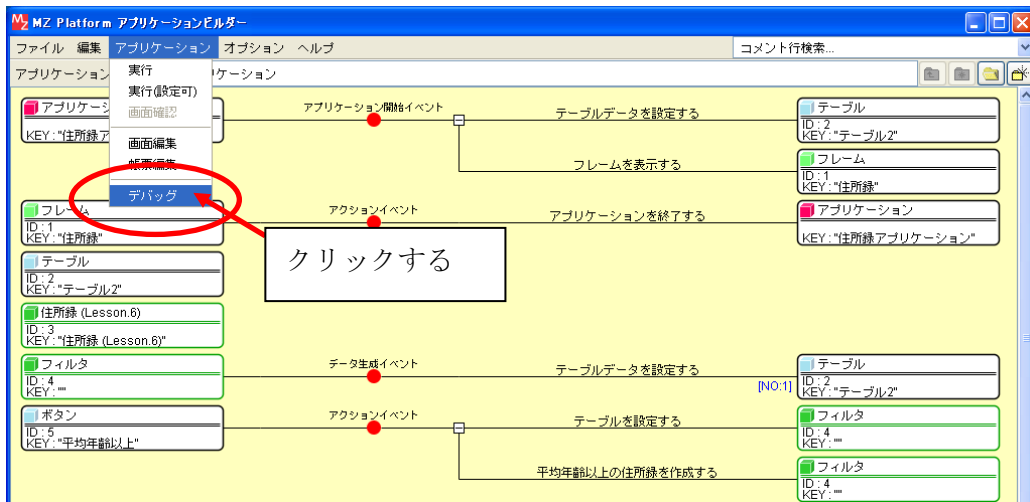
4-1-2. ブレークポイントの解除

- ① ブレークポイントが設定されている起動メソッドの接続先コンポーネント上でマウスを右クリックして、ブレークポイント設定/解除メニューを選択する
- ② 接続コンポーネントの横の●が消去される

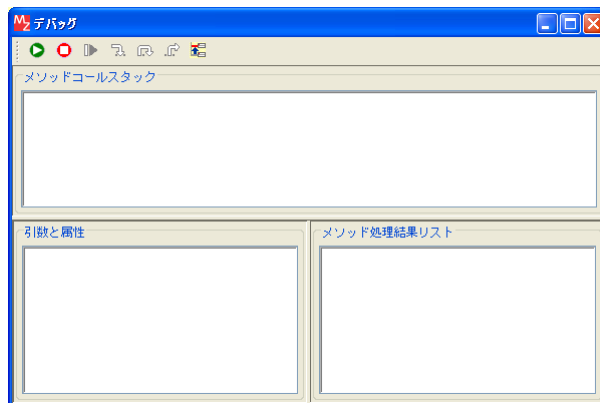


4-2.デバグの起動

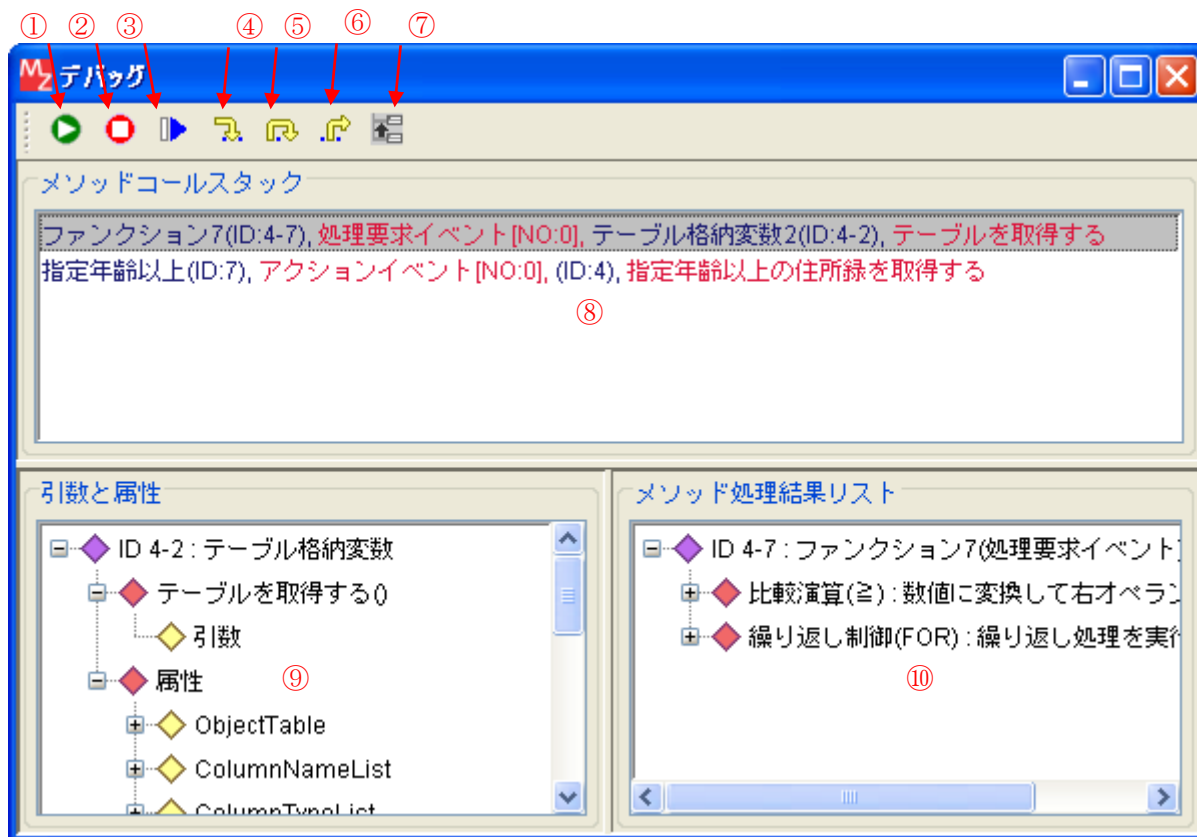
- ①アプリケーションビルダーのメニューバーから【アプリケーション】 - 【デバグ】を選択する
- ②デバグ画面が表示される



↓ デバグ画面が起動される



4-3.デバッグ画面の説明



番号	名称	機能
①	開始／再起動ボタン	デバッグの開始や再起動をおこなう
②	終了ボタン	デバッグを終了する
③	再開ボタン	*1
④	ステップインボタン	*1
⑤	ステップオーバーボタン	*1
⑥	ステップアウトボタン	*1
⑦	メソッド展開ボタン	ブレークポイントが設定されたメソッド起動の引数のうち、メソッド戻り値型の値取得メソッドについて、メソッド処理結果型に変更し展開する（デバッグ開始前・終了時のみ有効）
⑧	メソッドコールスタックパネル	処理が完結していないメソッドをリスト表示する*2
⑨	引数と属性パネル	実行されるメソッドの引数と属性を表示する*3
⑩	メソッド処理結果リストパネル	実行しているイベントのメソッド処理結果を表示する*4

*1 3.用語説明参照

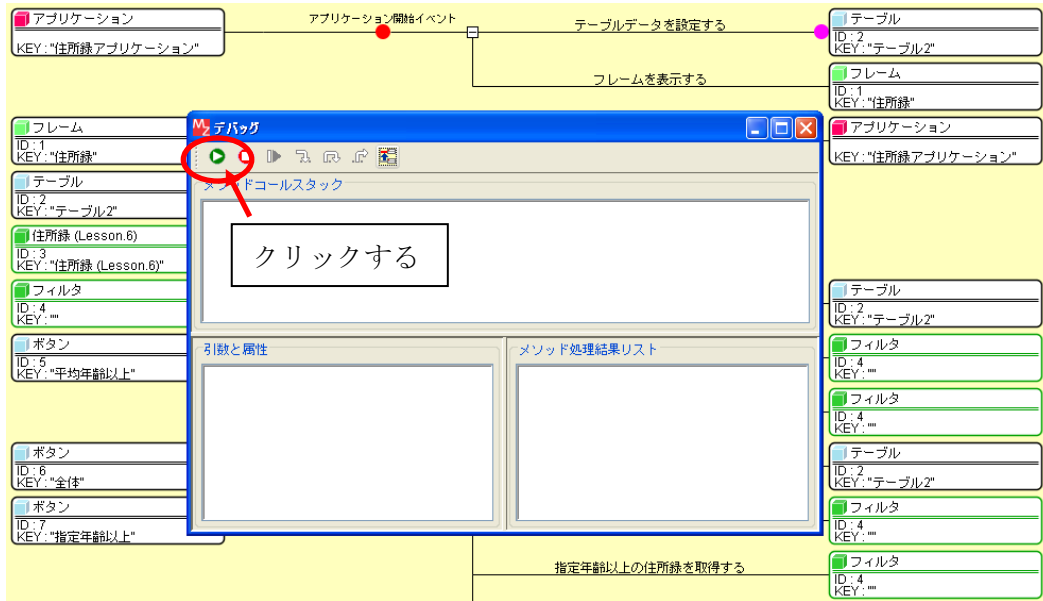
*2 4-11.メソッドコールスタックパネル参照

*3 4-13.引数と属性パネル参照

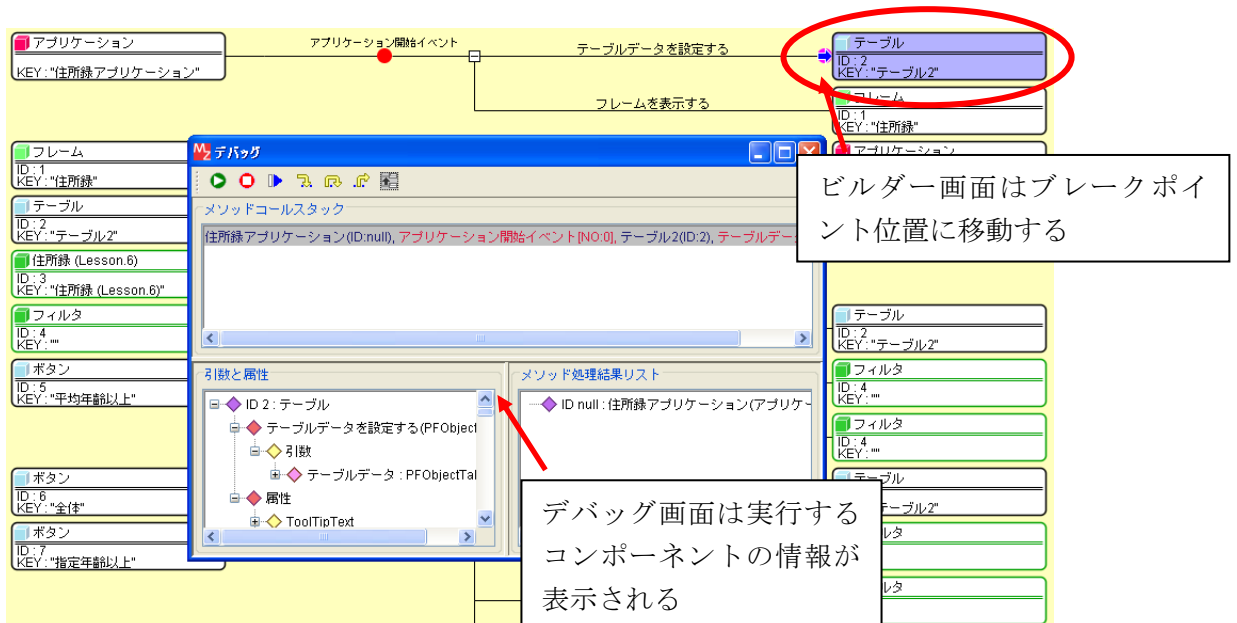
*4 4-14.メソッド処理結果リストパネル参照

4-4.開始／再起動

- ①開始／再起動ボタンを押下する
- ②ブレークポイントが設定されているコンポーネントの情報がデバッグ画面に表示され、ビルダー画面に矢印が表示される



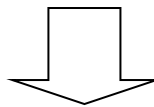
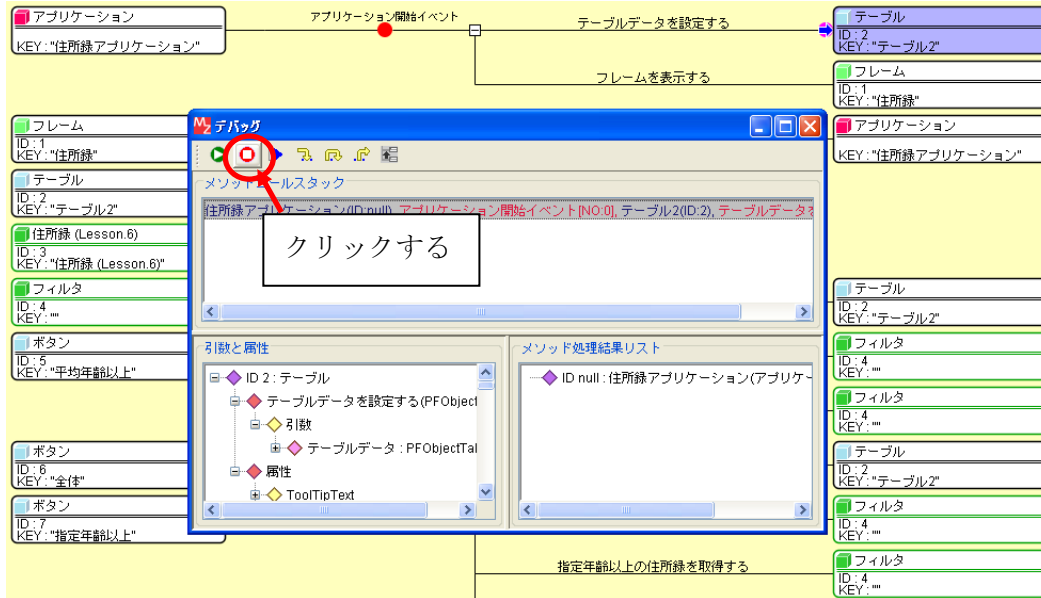
デバッグ画面にメソッド情報が表示される



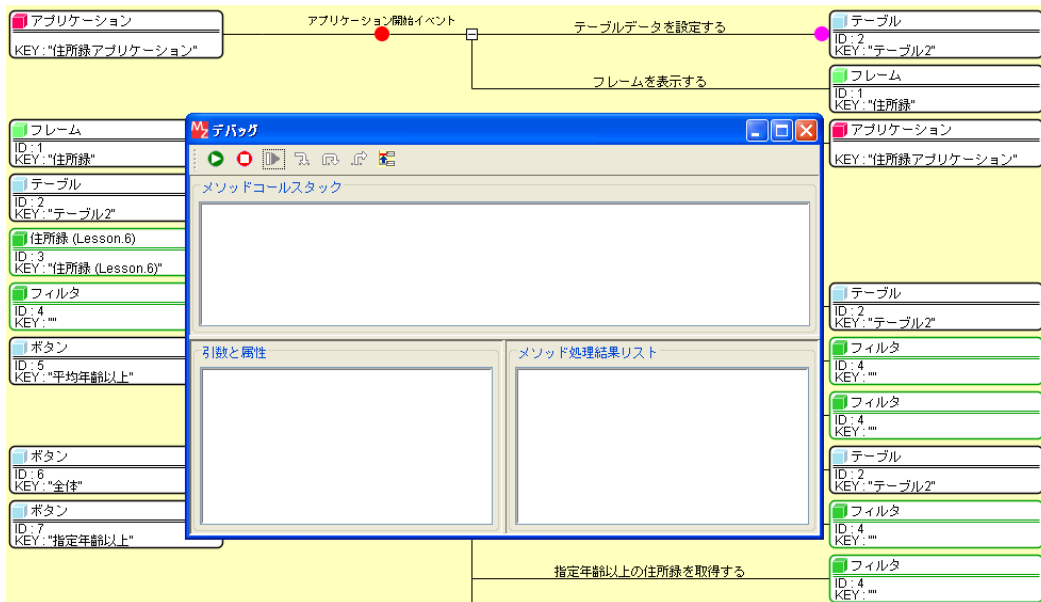
※デバッグ中に開始／再起動ボタンを押下すると、もう一度最初から処理が開始される

4-5.デバッグ終了

- ①終了ボタンを押下する
- ②デバッグ実行が終了する



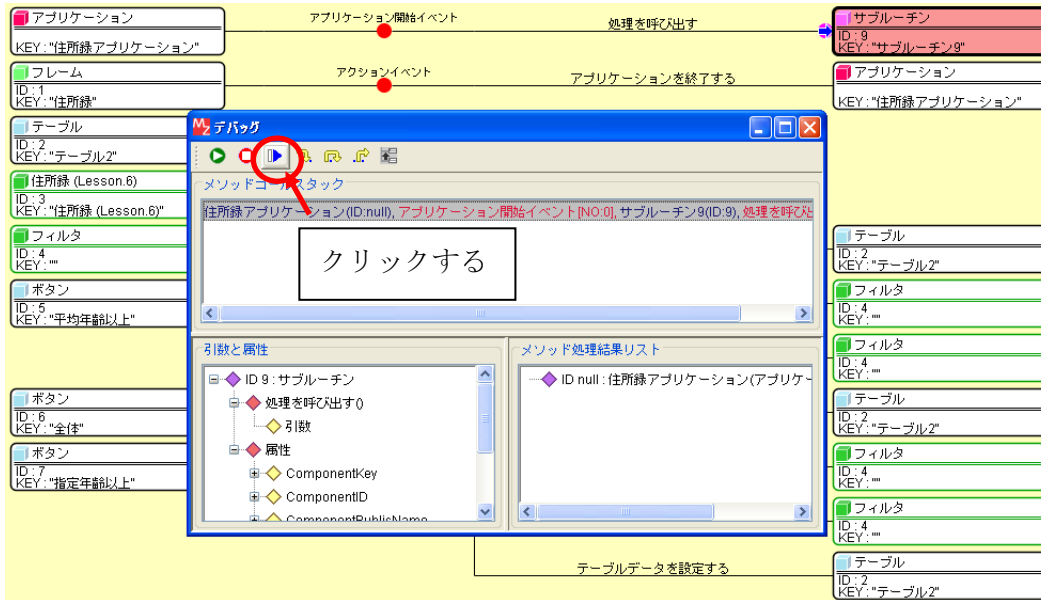
デバッグ情報がクリアされる



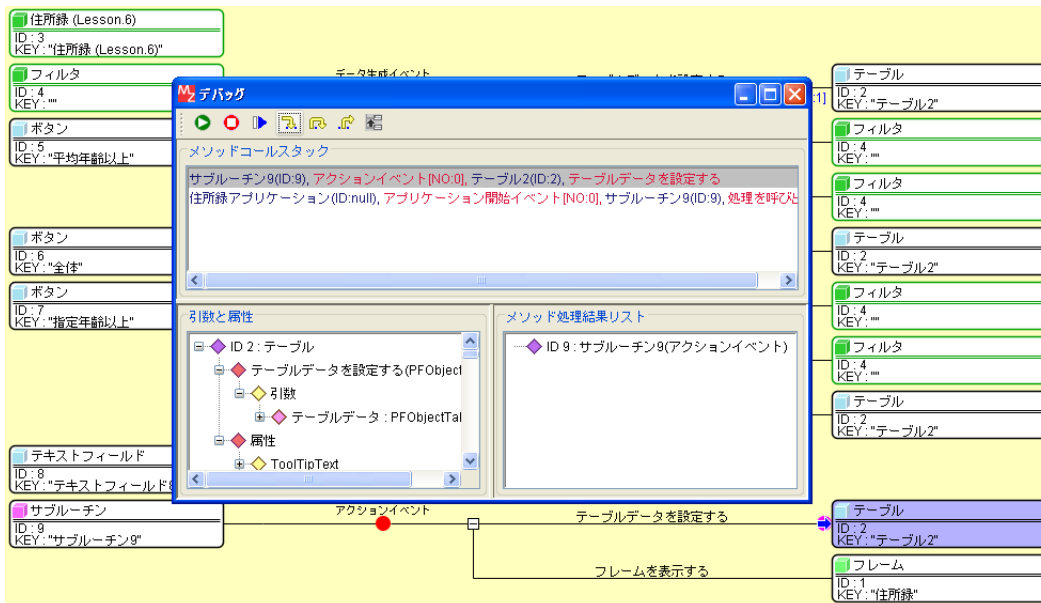
4-6.再開

- ①再開ボタンを押下する
- ②実行しているイベントの次のブレークポイントまで処理を進める

※ 実行しているイベントにブレークポイントが設定されていない場合は、そのイベントの処理が終了します。



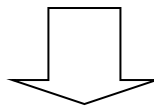
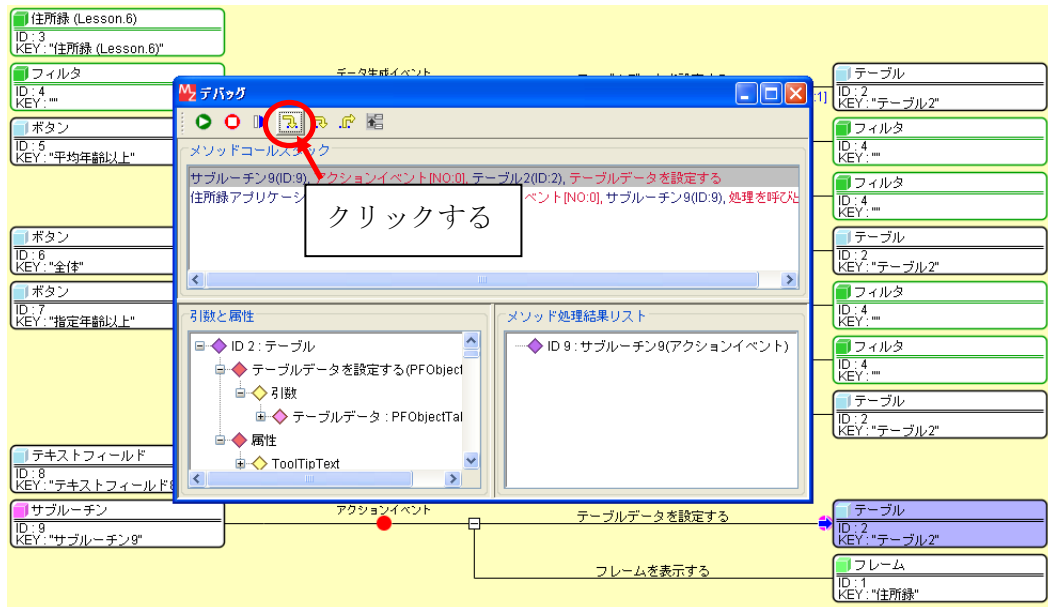
↓ 次のブレークポイントまで処理
が実行される



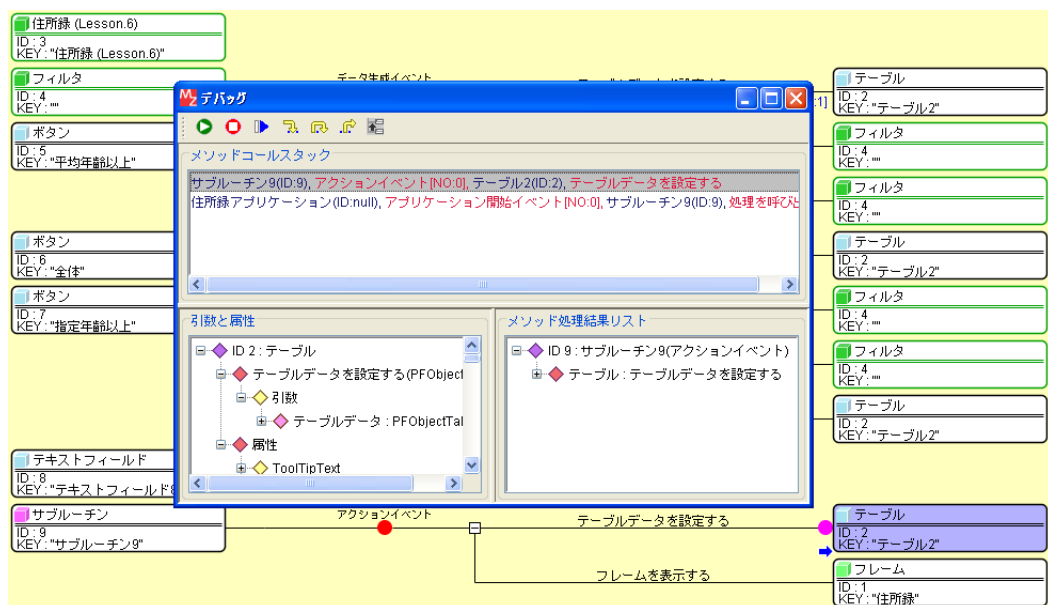
4-7.ステップイン

4-7-1.イベントの発生しないメソッド

- ①ステップインボタンを押下する
- ②メソッド実行後の状態に移動する



メソッド実行後の状態に移動する



4-7-2. イベントの発生するメソッド

- ① ステップインボタンを押下する
- ② 内部から呼び出されたイベントのメソッド処理に移動する

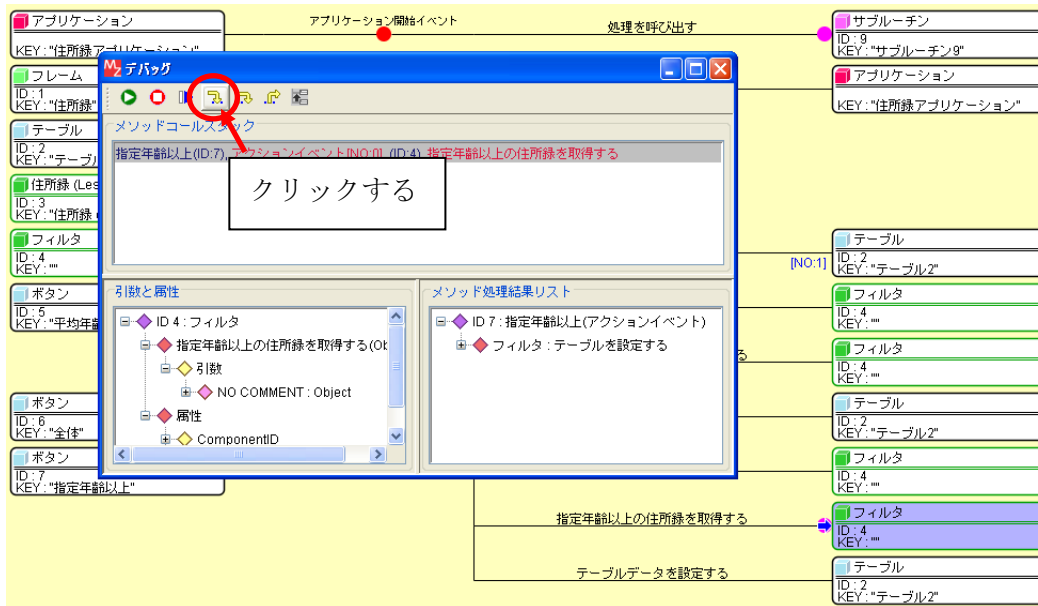


起動メソッドから呼び出された
イベントに処理が移る

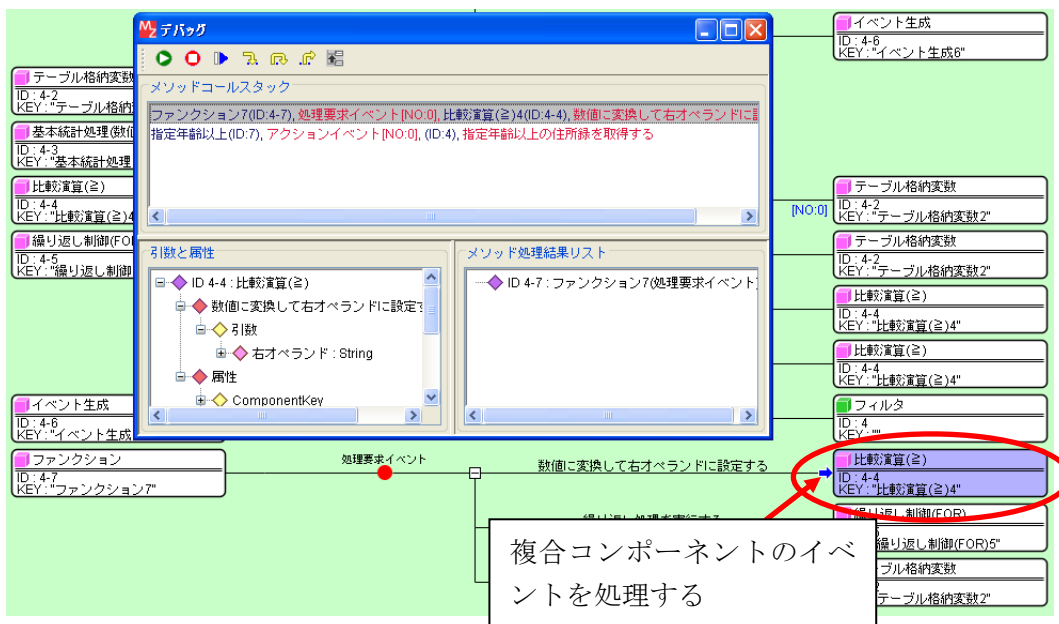


4-7-3.複合コンポーネント

- ①ステップインボタンを押下する
- ②複合コンポーネントの内の処理に移動する



↓
画面が複合コンポーネント
の中に遷移する

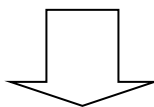
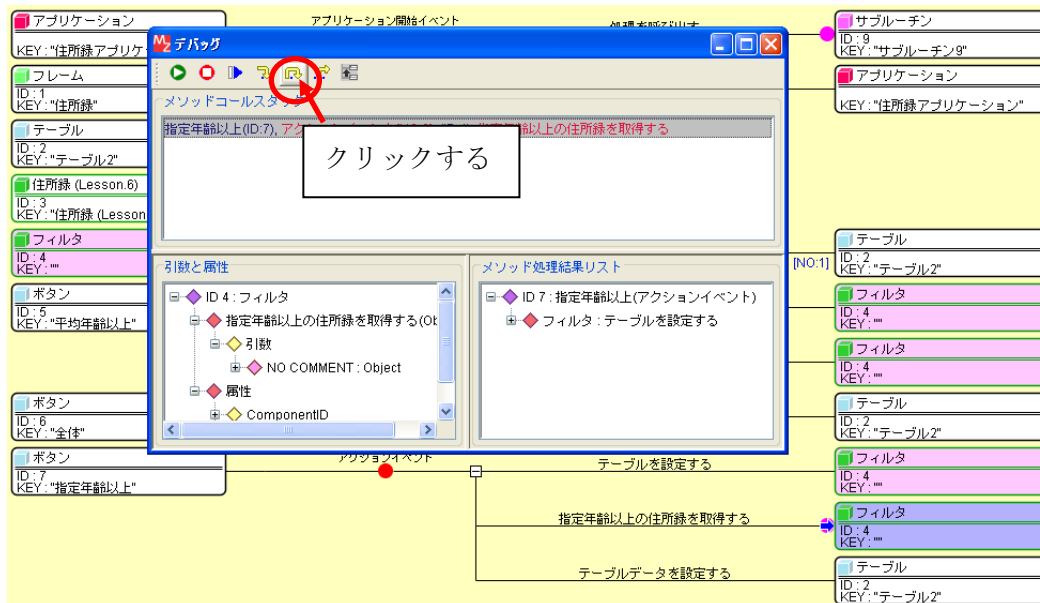


※複合コンポーネントの起動メソッドからイベントが発生しない場合は、4-7-1 と同じ処理をする

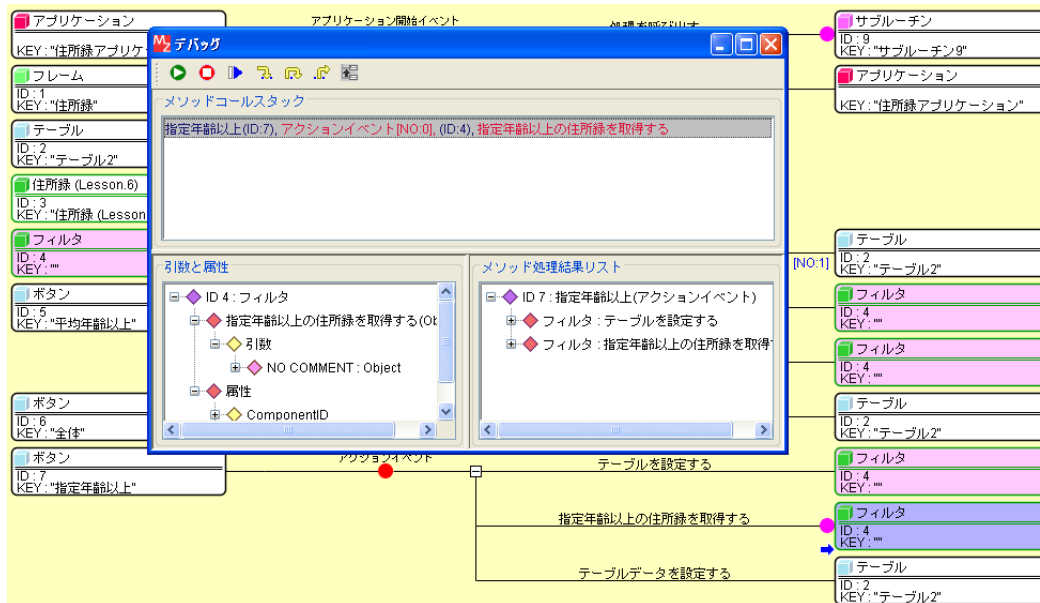
※パスワードロックが設定されている場合はステップオーバーと同じ処理を行う

4-8.ステップオーバー

- ① ステップオーバーボタンを押下する
- ② メソッド実行後の状態に移動する



メソッド実行後の状態に移動する



※ステップオーバーでは、起動メソッドから呼び出されるイベントや、複合コンポーネントであってもメソッド実行後の状態に移動する

4-9.ステップアウト

- ①ステップアウトボタンを押下する
- ②呼び出し元のメソッドの実行後の状態に移動する

このスクリーンショットは、M2デバッガのステップアウト機能を示しています。ツールバーのステップアウトボタン（赤い円で囲まれた）をクリックすると、呼び出し元のメソッドの実行後の状態に移動します。メソッドコールスタックには、ファンクション7 (ID:4-7) が実行中であることを示しています。このメソッドは、処理要求イベント [NO:0]、比較演算 (≧) 4 (ID:4-4)、および数値に変換して右オペランドに設定する指定年齢以上 (ID:7) のアクションイベントを受け取っています。メソッド処理結果リストには、ID 4-7: ファンクション7(処理要求イベント) が表示されています。左側の引数と属性には、ID 4-4: 比較演算(≧) が設定されており、右オペランドは String、属性は ComponentKey として定義されています。右側のCall Graphには、ファンクションA (ID:4-7) からファンクション7 (ID:4-7) への呼び出しが示されています。

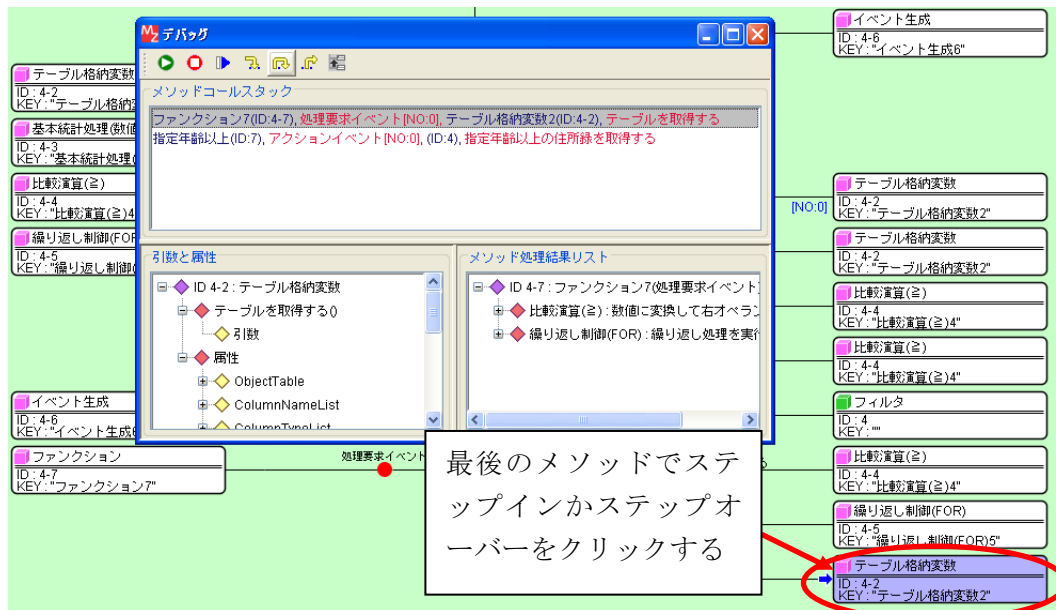
呼び出し元(親階層)のメソッドの実行後の状態に移動する

このスクリーンショットは、ステップアウト後の状態を示しています。メソッドコールスタックには、指定年齢以上 (ID:7)、アクションイベント [NO:0]、(ID:4)、および指定年齢以上の住所録を取得するが実行中であることを示しています。メソッド処理結果リストには、ID 7: 指定年齢以上(アクションイベント) が表示されています。左側の引数と属性には、ID 4: フィルタ が設定されており、右オペランドは Object、属性は ComponentID として定義されています。右側のCall Graphには、指定年齢以上 (ID:7) からアクションイベント (ID:4) への呼び出し、およびアクションイベント (ID:4) からテーブルを設定する (ID:2) への呼び出しが示されています。

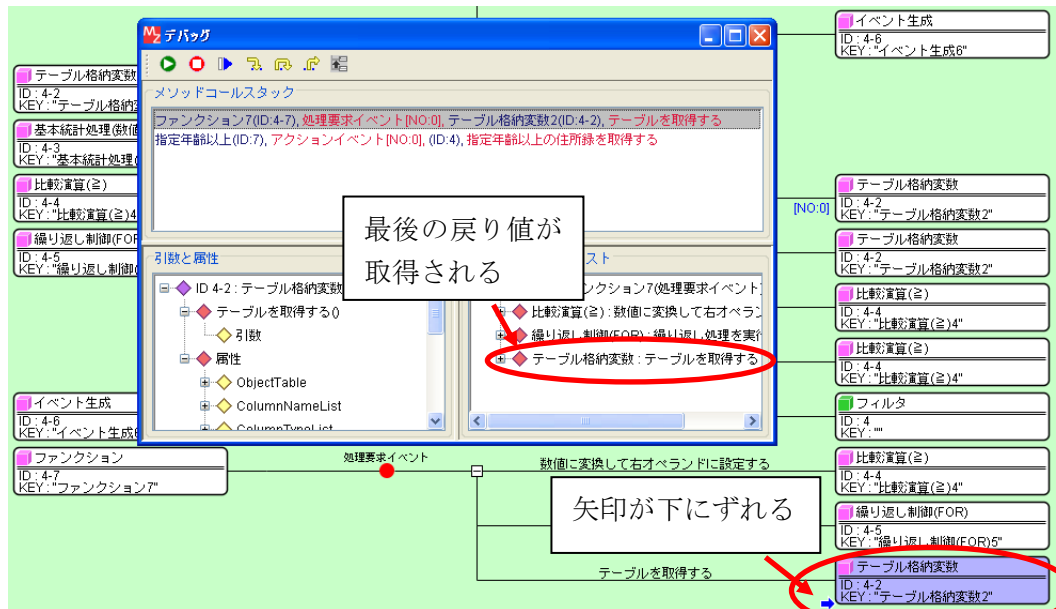
※起動メソッドから呼び出されたイベントや、複合コンポーネント内部のメソッド以外では再開ボタンを押下した時と同じ処理を行う

4-10.メソッド最後のステップ実行

- ①ステップインかステップオーバーを押下する
- ②最後のメソッドを処理して戻り値を取得する

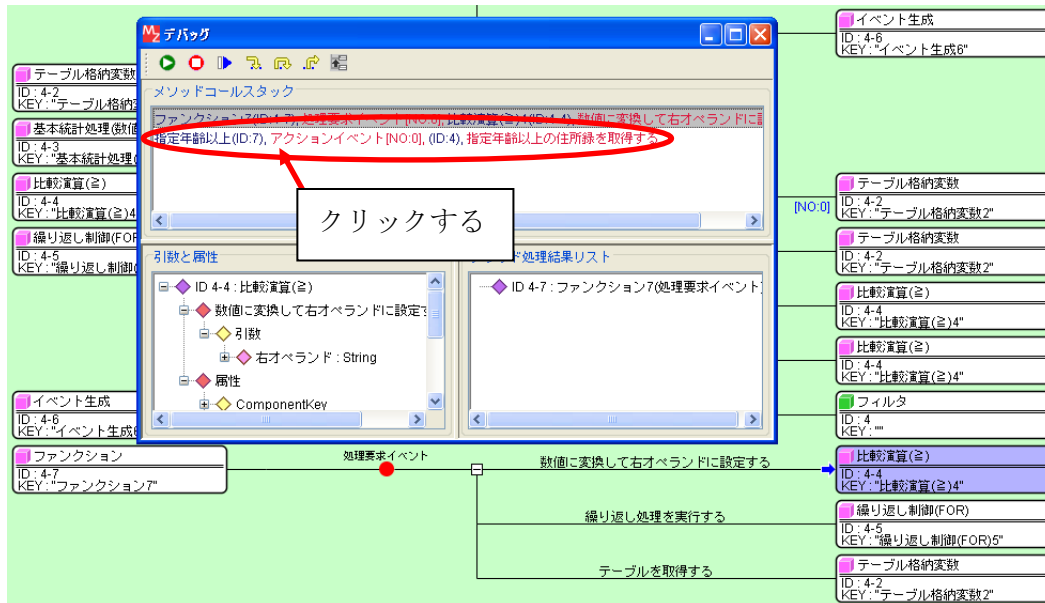


最後のメソッドを実行する

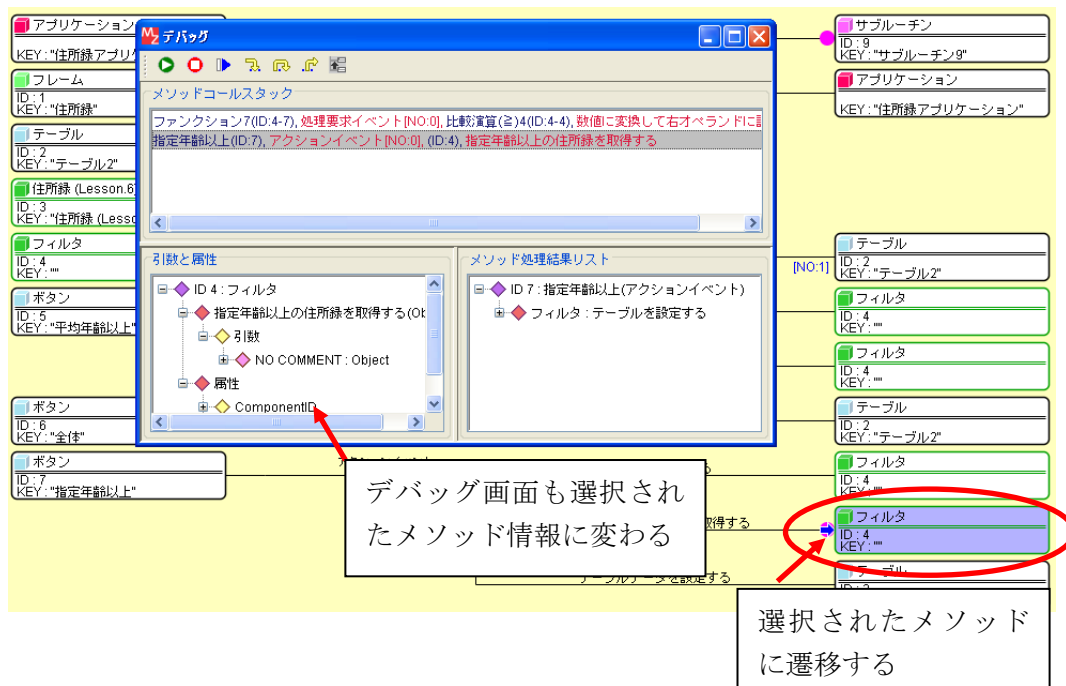


4-11.メソッドコールスタックパネル

- ①メソッド履歴パネルの行を選択する
- ②選択したメソッド情報を確認できる



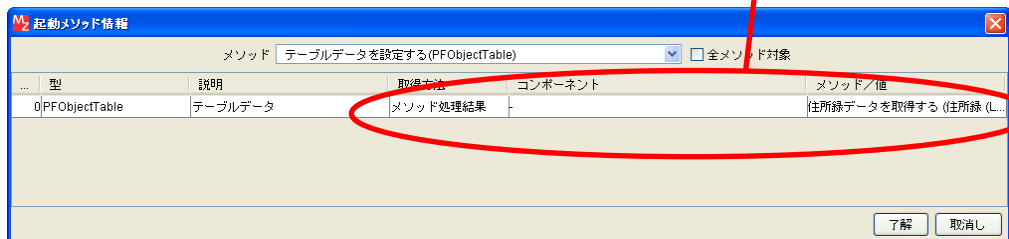
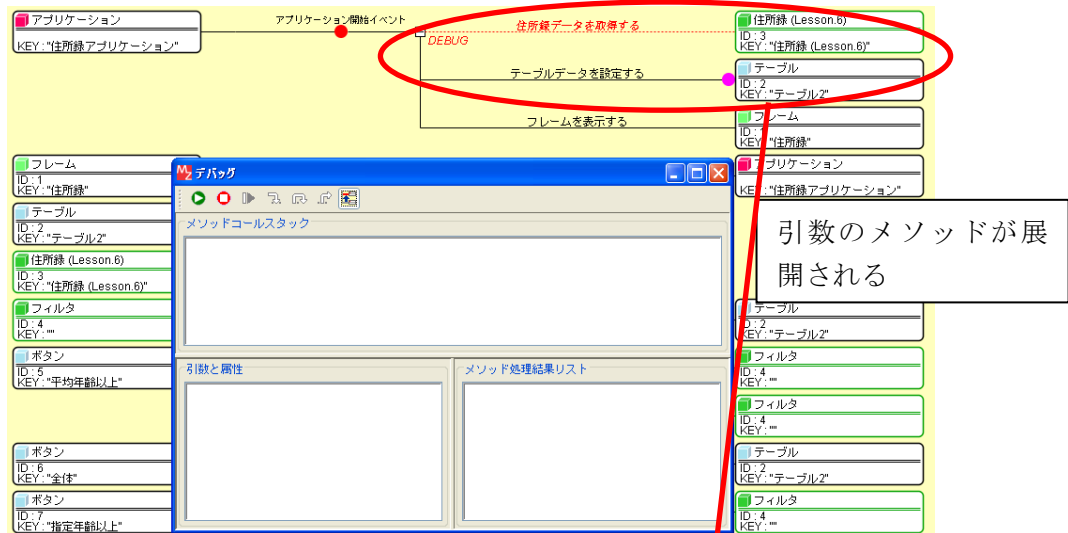
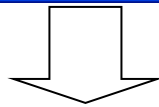
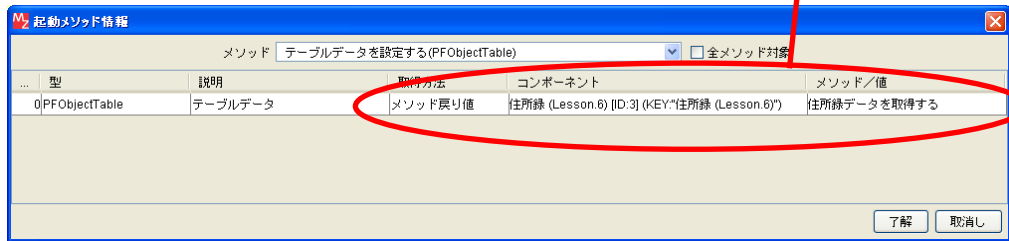
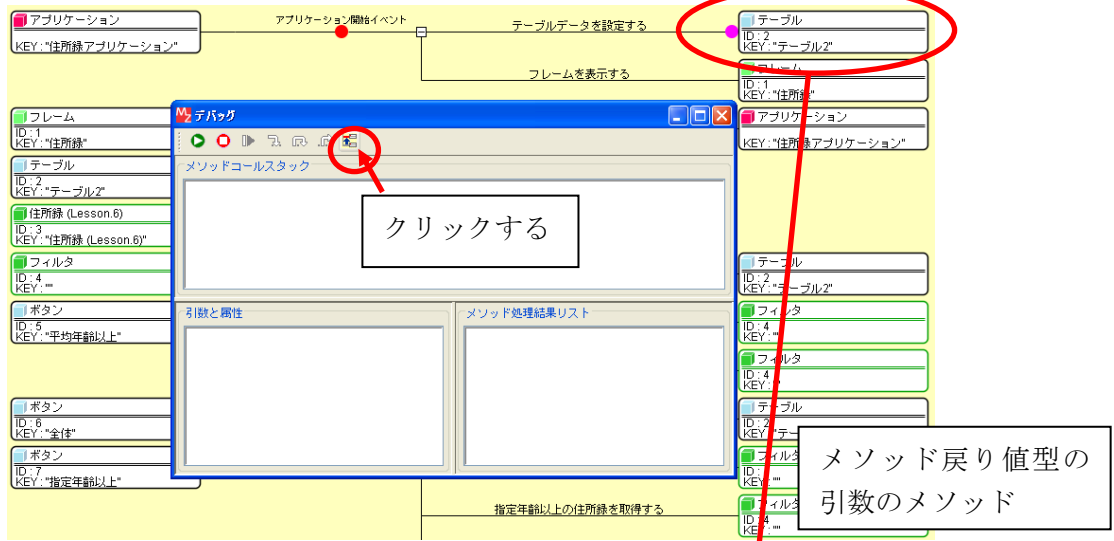
選択行の起動メソッド位置に画面遷移し、選択されたメソッドの引数、メソッド、戻り値が確認できる



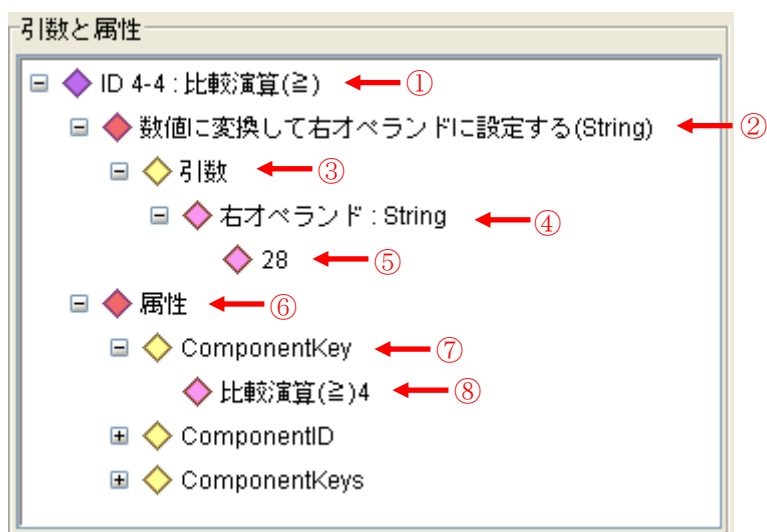
※画面表示を切り替えても実際の処理停止位置は移らない。

4-12.メソッド展開

- ①メソッド展開ボタンを押下する
- ②ブレークポイントが設定されたメソッド起動の引数のうち、メソッド戻り値型の値取得メソッドがメソッド処理結果型に変更してメソッドの外に展開する（デバッグ開始前・終了時のみ有効）



4-13.引数と属性パネル



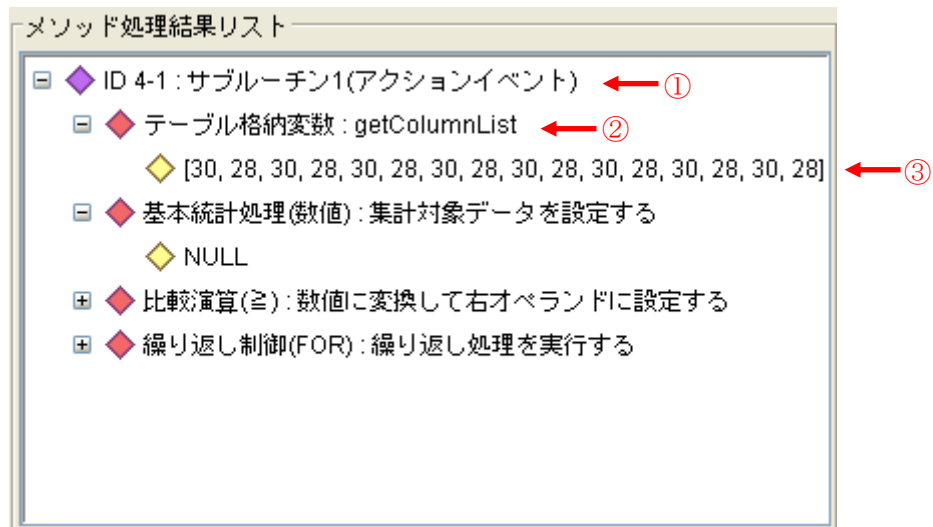
番号	説明
①	接続先コンポーネント名 (矢印が表示されているコンポーネント)
②	起動メソッド名 (ビルダーに表示されている起動メソッド名)
③	起動メソッド引数 (固定値)
④	引数の説明と型 (起動メソッド情報画面の説明と型) *1
⑤	引数の値 (起動メソッドで処理される値) *2
⑥	起動メソッド属性 (固定値)
⑦	属性名 (コンポーネント属性情報の属性名と同じ)
⑧	属性値 (コンポーネント属性情報の属性値と同じ) *3

*1.説明が無い場合は「NO COMMENT」と表示される

*2.起動メソッドに引数が無い場合は表示されない

*3.属性値に値が無い場合は表示されない

4-14.メソッド処理結果リストパネル

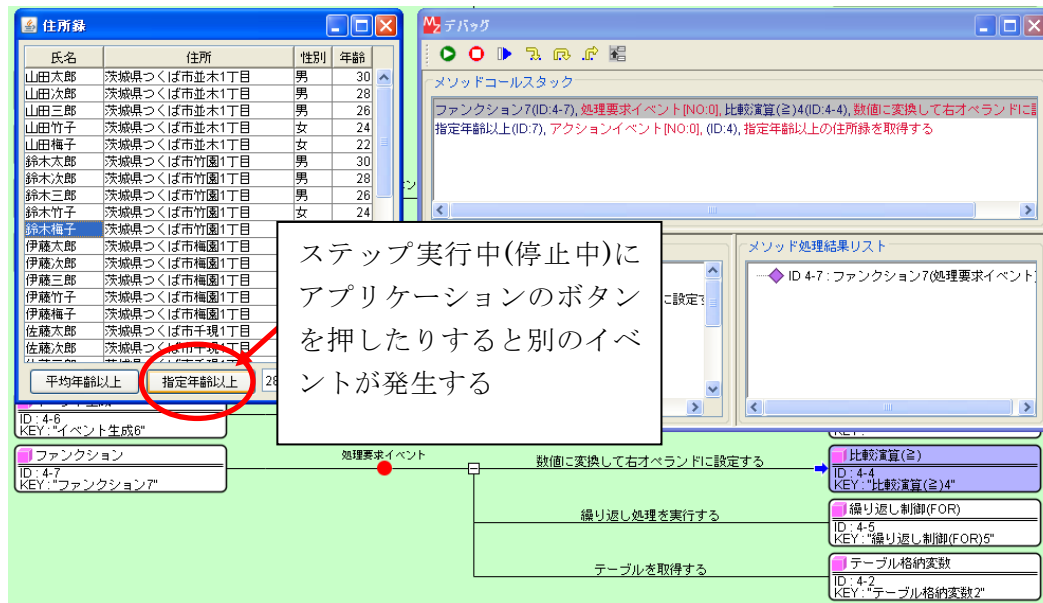


番号	説明
①	接続元コンポーネント名とイベント名
②	起動メソッド名 (ビルダーに表示されている起動メソッド名)
③	起動メソッド戻り値 (起動メソッドの処理結果) *1

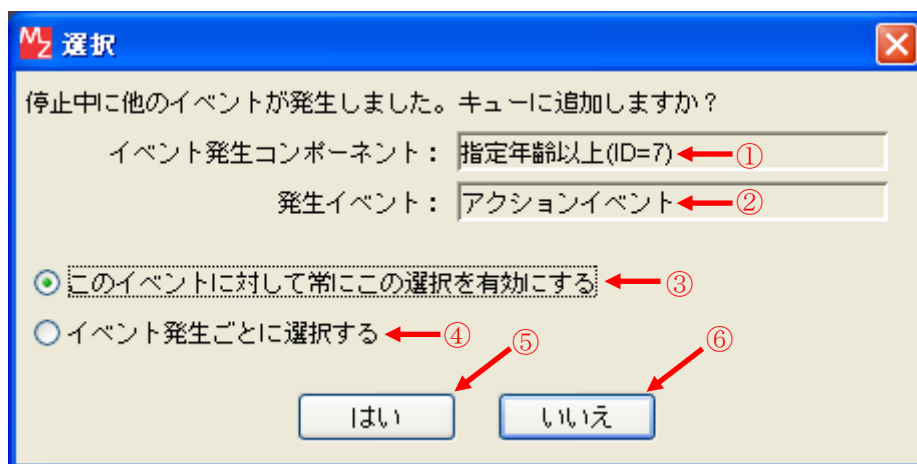
*1.戻り値が無い場合は「NULL」と表示される

4-15. イベントキュー画面

- ①ステップ実行中(停止中)に別のイベントが発生する
- ②イベントキュー処理画面が表示される



↓ イベントキュー追加選択画面が表示される



番号	説明
①	接続元コンポーネント名
②	発生イベント名
③	表示されているイベントに関して、次回以降の発生時に⑤または⑥の決定を常に有効にする。
④	表示されているイベントに関して、次回の発生時にも同じ選択画面を表示する。
⑤	③・④で選択した条件でイベントをキューに追加する
⑥	③・④で選択した条件でイベントをキューに追加しない

5. デバッガの制限事項

- デバッグ実行中は、ビルダー本体のメニューやボタンを押下しないで下さい。その場合には、その後のデバッグ実行の挙動は保証できません。
- パスワードロックされている複合コンポーネント内はステップ実行出来ません。
複合コンポーネント内をステップ実行したい場合は、ロックを解除してから実行して下さい。
- ステップ実行中に他のイベントが発生した場合（ボタンを押下、タイマーイベント等）、後から発生したイベントは、ステップ実行中のイベント処理が終了した後処理されます。
（イベントキューに追加されます。）
イベント発生時、確認ダイアログが表示されますので、イベントキューに追加しない事も出来ます。
- マルチスレッドでの並行処理を同時にステップ実行する事は出来ません。ステップ実行ではスレッドが一つずつ処理されます。
- ステップ実行中にモーダルダイアログを使用した処理に入ると正常に動作しません。デバッグ機能の性質上、ステップ実行中はすべてのダイアログを一時的に非モーダル化しているため、ダイアログが閉じてから実行されるはずの処理が先に実行されます。
- ステップ実行中にデバッグ画面の終了ボタンを押下した場合、アプリケーションを強制的に終了しますので、ファイルオープンやデータベース接続等のメソッドを追加した場合、それに対応するクローズ処理を行うメソッドを追加し、起動モードを **Finally** 起動にして下さい。
Finally 起動に設定されたメソッドは、強制終了した場合でも必ず実行されます。
（オープンしたファイルをクローズしないで終了した場合、次回のファイルオープンでエラーになります。）
- 実行対象でないイベント番号のメソッド処理結果もメソッド処理結果リストに表示されます。