



## **Consistency between Class and Sequence**

---

*by SparxSystems Japan*

Enterprise Architect 日本語版

クラス図とシーケンス図の整合性確保 マニュアル

(2011/12/6 最終更新)



## 1. はじめに

UML を利用したモデリングにおいて、クラス図は最も利用される図の 1 つです。クラス図は対象のシステムなどの構造をモデリングするために利用されます。一方で、振る舞いについてはクラス図では表現できないため、振る舞い図を利用してモデリングする必要があります。振る舞い図の中では、シーケンス図が最も多く利用される図です。

クラス図とシーケンス図の組み合わせで設計する場合には、クラス図とシーケンス図の間の整合性を確保することが重要になります。このドキュメントでは、無料で配布している「シーケンス図チェックアドイン」を利用し、この整合性を効率的に確保する方法について説明します。

なお、このドキュメントでは、**Enterprise Architect 9.2** ビルド 920 を利用しています。異なるバージョン・ビルドでは、表示内容などが異なる場合があります。

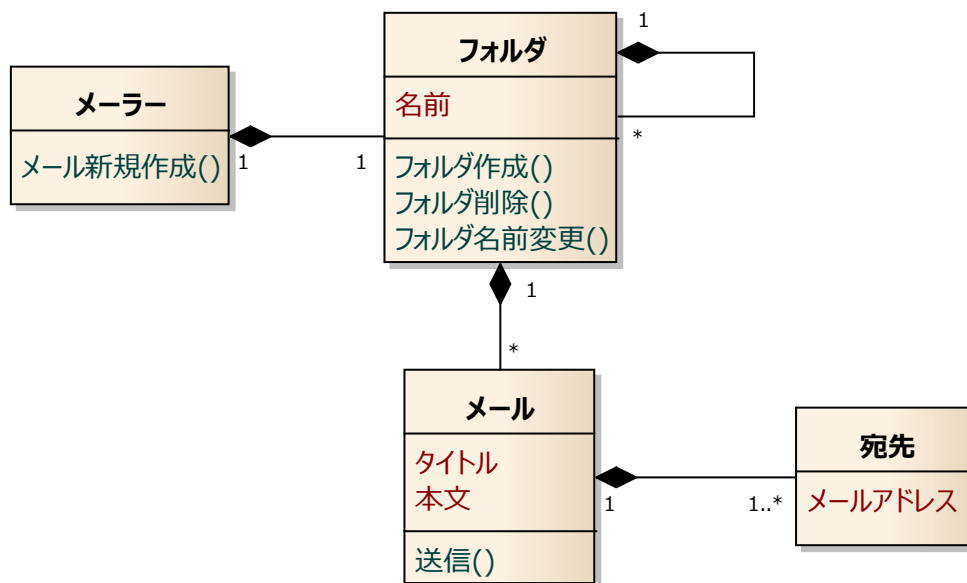
## 2. クラス図のモデリング

多くの場合には、シーケンス図等の振る舞い図より先に、クラス図で全体の構造をモデルとして表現します。このドキュメントでも、先にクラス図を作成するという仮定で話を進めます。

クラス図でのモデリングでは、それぞれのクラスに対して属性や操作を定義します。この属性や操作を過不足なく定義することが重要です。一方で、クラス図のみに着目していても、属性や操作の過不足がないかどうかを判断することは非常に困難です。

このドキュメントでは、一例として簡単なメール送受信ソフト(メーラー)を題材にします。多くの方がメーラーを日常的に利用していますので、メーラーのクラス図を書きなさい、というと、ある程度はクラス図を書くことができるのではないかと思います。

一例として、ここでは以下のようなクラス図を考えました。



なお、上記の例のように、可視性や型などの情報は非表示にすることで、注目すべき点に着目できます。

### 3. シーケンス図のモデリング

一方で、クラス図で定義した内容が適切かどうか、クラス図だけで検証することは困難です。レビューを経ることで不足している内容が見つかることは少なくありませんが、必要かどうかを判断することは容易ではありません。

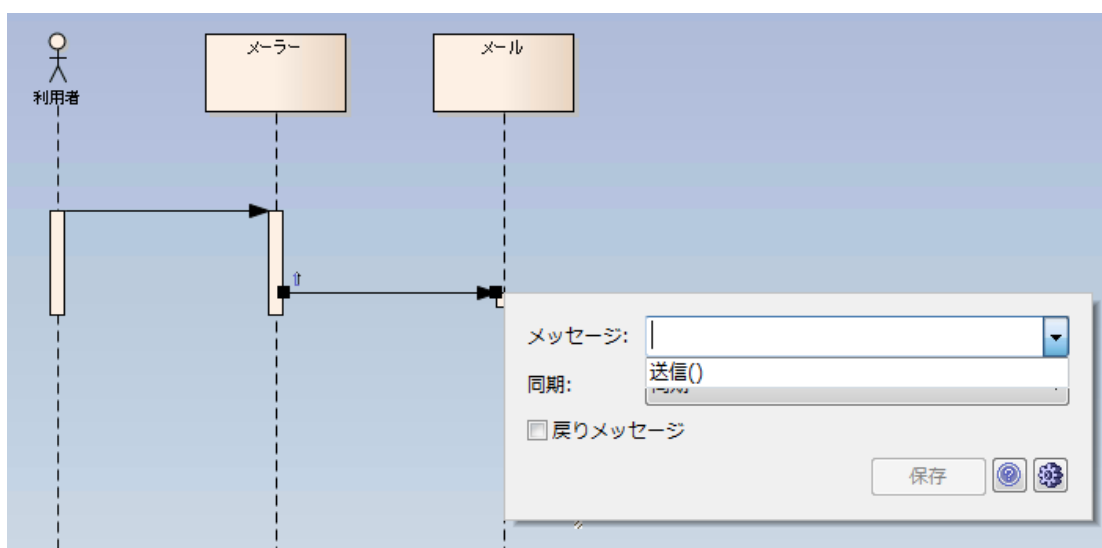
このような場合には、シーケンス図等の振る舞い図を利用して、振る舞いをモデリングすることにより、状況を洗い出すことが有効です。ここでは、クラス図の検証のために、シーケンス図を利用します。

具体的には、ユースケースのそれぞれに対して、正常系と異常系のシーケンス図を作成することになります。正常系は通常 1 つのユースケースに対して 1 つですが、異常系はその内容に応じて複数存在することが多いです。1 つのシーケンス図に複数の異常系を表現したり、正常系と異常系を複合フラグメントを利用して 1 つのシーケンス図に同居させることも可能ですし、それぞれの状況毎にシーケンス図を作成することもできます。

Enterprise Architect では、クラス図で作成したクラス要素を、プロジェクトブラウザか

らシーケンス図にドロップすることで、シーケンス図で利用することができます。

シーケンス図でメッセージを作成すると、クラス図で定義した操作(メソッド)を選択して利用することができます。以下の画像は、バージョン 9.2 以降でメッセージの作成時に表示されるシンプルプロパティ画面のコンボボックスで、クラス図で定義した操作を選択している状況の例です。

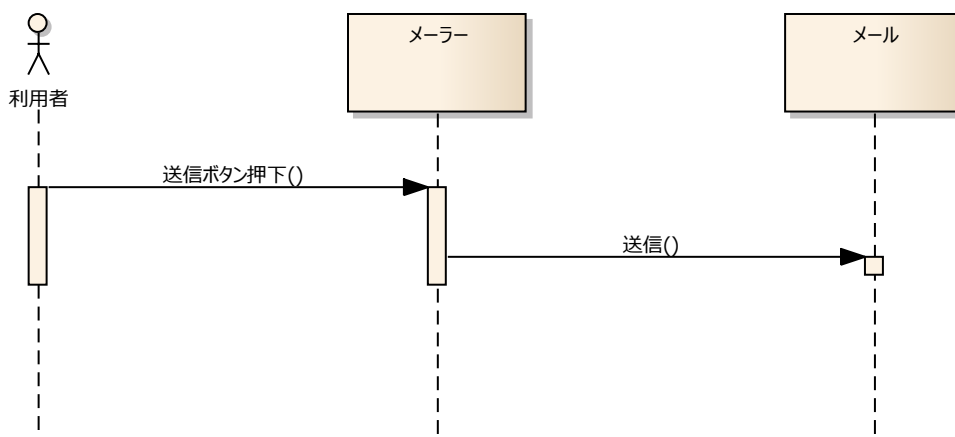


このようにして、シーケンス図を作成することで、それぞれのクラスの操作が不足なく定義されているかどうかを確認することができます。

では、コンボボックスの選択肢に必要な操作(メソッド)が存在しない場合、つまり、不足を見つけた場合には、どのようにすればよいのでしょうか。

このような状況の対応方法 1 つは、クラス図に戻り該当のクラス要素に操作を追加してから、再度シーケンス図でメッセージの操作を選択する方法があります。しかし、この方法は非効率です。(理由は後述します。)

このような状況では、メッセージのコンボボックスから選択するのではなく、必要な操作の内容を直接入力します。以下の例で、「利用者」から「メーカー」へのメッセージの「送信ボタン押下」は先ほどのクラス図には定義されていなかった操作です。メッセージのプロパティ画面で、名前に直接「送信ボタン押下」と入力しました。



一方で、ここで入力した「送信ボタン押下」は、この時点ではクラス図のメールークラスには追加されていません。

メッセージを作成する都度、クラス要素の操作として追加するという方法もありますが、この方法には以下の問題があります。

- ✓ この時点では、このシーケンス図の内容は作成中であり、この図の内容の作成が進むにつれて、作成済みの内容を変更する可能性もある。そのたびにクラス要素・クラス図の内容が変わるのは、混乱を招く  
(クラス図やクラス要素は他の設計者も共有し、参照しているという前提)
- ✓ 今後設計が進むにつれてメッセージが変更・削除される場合もあるので、作成した直後にクラス要素にも修正を加えるのが良い方法とは限らない
- ✓ 作成したメッセージを削除した場合、クラス要素に操作として残すのかそれとも削除するのか、ツールとしては判断することが難しい  
(つまり、追加は自動的に行われ、削除は手動になるという非対称の動作になる)

そのため、シーケンス図の作成中(設計中)にリアルタイムに操作として追加する(あるいは、ツール側で自動的に追加される)のは、適切とは言い切れません。

一方で、このようにしてシーケンス図側で名前入力した場合に、以下の問題があります。

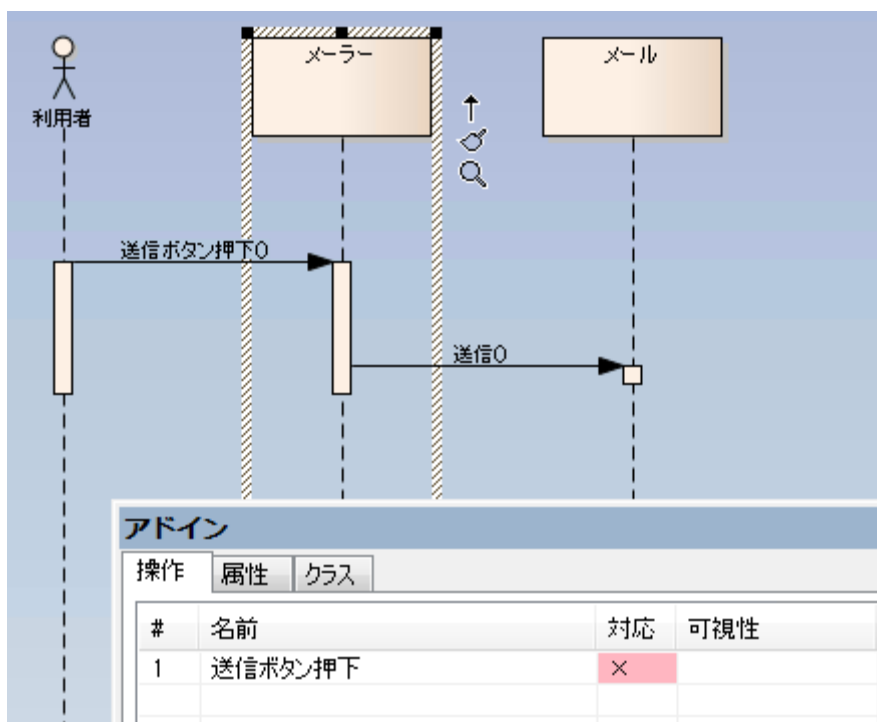
- ✓ クラスの操作から選択して設定したメッセージと、直接入力して名前を設定したメッセージの区別が付かない

- ✓ 最終的にはクラス要素の操作として反映する必要があるが、いつどのような方法で反映させるのか

この問題は、無料の「シーケンス図チェックアドイン」を利用することで解決できます。アドインが有効になっていれば、メインメニューから「アドイン・拡張」→「アドインサブウィンドウ」を実行すると、表示することができます。  
(このアドインについては、「6. シーケンス図チェックアドインについて」もご覧ください。)

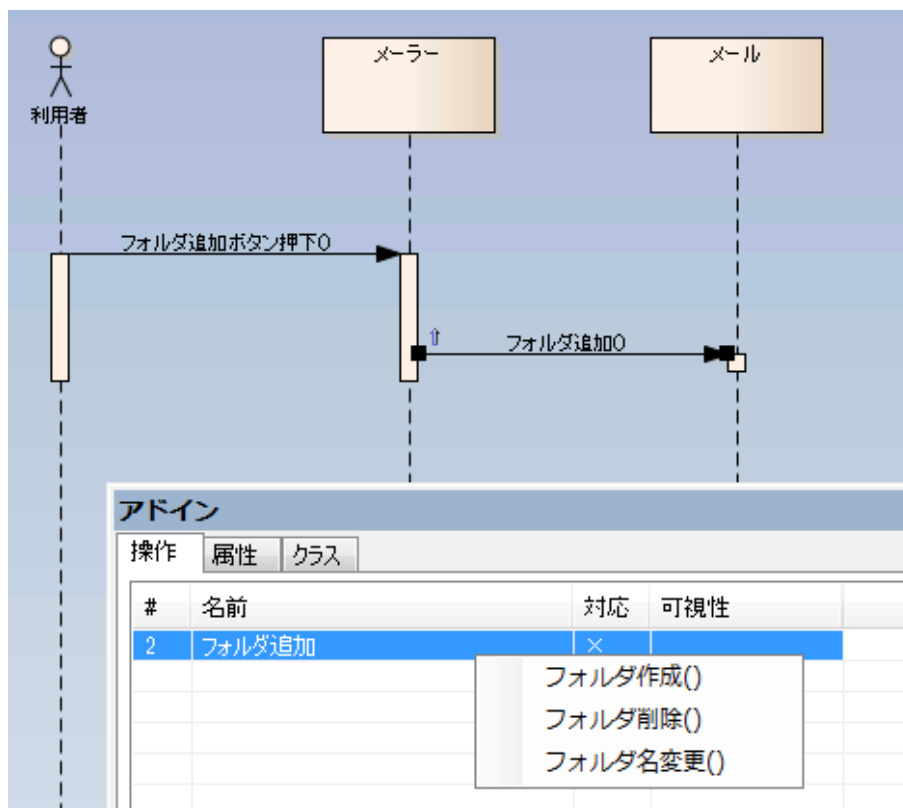
作成済みの全てのシーケンス図の内容を完成させた段階で、このアドインを利用してクラス図とシーケンス図の間の整合性を確保します。シーケンス図の内容のレビューを行う場合には、レビュー中にこのアドインも起動し、内容のレビューと整合性確認を同時に進めると効果的です。  
(まず、シーケンス図のそれぞれのメッセージが妥当かどうかをレビューによって確認し、妥当な場合でクラス図に操作が存在しない場合には、このアドインを利用して操作を追加する。)

先ほどのシーケンス図を例にしますと、このアドインが表示されている状態で、シーケンス図内の要素をクリックします。すると、下の図のように、メッセージの内容を検証することができます。



この例では、「対応」の欄が「×」、つまり、クラス図に操作として存在しないことを示しています。この場合に、該当の行をダブルクリックすることで、クラスの操作として追加することができます。

また、既存の操作に「差し替える」ことができます。以下の例をご覧ください。



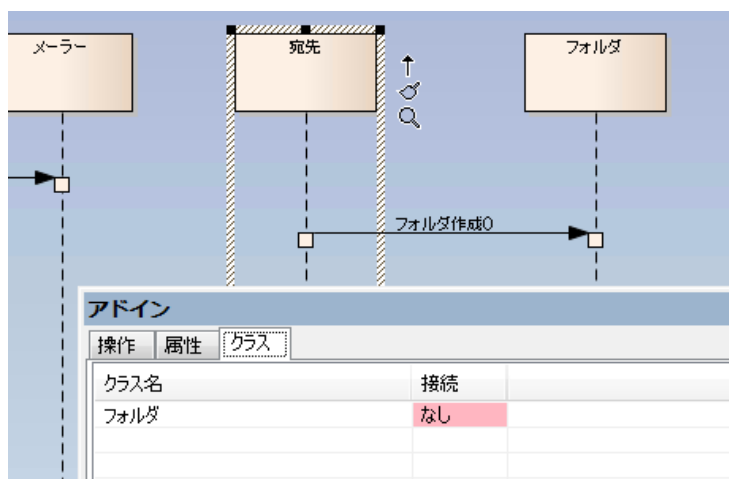
この例では、シーケンス図の作成中には「フォルダ追加」と直接入力しましたが、実は「フォルダ作成」という名前の同機能の操作が既に存在しました。つまり、「フォルダ追加」という操作を新規に追加することは適切ではなく、既存の「フォルダ作成」を利用するようにならなければなりません。

右クリックすることで既存の操作をコンテキストメニューの項目として表示します。希望する操作を選択することで、差し替えることができます。

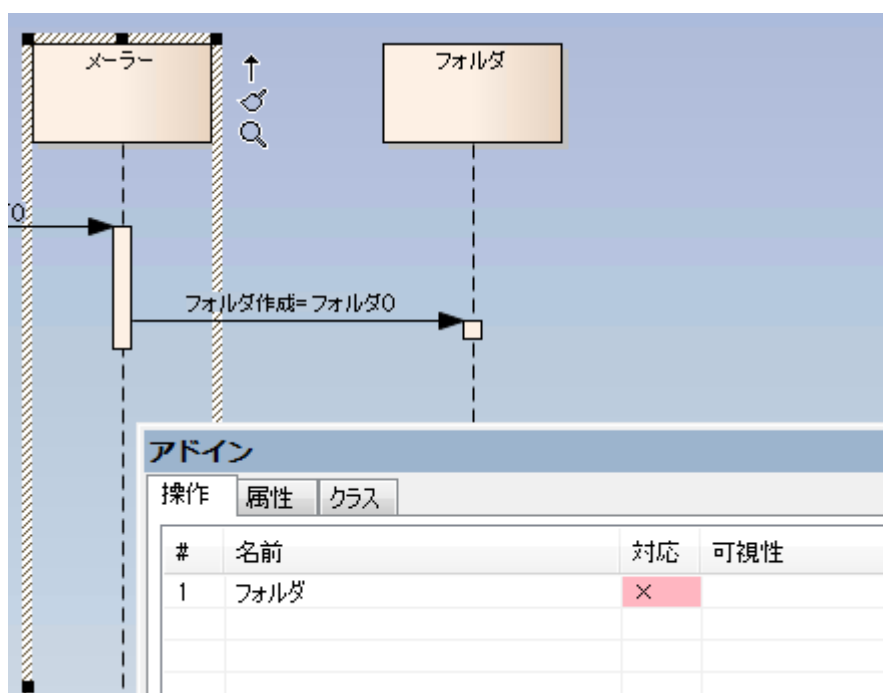
なお、このシーケンス図チェックアドインでは、以下の内容の確認も可能です。これらの機能を利用することで、クラス図とシーケンス図の間の整合性を確認することができます。

- ✓ private な操作を外部から呼び出していないかどうか
- ✓ クラス要素間で関係がないにもかかわらず、シーケンス図でメッセージを送信していないかどうか

(以下の例では、クラス図で関係のない「宛先」クラスを配置して、この整合性に問題がある状況を作りました。「クラス」タブでは、クラス要素間の関係と整合しているかどうかを確認することができます。)



なお、メッセージの戻りを属性に格納する場合に、プロパティ画面の「割当」に入力することで、クラス要素の属性とシーケンス図の整合性も確認できます。下の図では、「属性」タブに表示される内容で、属性についての整合性を確認しています。





## 4. 整合性の最終確認

ここまで説明した方法で、以下の内容ができました。

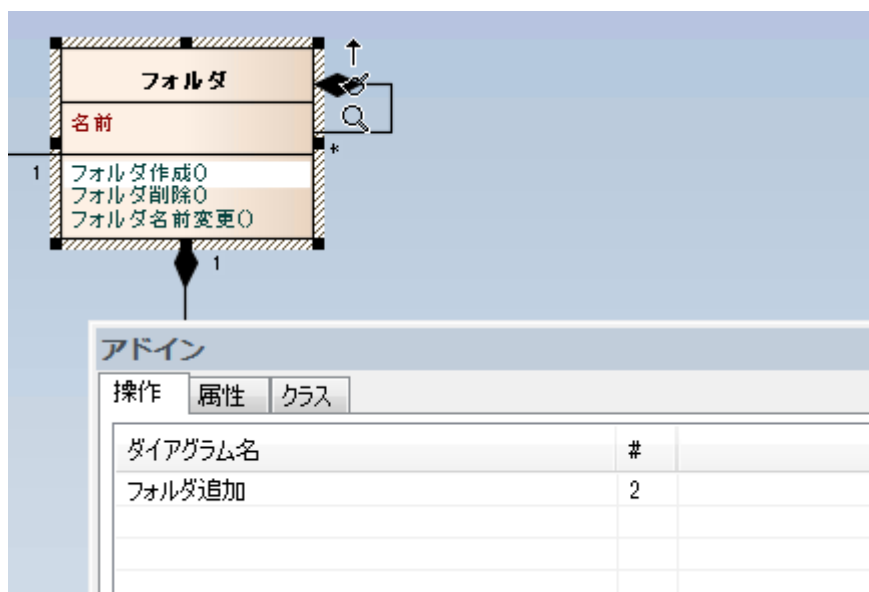
1. まず、クラス図で全体の構成・クラスを定義する
2. それぞれのクラスの属性や操作を定義する
3. さまざまな状況でのシーケンス図を作成することで、不足している操作や属性を発見する
4. レビュー後、最終的にクラスに追加すべき項目を、クラス要素に追加する

一方で、ここまでの手順の中では、上記の 2 番で定義した操作や属性のうち、どこのシーケンス図でも利用されていないもの、つまり「過不足」の「過」を見つけることができていません。

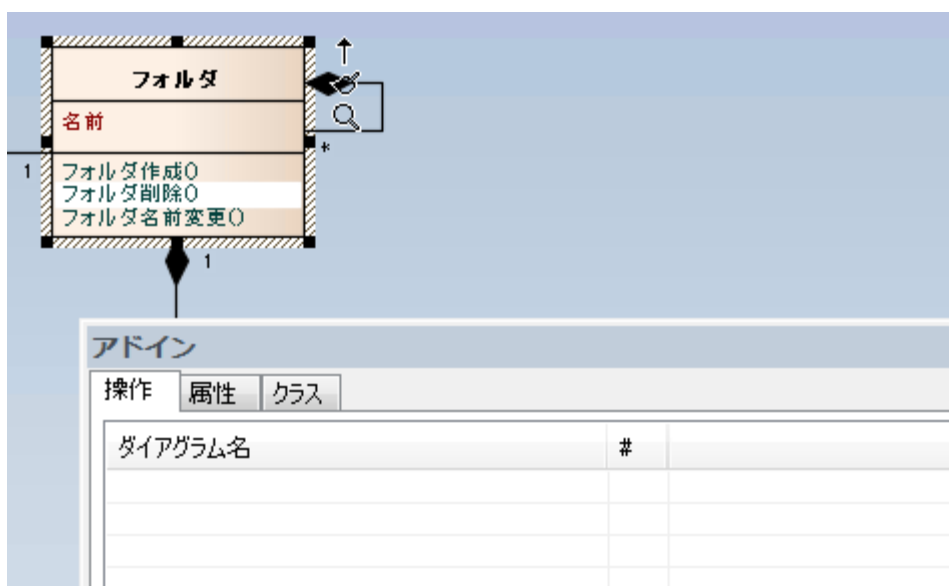
この「過」を見つけるために、クラス図を開きます。

シーケンス図チェックアドインが表示されている状態で、クラス図やプロジェクトブラウザ内で操作や属性を選択します。すると、その操作が利用されているシーケンス図の情報が、表示されます。下の例の場合には、「フォルダ作成」の操作は、「フォルダ追加」シーケンス図の 2 番目のメッセージに利用されていることが分かります。

(一覧内の行をクリックすると、該当のシーケンス図が開きます)



一方で、「フォルダ削除」の操作を選択した場合には、この一覧に何も表示されません。



これは、次のいずれかの状況であることを示しています。

- ✓ この操作はどこのシーケンス図でも利用されていないので、不要である
- ✓ この操作が含まれるシーケンス図の作成作業が漏れている

不要な場合には操作を削除し、この操作が必要な場合には、対応するシーケンス図を作

成して設計内容を検討・検証します。

このようにすることで、クラス図とシーケンス図の間の過不足をなくし、整合性を確保することができます。

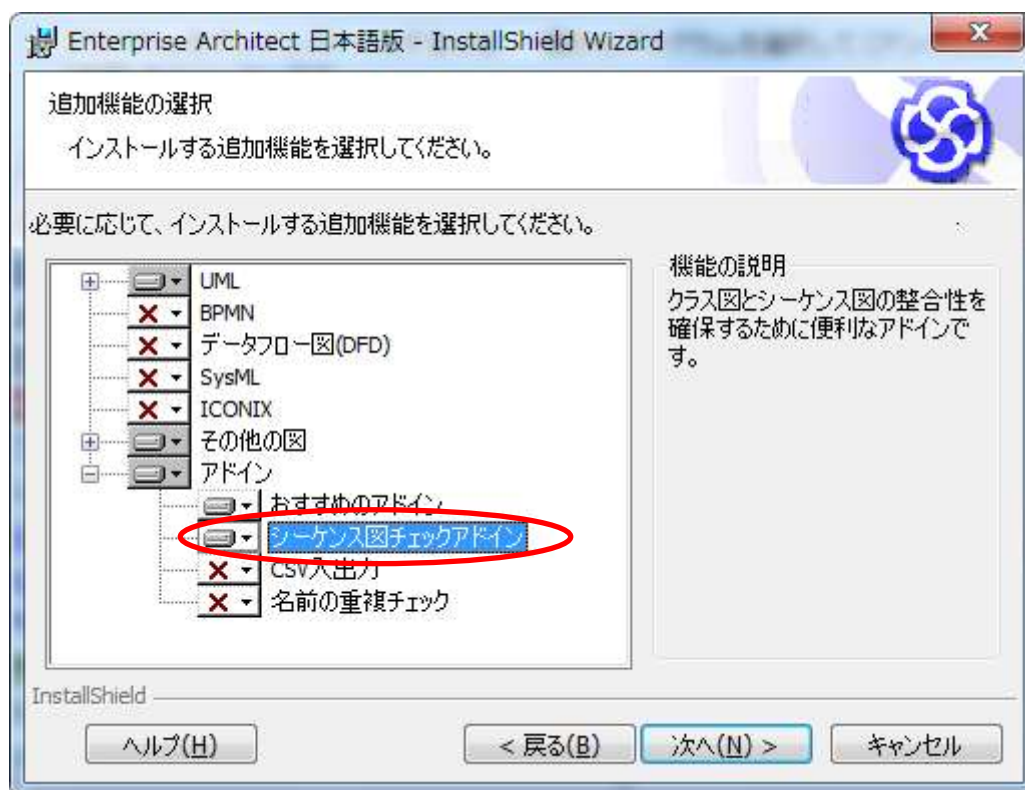
## 5. まとめ

クラス図とシーケンス図の整合性確保は、無料の「シーケンス図チェックアドイン」を利用することで実現できます。この段階で考慮漏れや無駄のないクラス図を定義することにより、この後の工程の無駄を省くこともできます。

ぜひご活用ください！

## 6. シーケンス図チェックアドインについて

このドキュメントで利用しているシーケンス図チェックアドインは、**Enterprise Architect** で無料で利用できるアドインです。利用するためには、インストール時に表示される機能選択の画面で、以下のように「シーケンス図チェックアドイン」をインストール対象に追加してください。



このアドインについての詳細は、ヘルプファイルの「シーケンス図チェックアドイン」のページをご覧ください。