ENTERPRISE ARCHITECT

Série de Guides d'Utilisateur

Enregistrement

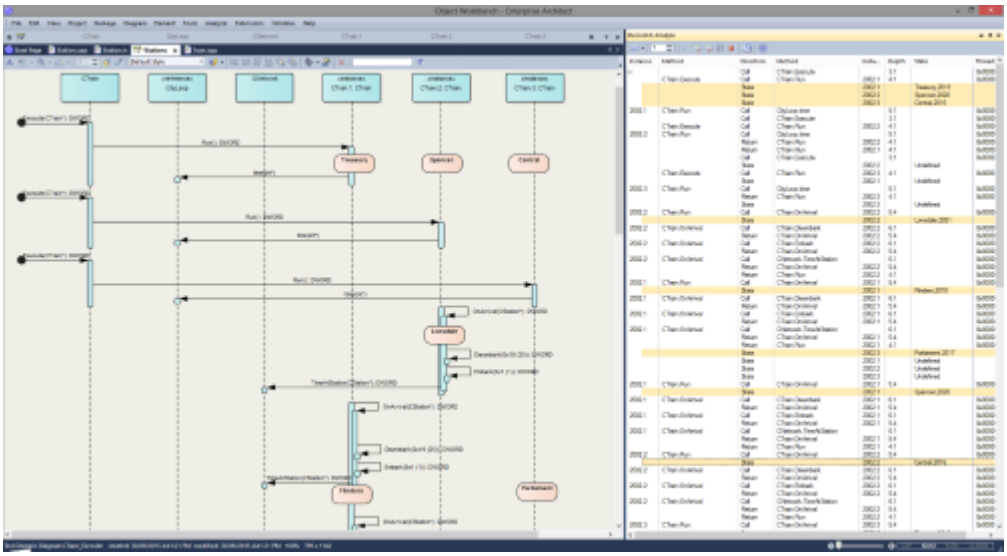
Author: Sparx Systems
Date: 23/11/2023
Version: 16.1

CRÉÉ AVEC ENTERPRISE ARCHITECT

Table des Matières

Enregistrement 3
- Comment ça fonctionne 7
  - L' Historique d'Enregistrement 9
  - Fonctionnalités Diagramme 11
- Configuration pour Enregistrement 12
  - Contrôle Profondeur de Pile 13
- Placer des Marqueurs d'Enregistrement 14
  - Coffret Marqueurs d'Enregistrement 15
  - Types de Marqueurs 16
  - La Fenêtre Points d'Arrêt et Balises 18
  - Travailler avec Ensembles de Marqueurs 19
- Contrôler la Session d'Enregistrement 21
  - Barre d'Outils d'Enregistreur 22
  - Travailler avec Historique d'Enregistrement 24
  - Démarrer Enregistrement 26
  - Parcourez Appels de Fonction 28
  - Marqueurs d'Enregistrement Nested 29
- Génération Diagrammes Séquence 30
- Transitions State Rapportage 32
  - Rapportage d’une Statemachine 33
  - Enregistrement et Mappage Changements de State 35
- Analyseur de State 36
- Synchronisation 43

Enregistrement



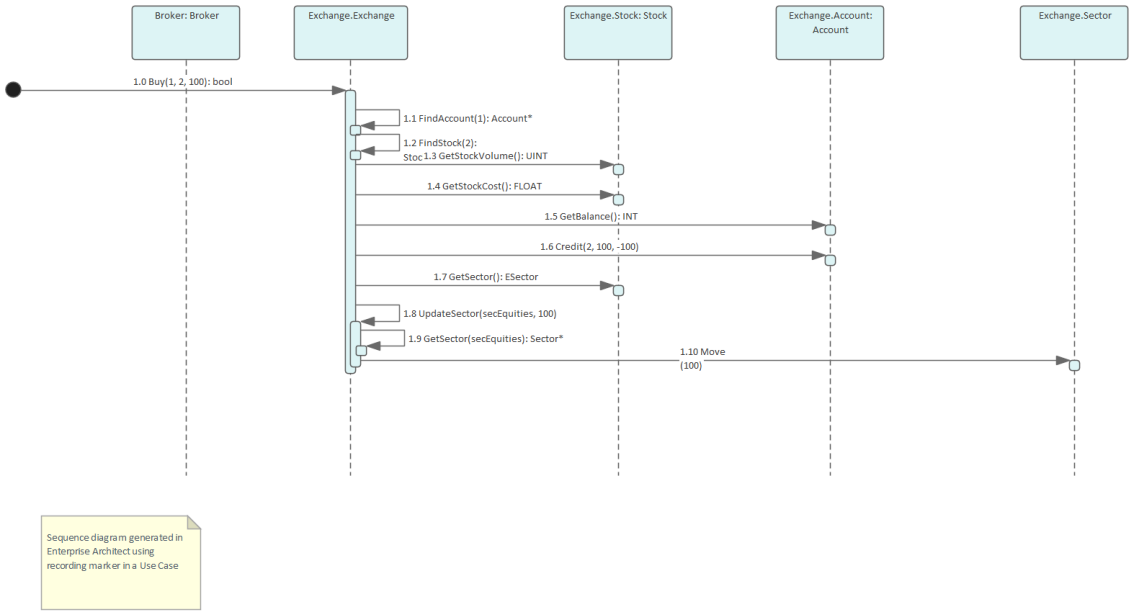
diagrammes Séquence sont une aide précieuse pour comprendre le comportement. diagrammes de collaboration de classe peuvent également être utiles. En plus de cela, un Graphique d'Appel est parfois exactement ce dont nous avons besoin. Là encore, si vous disposez de ces informations, vous pouvez les utiliser pour documenter un cas d'utilisation, et pourquoi ne pas créer un domaine Test pendant que vous y êtes ? Enterprise Architect Analyzer peut générer tous ces éléments pour vous et à partir d'un seul enregistrement. Pour ce faire, il enregistre un programme en cours d'exécution et fonctionne sur toutes les plateformes les plus populaires.

Accéder

Ruban	Exécuter > Outils > Enregistreur

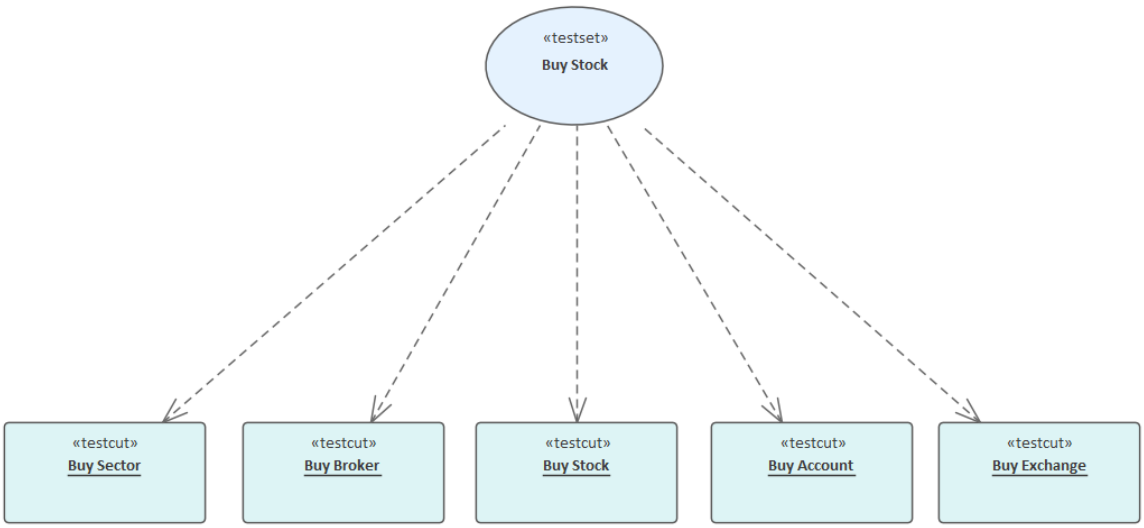
Aperçu

Dans sa forme la plus simple, un diagramme Séquence peut être produit en très peu d'étapes, même en utilisant un tout nouveau modèle. Vous n'avez même pas besoin de configurer un script Analyzer. Ouvrez l'éditeur de code Enterprise Architect (Ctrl+Shift+O), placez un marqueur d'enregistrement dans une fonction de votre choix, puis attachez le Débogueur Enterprise Architect à un programme exécutant ce code. Chaque fois que cette fonction est appelée, son comportement sera capturé pour former un historique d'enregistrement. À partir de cet historique, ces diagrammes peuvent être facilement créés.



Le diagramme Séquence de l'enregistrement Exemple Modèle .

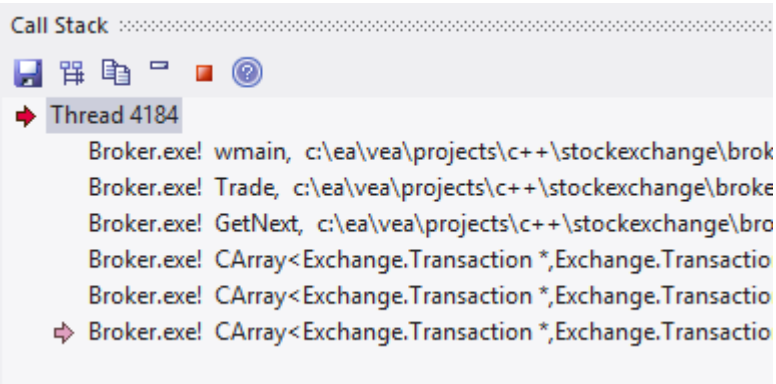
Le diagramme de collaboration de classe du même enregistrement.



Le diagramme du domaine Test du même enregistrement.

Bien sûr, un script Analyzer reste la meilleure idée et ouvre un environnement de développement incroyablement riche, mais il convient de noter que des résultats significatifs peuvent être obtenus sans un tel script. Cela est également vrai pour les outils Enterprise Architect Débogueur et Profiler.

Un point intéressant : vous pouvez visualiser le comportement d'un fil pendant son enregistrement. L'affichage de la Pile d'Appel pendant un enregistrement affichera les mises à jour de la pile d'un fil en temps réel, un peu comme une animation. Il s'agit d'un bon outil de rétroaction et, dans certaines circonstances, cela peut suffire.



Aperçu des Fonctionnalités

Génération Diagramme

- diagramme Séquence
- diagramme de collaboration de classe
- diagramme du domaine Test
- Capture de transition State
- Graphique d'Appel

Contrôle

- Support des modèles multithread et monothread
- Support du contrôle de la profondeur de la pile

- Filtres Support pour restreindre la capture
- Filtrer support des caractères génériques

- Mise à jour de la pile en temps réel

L'intégration

- Modèle de classe
- Domaine Test
- Statemachine
- Statemachines Exécutables
- Tests unitaires

Plateformes

- .NET
- Microsoft natif
- Java
- PHP
- GDB
- Android

Exigences

- Enregistrement est disponible pour les utilisateurs de toutes les éditions d' Enterprise Architect

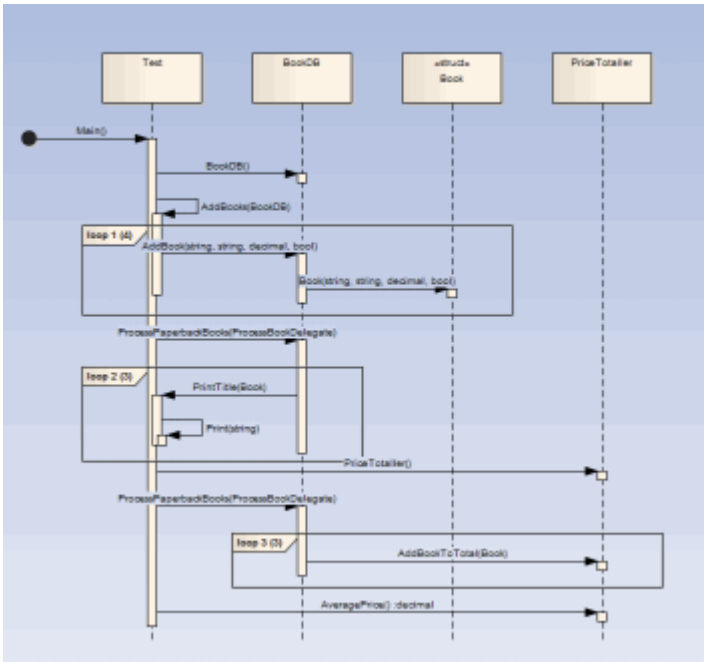
Notes

- Les fonctionnalités de débogage et d'enregistrement de l' Analyseur d'Exécution Visuelle ne sont pas supportées pour la plateforme serveur Java 'Weblogic' d'Oracle

Comment ça fonctionne

Cette rubrique explique comment l' Analyseur d'Exécution Visuelle génère diagrammes Séquence .

Explication

Points	Détail
Usage	L' Analyseur d'Exécution Visuelle permet de générer un diagramme Séquence à partir des enregistrements de l'exécution live d'une application. Au fur et à mesure de l'exécution de l'application, l'historique de chaque thread est enregistré. Cet historique peut être utilisé pour générer le diagramme Séquence . Il s'agit d'un diagramme Séquence généré à partir d'un programme qui calcule le prix des livres :  Comment l'enregistreur sait-il quoi enregistrer ? • L'enregistreur fonctionne à partir de marqueurs d'enregistrement ; ceux-ci sont placés par vous dans les fonctions qui vous intéressent Les piles d'appels en Java peuvent s'étendre plus loin que ce que l'œil peut voir. Comment pouvons-nous limiter l'enregistrement à dix images seulement ? • L'enregistreur est contrôlé par la profondeur soit définie sur la barre d'outils de l'enregistreur, soit associée à un jeu de marqueurs stocké dans le modèle.
C'est la vraie chose	En enregistrement, l'application cible n'est pas modifiée ; aucune instrumentation d'une image ou d'un module ne se produit du tout. Un enregistrement produit à l'aide d'une version « Release » d'un programme est un document fiable de ce qu'un programme a fait.
Par où commencer	Nous avons une très grande application serveur ; alors par où commencer ? Si vous avez peu ou pas de compréhension du programme que vous avez l'intention d'enregistrer et peu ou pas de modèle à défendre, il serait peut-être préférable de

	commencer avec le Profiler. L'exécution du profileur tout en utilisant un programme d'une manière spécifique peut identifier rapidement les cas d'utilisation à partir des points d'entrée et des graphiques d'appel présentés. Avoir ces connaissances peut vous permettre de vous concentrer sur les zones découvertes et d'enregistrer ces fonctions. Si vous disposez du code source, il vous suffit de placer un marqueur d'enregistrement dans une fonction qui vous intéresse. Nous vous déconseillons de placer plusieurs marqueurs d'enregistrement dans plusieurs fonctions en même temps. En pratique, cela s'est révélé moins utile. Où placer un marqueur d'enregistrement ? Pour les programmes UI Windows, et en relation avec certains cas d'utilisation professionnelle, vous pouvez commencer par en placer un dans les gestionnaires d'événements pour un message qui semble le plus pertinent. Si vous étudiez une fonction utilitaire, définissez simplement un marqueur d'enregistrement de méthode au début ou à proximité. Pour les services, les démons et les processus par lots, vous souhaiterez peut-être profiler le programme une fois pour chaque comportement d'intérêt et utiliser le rapport pour explorer les domaines découverts.
Conseil	C'est une bonne idée de jeter un coup d'œil rapide à la fenêtre Points d'Arrêt et marqueurs avant le débogage, et de vérifier que les marqueurs répertoriés ici correspondent à ce que vous attendez.
Scénarios	• Microsoft Natif C et C++, VB (Programmes Windows , services Windows, programmes console, serveurs COM, modules IIS ISAPI, Legacy) • .NET (ASP.NET, Windows Presentation Foundation (WPF), Windows Forms, Workflow Services, appareils, émulateurs) • Java (Applications, applets, servlets, beans) • Android (en utilisant le pont de débogage Android pour les appareils et les émulateurs) • PHP (Scripts de sites Web) • GDB (Interopérabilité Windows /Linux)

L' Historique d'Enregistrement

Lorsque l'analyse de l'exécution d'une application rencontre des marqueurs d'enregistrement définis par l'utilisateur, toutes les informations enregistrées sont conservées dans la fenêtre Enregistrer et analyser.

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur

Facilités

Facilité	Informations/Options
Affichage des informations	Les colonnes de la fenêtre Enregistrer et analyser sont : • Séquence - Le numéro de séquence unique • Threads - L' ID du thread du système d'exploitation • Delta - Le temps CPU du thread écoulé depuis le début de la séquence • Méthode - Il y a deux colonnes Méthode : la première affiche l'appelant pour un appel ou pour une trame en cours en cas de retour ; le second montre la fonction appelée ou la fonction à laquelle elle revient • Direction - Stack Frame Movement, peut être Call, Return, State , Point d'Arrêt ou Escape (Escape est utilisé en interne lors de la production d'un diagramme Séquence , pour marquer la fin d'une itération) • Profondeur : la profondeur de la pile au moment d'un appel ; utilisé dans la génération de diagrammes Séquence • State - L'état entre les séquences • Source - Il y a deux colonnes Source : la première affiche le nom du fichier source et le numéro de ligne de l'appelant pour un appel ou, en cas de retour, pour une trame en cours ; le second affiche le nom du fichier source et le numéro de ligne de la fonction appelée ou de la fonction renvoyée • Instance - Il existe deux colonnes Instance, qui n'ont des valeurs que lorsque le diagramme Séquence produit contient des transitions State ; les valeurs sont constituées de deux éléments séparés par une virgule : le premier élément est un numéro unique pour l'instance de la classe qui a été capturée et le second est l'instance réelle de la classe. Par exemple : supposons qu'une classe 'CName' ait une valeur interne de 4567 et que le programme ait créé deux instances de cette classe ; les valeurs pourraient être : - 4567,1 - 4567,2 La première entrée montre la première instance de la classe et la deuxième entrée montre la deuxième instance
Opérations sur l'information	La barre d'outils de la fenêtre Enregistrer et analyser offre une gamme de facilités pour contrôler l'enregistrement de l'exécution d'un script Analyzer. Vous pouvez effectuer un certain nombre d'opérations sur les résultats d'un

	enregistrement, à l'aide du menu contextuel de la fenêtre Enregistrer et analyser, une fois l'enregistrement terminé.

Notes

- La case à cocher en regard de chaque opération est utilisée pour contrôler si cet appel peut être utilisé ou non pour créer un diagramme Séquence , de classe de domaine Test ou de classe collaborative à partir de cet historique.
- En plus d'activer ou de désactiver l'appel à l'aide de la case à cocher, vous pouvez utiliser les options du menu contextuel pour activer ou désactiver un appel entier, tous les appels à une méthode donnée ou tous les appels à une classe donnée.

Fonctionnalités Diagramme

Lorsque vous générez un diagramme Séquence , il inclut ces fonctionnalités :

Fonctionnalités

Fonctionnalité	Détail
Les références	Lorsque l' Analyseur d'Exécution Visuelle ne peut pas faire correspondre un appel de fonction à une opération dans le modèle, il crée quand même la Séquence mais crée également une référence pour toute Classe qu'il ne peut pas localiser. Il fait cela pour toutes les langues.
Fragments	Les fragments affichés dans le diagramme Séquence représentent des boucles ou des itérations d'une ou plusieurs sections de code. L' Analyseur d'Exécution Visuelle tente de faire correspondre la portée de la fonction avec les appels de méthode pour représenter visuellement l'exécution aussi précisément que possible.
States	Si une Statemachine a été utilisée pendant le processus d’enregistrement, toutes les transitions dans State sont présentées après l’appel de méthode qui a provoqué la transition. States sont calculés au retour de chaque méthode à son appelant.

Configuration pour Enregistrement

Cette section explique comment préparer l'enregistrement de l'exécution de l'application.

Pas

Étape
Prérequis - Pour configurer l'environnement d'enregistrement diagrammes Séquence , vous devez : • Avoir terminé la configuration de base de Build & Déboguer et créé des scripts d'analyse d'exécution pour le Paquetage • Être capable de déboguer avec succès l'application
Affinez le focus d'un enregistrement en appliquant des filtres.
Contrôlez les détails d'un enregistrement en ajustant la profondeur de la pile.

Contrôle Profondeur de Pile

Lors de l'enregistrement de points de niveau particulièrement élevé dans une application, le nombre d'images de pile peut entraîner la collecte de nombreuses informations ; pour obtenir une image plus rapide et plus claire, il est préférable de limiter la profondeur de pile dans la barre d'outils :

- La fenêtre Point d'Arrêt et marqueurs ou
- La fenêtre Enregistrer et analyser

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur

Définir la profondeur de la pile d'enregistrement

Vous définissez la profondeur de la pile d'enregistrement dans le champ numérique de la barre d'outils de la fenêtre Points d'Arrêt & Marqueurs ou de la fenêtre Enregistrer & Analyser :

Par défaut, la profondeur de pile est définie sur trois images. La profondeur maximale pouvant être saisie est de 30 images.

La profondeur est relative au cadre de pile où un marqueur d'enregistrement est rencontré ; ainsi, lorsque l'enregistrement commence, si l'image de pile est 6 et la profondeur de pile est définie sur 3, le Débogueur enregistre les images 6 à 8.

Pour les situations où la pile est très grande, il est recommandé d'utiliser d'abord une faible profondeur de pile de 2 ou 3. À partir de là, vous pouvez augmenter progressivement la profondeur d'enregistrement de la pile et insérer des marqueurs d'enregistrement supplémentaires pour agrandir l'image jusqu'à ce que toutes les informations nécessaires soient affiché.

Placer des Marqueurs d'Enregistrement

Cette section explique comment placer des marqueurs d'enregistrement, qui vous permettent d'enregistrer silencieusement l'exécution de code entre deux points. L'enregistrement peut être utilisé pour générer un diagramme Séquence .

Comme ce processus enregistre l'exécution de plusieurs threads, il peut être particulièrement utile pour capturer des séquences événementielles (telles que les événements de souris et de minuterie).

Accéder

Ruban	Exécuter > Windows > Points d'Arrêt

Actions

Action
Différents marqueurs d'enregistrement peuvent être utilisés pour enregistrer le flux d'exécution ; consultez les liens connexes pour plus d'informations sur les propriétés et l'utilisation de ces marqueurs.
Gérez les points d'arrêt dans la fenêtre Point d'Arrêt et marqueurs.
Activer et désactiver les marqueurs.
Travailler avec Ensembles de Marqueurs - lorsque vous créez un point d'arrêt ou un marqueur, il est automatiquement ajouté à un ensemble de marqueurs, soit l'ensemble par défaut, soit un ensemble que vous créez dans un but spécifique.

Notes

- La *Gestion des Point d'Arrêt et Marqueurs*
  le sujet (Ingénierie de Logiciel) décrit une perspective différente

Coffret Marqueurs d'Enregistrement

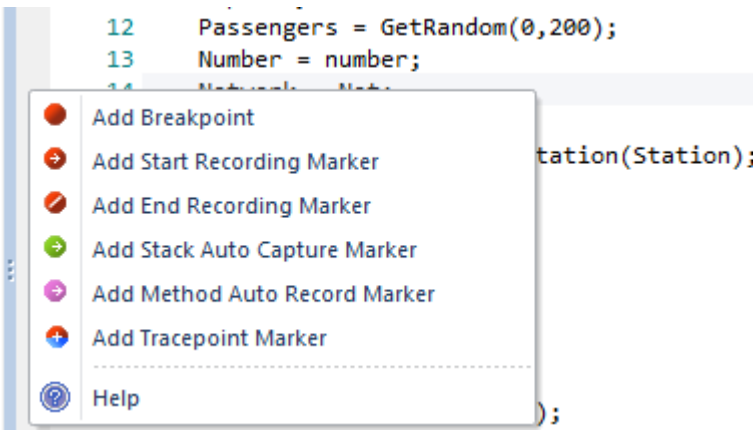
Les marqueurs sont définis dans l'éditeur de code source. Ils sont placés sur une ligne de code ; lorsque cette ligne de code s'exécute, l' Analyseur d'Exécution effectue l'action d'enregistrement appropriée au marqueur.

Accéder

Utilisez l'une des méthodes décrites ici pour afficher la fenêtre Éditeur de Code et charger le code source associé à la Classe ou à l'élément Classe sélectionné.

Ruban	Exécuter > Source > Modifier > Modifier la source de l'élément Exécuter > Source > Modifier > Fichier Open Source
Raccourcis Clavier	Sur un élément appuyez sur Ctrl+E ou F12 Pour afficher le navigateur « Fichier Open Source », appuyez sur Ctrl+Alt+O.

Définir un marqueur d'enregistrement

Étape	Action
1	Ouvrez le code source à déboguer, dans l'éditeur de code source intégré.
2	Trouvez la ligne de code appropriée et cliquez-droit dans la marge de gauche (Point d'Arrêt) pour faire apparaître le menu contextuel du point d'arrêt/marqueur ; sélectionnez le type de marqueur requis : 
3	Si un marqueur Démarrer Enregistrement a été défini, vous devez également définir un marqueur de fin Enregistrement .

Types de Marqueurs

Les marqueurs sont vraiment fantastiques. Inhabituels par leur très faible encombrement lorsqu'ils sont utilisés avec précaution, leur impact sur les performances des programmes en cours d'enregistrement peut être négligeable. Les marqueurs sont disponibles en plusieurs saveurs (enfin des couleurs en fait) et d'autres sont toujours ajoutés. Ils sont placés et visibles dans la marge gauche de l'éditeur, vous aurez donc besoin d'avoir du code source.

Avoir l'habitude de

- Enregistrer une seule fonction
- Enregistrer des parties d'une fonction
- Cas d'utilisation couvrant plusieurs fonctions
- Enregistrer les piles d'appels
- Générer diagrammes Séquence
- Générer diagrammes de domaines Test
- Générer diagrammes de collaboration de classes

Référence

Marqueur	Détail

Marqueurs Démarrer / fin Enregistrement	Placez les marqueurs sur les lignes de début et de fin du code à enregistrer. Il n'est pas nécessaire que ceux-ci relèvent de la même fonction.
	`17 private int m_delivery;`
	`18`
	`19 public ClassLib(){`
	`20`
	`21 }`
	`22`
	`23 /**`
	`24 *`
	`25 * @exception Throwable`
	`26 */`
	`27 public void finalize()`
	Lorsque le programme rencontre un marqueur de début d'enregistrement, un nouvel enregistrement est lancé (*la caméra commence à rouler !).* Lorsqu'un marqueur de fin est rencontré, l'enregistrement en cours se termine *(c'est une prise*). La manière dont vous utilisez ces marqueurs dépend de vous et de votre connaissance du système dont vous avez la garde.
	Trucs avancés (marqueurs imbriqués) :
	Si un marqueur Démarrer l'enregistrement est rencontré alors qu'un enregistrement est en cours, mais où *la capture est inhibée par la valeur de profondeur de pile utilisée* , un enregistrement séparé sera lancé. Chaque enregistrement est conservé sur une pile. Quand on termine, on le supprime. Cette technique peut être utilisée dans Enterprise Architect pour enregistrer et restituer des scènes dans des systèmes très complexes. Cela ressemble à assembler de courtes scènes d'une vidéo pour créer une bande-annonce. Si vous souhaitez uniquement enregistrer une seule

	fonction, vous devez utiliser un marqueur d'enregistrement automatique.

Méthode marqueur d'Auto-Enregistrement	Une Méthode marqueur d'Auto-Enregistrement permet d'enregistrer une fonction particulière. Le débogueur mettra automatiquement fin à l'enregistrement une fois la fonction terminée. C'est bien car l'enregistrement est une opération intensive.
	Le marqueur de fonction combine un marqueur Démarrer Enregistrement et un marqueur Fin Enregistrement en un seul, de sorte que l'enregistrement est exécuté après le point marqueur et s'arrête toujours lorsque cette fonction se termine.
	`185 ////////////////////////////////////////////////////////////////////////`
	`186 // CRecurrenceDlg message handlers`
	`187`
	`188 BOOL CRecurrenceDlg::OnInitDialog()`
	`189 {`
	`190  CBCGPDialog::OnInitDialog();`
	`191`
	`192  UINT nMask =`
	`193  CBCGPDateTimeCtrl::DTM_SPIN  |`
	`194  CBCGPDateTimeCtrl::DTM_DATE  |`
	`195  CBCGPDateTimeCtrl::DTM_TIME  |`
	`196  CBCGPDateTimeCtrl::DTM_CHECKBOX  |`
	`197  CBCGPDateTimeCtrl::DTM_DROPCALENDAR |`
	`198  CBCGPDateTimeCtrl::DTM_CHECKED;`
	`199`
	`200  UINT nFlags = CBCGPDateTimeCtrl::DTM_CHECKED | CBCGPDateTimeCtrl::DT`
	`201  //-------------------`
	`202  // Setup date fields:`
	Les marqueurs Enregistrement peuvent être imbriqués. Lorsqu'une nouvelle Méthode marqueur d'Auto-Enregistrement est atteinte pendant l'enregistrement, la profondeur de pile à laquelle enregistrer sera étendue pour inclure la méthode actuelle et la profondeur requise de cette fonction.

Piler marqueur de Auto-Capture	`76  /* End - EA generated code for Parts and Ports */`
	`77  /* Begin - EA generated code for  Activities and I`
	`78  public void ClassLib_ActivityGraphWithActionPin()`
	`79  {`
	Les marqueurs de pile vous permettent de capturer toutes les traces de pile uniques qui se produisent à un moment donné dans une application ; ils fournissent une image rapide et utile de l'endroit à partir duquel un point dans une application est appelé.
	Pour insérer un marqueur à l'endroit souhaité dans le code, cliquez-droit sur la ligne et sélectionnez l'option 'Ajouter un marqueur de capture automatique de pile'.
	Chaque fois que le débogueur rencontre le marqueur, il effectue une trace de pile ; si la trace de pile ne figure pas dans l'historique d'enregistrement, elle est copiée et l'application continue de s'exécuter.

Limiter la profondeur d'enregistrement	Vous pouvez limiter la profondeur des images dans n'importe quel enregistrement à l'aide du contrôle de profondeur de pile dans les barres d'outils de l'enregistreur et des points d'arrêt.

La Fenêtre Points d'Arrêt et Balises

À l'aide de la fenêtre Points d'Arrêt & Marqueurs, vous pouvez appliquer un contrôle à Analyse d'Exécution Visuelle lors de l'enregistrement de l'exécution pour générer diagrammes Séquence ; par exemple, vous pouvez :

- Activer, désactiver et supprimer des marqueurs
- Gérer les marqueurs en tant qu'ensembles
- Organisez la façon dont les marqueurs sont affichés, soit sous forme de liste, soit regroupés par fichier ou par classe.

Accéder

Ruban	Exécuter > Windows > Points d'Arrêt

Travailler avec Ensembles de Marqueurs

Les jeux de marqueurs vous permettent de créer des marqueurs sous forme de groupe nommé, que vous pouvez réappliquer à un fichier de code à des fins spécifiques.

Vous pouvez effectuer certaines opérations à partir de la fenêtre Points d'Arrêt & Markers uniquement, mais pour comprendre et utiliser les marqueurs et les jeux de marqueurs, vous devez également afficher le fichier de code approprié dans la ' Visionneuse de code source ' (cliquez sur l'élément Class et appuyez sur F12).

Accéder

Ruban	Exécuter > Windows > Points d'Arrêt : icône de la barre d'outils

Utilisant Ensembles des Marqueur

Action	Détails
Exemple d'utilisation	Vous pouvez créer un ensemble de marqueurs d'enregistrement automatique de méthode pour enregistrer l'action de diverses fonctions dans le code, ainsi qu'un ensemble de marqueurs de capture de pile pour enregistrer la séquence d'appels qui provoquent l'appel de ces fonctions. Vous pouvez ensuite créer diagrammes Séquence à partir des enregistrements sous chaque ensemble.
Créer un Ensemble de Marqueurs	Pour créer un jeu de marqueurs à partir de la fenêtre Points d'Arrêt & Marqueurs, cliquez sur la flèche déroulante de l'icône et sélectionnez l'option 'New Set'. La dialogue « Nouveau jeu de marqueurs Point d'Arrêt » s'affiche ; dans le champ « Entrer le nouveau nom de l'ensemble », saisissez un nom pour l'ensemble et cliquez sur le bouton Enregistrer. Le nom de l'ensemble s'affiche dans le champ de texte à gauche de l'icône ' Définir Options '. Alternativement, vous pouvez sélectionner l'option « Enregistrer sous l'ensemble » dans la liste déroulante « Définir Options » pour créer une copie exacte de l'ensemble actuellement sélectionné, que vous pourrez ensuite modifier.
Accéder Ensembles	Pour accéder à un jeu de marqueurs, cliquez sur la flèche déroulante dans le champ de texte à gauche de l'icône ' Définir Options ' et sélectionnez le jeu souhaité dans la liste. Les marqueurs de l'ensemble sont répertoriés dans la fenêtre Points d'Arrêt & Marqueurs. Vous chargeriez normalement un jeu de marqueurs avant le point auquel une action doit être capturée. Par exemple, pour enregistrer une séquence impliquant un dialogue particulier, lorsque vous commencerez le débogage, vous chargerez l'ensemble avant d'invoquer le dialogue ; une fois que vous ouvrez le dialogue dans l'application, les opérations que vous avez marquées sont enregistrées.

Ajouter Marqueurs à Ensemble	Pour ajouter des marqueurs à un jeu de marqueurs, ajoutez chaque marqueur requis à la ligne de code appropriée dans la ' Visionneuse de code source '. Le marqueur est immédiatement ajouté à l'ensemble actuellement répertorié dans la fenêtre Points d'Arrêt et marqueurs. Chaque marqueur répertorié dans le dialogue a une case à cocher dans la colonne « Activé » ; les marqueurs nouvellement ajoutés sont automatiquement activés, mais vous pouvez désactiver et réactiver les marqueurs rapidement lorsque vous vérifiez le code.
Stockage des Ensembles	Lorsque vous créez un jeu de marqueurs, il est immédiatement enregistré dans le modèle ; tout utilisateur utilisant le modèle a accès à cet ensemble. Cependant, l'ensemble par défaut, qui existe toujours pour un modèle, est un espace de travail personnel, n'est pas partagé et est stocké en dehors du modèle.
Supprimer un marqueur d'un ensemble	Cliquez-droit sur le marqueur et sélectionnez l'option 'Supprimer Point d'Arrêt '.
Supprimer un ensemble	Si vous n'avez plus besoin d'un jeu de marqueurs, accédez-y dans la fenêtre Points d'Arrêt & Marqueurs et sélectionnez l'option ' Supprimer Sélectionnée Set' dans la liste déroulante ' Définir Options '. Vous pouvez également effacer tous les jeux de marqueurs définis par l'utilisateur en sélectionnant l'option « Supprimer tous les jeux » ; une prompt s'affiche pour confirmer la suppression.

Notes

- Ensembles de marqueurs sont très simples et flexibles mais, comme ils peuvent être utilisés par tout utilisateur du modèle, ils peuvent être facilement corrompus ; considérez ces lignes directrices :
  - Lorsque vous nommez un ensemble, utilisez vos initiales dans le nom et essayez d'indiquer son utilisation, afin que l'autre modèle
  les utilisateurs peuvent reconnaître son propriétaire et son objectif
  - Lorsque vous utilisez un ensemble autre que Default, évitez les expérimentations excessives afin de ne pas ajouter beaucoup de marqueurs ad hoc à l'ensemble
  - Assurez-vous de savoir quel ensemble de marqueurs est exposé dans la fenêtre Points d'Arrêt et marqueurs.
  comme vous pouvez facilement ajouter par inadvertance des marqueurs à l'ensemble qui ne sont pas pertinents pour le fichier de code, le
  l'ensemble a été créé pour
  - Dans n'importe quel ensemble, si vous avez ajouté des marqueurs qui ne doivent pas être conservés, supprimez-les pour conserver le
  but de l'ensemble ; cela est particulièrement vrai pour l'ensemble par défaut, qui peut rapidement s'accumuler marqueurs ad hoc redondants

Contrôler la Session d'Enregistrement

La fenêtre Enregistrer et analyser vous permet de contrôler une session d'enregistrement. Le contrôle dispose d'une barre d'outils et d'une fenêtre d'historique qui affiche l'historique d'enregistrement au fur et à mesure de sa capture. Chaque entrée de cette fenêtre représente une séquence d'appels composée d'un ou plusieurs appels de fonction.

Accéder

Ouvrez la fenêtre Enregistrer et analyser en utilisant l'une des méthodes décrites ici.

Vous devez également ouvrir la fenêtre Analyseur d'Exécution ('Execute > Analyzer | Scripts d'Analyseur '), qui liste tous les scripts du modèle ; vous devez sélectionner et activer le script approprié pour l'enregistrement.

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur

Barre d'Outils d'Enregistreur

Vous pouvez accéder à facilités pour démarrer, arrêter et modérer une session d'enregistrement d'analyse d'exécution via la barre d'outils Enregistrer et analyser.

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur Explorer > Portails > Afficher la barre d'outils > Enregistrer

Boutons

Bouton	Description
	Affichez un menu d'options pour définir sur quoi fonctionne la session d'enregistrement. • Attacher au processus : activée même si aucun script Analyzer n'existe, cette option affiche une dialogue à travers laquelle vous sélectionnez un processus à enregistrer et une plate-forme de débogage à utiliser ; vous pouvez également éventuellement sélectionner un jeu de marqueurs d'enregistrement et/ou une Statemachine à utiliser pendant l'enregistrement • Générer Diagramme de Séquence à partir de Enregistrement - générer un diagramme Séquence / State à partir de la trace de l' Analyseur d'Exécution • Générer Testpoint Diagramme from History - générer un diagramme de domaine Test à partir de la trace de l' Analyseur d'Exécution , qui peut être utilisé avec Testpoint facilité • Générer Diagramme de classes à partir de l'historique - générer un diagramme de classes de collaboration à partir de la trace de l' Analyseur d'Exécution , décrivant uniquement les classes et opérations impliquées dans l'action enregistrée (cas d'utilisation) • Générer Graphique d'Appel à partir de l'Historique - générer un Graphique d'Appel dynamique à partir de l'historique d'enregistrement, comme vous pouvez le voir dans la disposition d'analyse d'exécution de l'espace de travail du profil VEA ; cela peut être plus utile que le diagramme Séquence pour identifier les piles d'appels uniques impliquées • Générer All - générer ensemble les diagrammes Séquence , Testpoint et Collaboration Class à partir de la trace Analyseur d'Exécution • Enregistrer en tant qu'artefact - créez un élément Artefact contenant l'historique d'enregistrement actuel, sous le Paquetage actuellement sélectionné dans la fenêtre du Navigateur ; si vous faites ensuite glisser cet élément d'artefact sur un diagramme de classes et double-cliquez dessus, l'historique enregistré dans l'artefact est recopié dans la fenêtre Enregistrer et analyser • Charger l'historique Séquence à partir d'un fichier : sélectionnez un fichier XML à partir duquel restaurer un historique d'enregistrement précédemment enregistré. • Enregistrer l'historique Séquence dans un fichier - enregistrez l'historique d'enregistrement dans un fichier XML

7	Sélectionnez la profondeur de la pile d'enregistrement pour le jeu de marqueurs ; c'est-à-dire le nombre d'images à partir du point auquel l'enregistrement a commencé.
	Lancez et enregistrez l'application décrite dans le script ; vous pouvez éventuellement sélectionner un jeu de marqueurs d'enregistrement et/ou une Statemachine à utiliser pendant l'enregistrement. L'icône est activée lorsque le script Analyzer actif est configuré pour le débogage.
	Effectuez un enregistrement manuel ad hoc du thread actuel pendant une session de débogage. Utilisez cette fonction avec les boutons 'step' du débogueur ; chaque fonction appelée en raison d'une commande d'étape est enregistrée dans la fenêtre d'historique. L'icône est activée si aucun enregistrement n'est en cours et que vous êtes actuellement à un point d'arrêt (c'est-à-dire en cours de débogage).
	Effectuez un enregistrement automatique ad hoc pendant une session de débogage. Lorsque vous cliquez sur cette icône, l'analyseur commence l'enregistrement et ne s'arrête que lorsque le programme se termine, que vous arrêtez le débogueur ou que vous cliquez sur l'icône Arrêter. Cette icône est activée si aucun enregistrement n'est en cours et que vous êtes actuellement à un point d'arrêt (c'est-à-dire en cours de débogage).
	Entrez dans une fonction, enregistrez l'appel de fonction dans la fenêtre Historique et revenez en arrière. Activé pour l'enregistrement manuel uniquement.
	Arrêtez d'enregistrer la trace d'exécution.
	Affichez la dialogue 'Synchronize Modèle ' à travers laquelle vous pouvez synchroniser le modèle avec les fichiers de code générés lors d'une opération d'enregistrement de profil.

Travailler avec Historique d'Enregistrement

Vous pouvez effectuer un certain nombre d'opérations sur ou à partir des résultats d'une session d'enregistrement, à l'aide du menu contextuel de la fenêtre Enregistrer et analyser.

Possibilités

Option	Action
Afficher la source de l'appelant	Afficher le code source, dans la Visionneuse de code source , de la méthode appelant la séquence.
Afficher la source de l'appelé	Afficher le code source, dans la Visionneuse de code source , de la méthode appelée par la séquence.
Générer Diagramme pour Séquence sélectionnée	Générer un diagramme Séquence pour une seule séquence sélectionnée dans l'historique d'enregistrement.
Générer Diagramme de Séquence	Générer un diagramme Séquence incluant toutes les séquences de l'historique d'enregistrement.
Effacer	Effacer l'historique d'enregistrement actuellement affiché dans la fenêtre Enregistrer et analyser.
Enregistrer Historique d'Enregistrement dans un fichier	Enregistrez l'historique des enregistrements dans un fichier XML. Une fenêtre de navigateur s'affiche, dans laquelle vous spécifiez le chemin d'accès et le nom du fichier XML.
Charger Historique d'Enregistrement à partir du fichier	Chargez un historique d'enregistrement précédemment enregistré à partir d'un fichier XML. Une fenêtre de navigateur s'affiche, dans laquelle vous spécifiez le chemin d'accès et le nom du fichier XML à charger.
Désactiver tous les appels	Désactivez tous les appels répertoriés dans la fenêtre Enregistrer et analyser.
Désactiver cet appel	Désactivez l'appel sélectionné.
Désactiver cette méthode	Désactivez la méthode sélectionnée.
Désactiver cette classe	Désactivez la classe sélectionnée.
Désactiver tous les appels en dehors de cet appel	Désactivez tous les appels répertoriés dans la fenêtre Enregistrer et analyser, à l'exception de l'appel sélectionné.
Activer tous les appels	Activez chaque appel répertorié dans la fenêtre Enregistrer et analyser.
Activer cet appel	Activez l'appel sélectionné.
Activer cette méthode	Activez la méthode sélectionnée.

Activer cette classe	Activez la classe sélectionnée.
Aide	Affichez la rubrique d'aide de la fenêtre Enregistrer et analyser.

Démarrer Enregistrement

Lorsque vous enregistrez le flux d'exécution sous forme de diagramme Séquence , vous démarrez l'enregistrement en sélectionnant l'icône ' Enregistrement ' dans la barre d'outils de la fenêtre Enregistrer et analyser. La dialogue « Enregistrer » s'affiche avec les options d'enregistrement définies par défaut ; c'est-à-dire le Point d'Arrêt et le jeu de marqueurs actuels, les filtres définis dans le script d'analyseur actuel et le mode d'enregistrement comme base.

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur :

Options Dialogue d'enregistrement

Champ/Bouton	Détail
Ensemble Enregistrement	Les marqueurs Enregistrement déterminent ce qui est enregistré. Si vous disposez d'un ensemble d'enregistrement à utiliser, cliquez sur la flèche déroulante et sélectionnez-le.
Filtres supplémentaires	Les filtres sont utilisés par le débogueur pour exclure les appels de fonction correspondants de l'historique d'enregistrement. Les filtres Enregistrement sont définis dans le script Analyzer. Dans le champ « Filtres supplémentaires », vous pouvez ajouter d'autres filtres pour cet exécuter spécifique. si vous spécifiez plusieurs filtres, séparez-les par un point-virgule.
Mode Enregistrement de base	En mode de base, le débogueur enregistre un historique des appels de fonction effectués par le programme chaque fois qu'il rencontre un marqueur d'enregistrement approprié.
Suivre les instances de classes nommées	En mode Suivi des instances, le débogueur capture également la création d'instances des classes que vous spécifiez. Il inclut ensuite ces informations dans l'historique. Le diagramme Séquence résultant peut alors montrer des lignes de vie pour chaque instance de cette classe avec, le cas échéant, des appels de fonction liés à la ligne de vie.
Suivre les transitions State	L'enregistrement peut également capturer les changements d' State à l'aide d'un diagramme Statemachine spécifié. Le diagramme Statemachine doit exister en tant qu'enfant d'une classe. L' Analyseur d'Exécution capture les instances de cette classe et calcule l' State de chaque instance chaque fois qu'une fonction de la séquence d'enregistrement en cours revient.
OK	Cliquez sur ce bouton pour démarrer le débogueur.

Parcourez Appels de Fonction

La commande « Pas à pas » peut être exécutée en cliquant sur le bouton Pas à pas dans la barre d'outils de la fenêtre Enregistrer et analyser.

Vous pouvez également appuyer sur Maj+F6 ou sélectionner l'option de ruban « Exécuter > Exécuter > Entrer ».

La commande 'Step Through' provoque l'exécution d'une commande 'Step Into' ; si une fonction est détectée, cet appel de fonction est enregistré dans la fenêtre Historique.

Le Débogueur sort alors et le processus peut être répété.

Ce bouton vous permet d'enregistrer un appel sans avoir à accéder à une fonction ; le bouton n'est activé qu'à un point d'arrêt et en mode d'enregistrement manuel.

Marqueurs d'Enregistrement Nested

Lorsqu'un marqueur d'enregistrement est rencontré pour la première fois, l'enregistrement commence à l'image de pile actuelle et se poursuit jusqu'à ce que l'image apparaisse, enregistrant des images supplémentaires jusqu'à la profondeur définie dans la barre d'outils Enregistrement . Considérez cette séquence d'appel :

A -> B -> C -> D -> E -> F -> G -> H -> I -> J -> K -> L -> M -> N -> O -> P -> Q - > R -> S -> T -> U -> V -> W -> X -> Y -> Z

Si vous définissez un marqueur d'enregistrement sur K et définissez la profondeur d'enregistrement sur 3, cela enregistrera la séquence d'appel :

K -> L -> M

Si vous souhaitiez également enregistrer les appels X, Y et Z dans le cadre du diagramme Séquence , vous placeriez un autre marqueur d'enregistrement en X et l'analyseur enregistrerait :

K -> L -> M -> X -> Y -> Z

Cependant, lorsque l'enregistrement se termine pour le composant XYZ (l'image X est sautée), l'enregistrement reprendra lorsque l'image M de la séquence KLM sera réintégrée. L'utilisation de cette technique peut être utile là où les informations du diagramme enregistré seraient exclues en raison de la profondeur de la pile, et elle vous permet de vous concentrer sur les zones particulières à capturer.

Génération Diagrammes Séquence

Cette rubrique décrit ce que vous pouvez faire avec l'enregistrement d'une session d'analyse d'exécution.

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur

Référence

Action	Détail
Générer un diagramme	Sélectionnez le Paquetage approprié dans la fenêtre Navigateur , dans lequel stocker le diagramme Séquence . Pour créer le diagramme à partir de toutes les séquences enregistrées, soit : • Cliquez sur l'icône ' Enregistreur Menu' ( ) dans la barre d'outils de la fenêtre Record & Analyse, et sélectionnez l'option ' Générer Diagramme de Séquence à partir Enregistrement ', ou • Cliquez-droit sur le corps de la fenêtre et sélectionnez l'option ' Générer Diagramme de Séquence ' Pour créer le diagramme à partir d'une seule séquence, soit : • Cliquez sur l'icône ' Enregistreur Menu' ( ) dans la barre d'outils de la fenêtre Record & Analyse, et sélectionnez l'option ' Générer Diagramme de Séquence à partir Enregistrement ', ou • Cliquez-droit sur la séquence et sélectionnez l'option ' Générer Diagramme à partir de Séquence Sélectionnée '
Enregistrer une séquence enregistrée dans un fichier XML	Cliquez sur la séquence, cliquez sur l'icône 'Menu Enregistreur ' ( ) dans la barre d'outils de la fenêtre Enregistrer et analyser, et sélectionnez l'option 'Enregistrer l'historique Séquence dans un fichier'.
Accéder à un fichier XML de séquence existant	Soit: • Cliquez sur le dans la barre d'outils de la fenêtre Enregistrer et analyser et sélectionnez l'option « Charger l'historique Séquence à partir d'un fichier », ou • Cliquez-droit sur une zone vide de l'écran et cliquez sur l'option 'Load Séquence From File' La dialogue « Windows Ouvrir » s'affiche, à partir de laquelle vous sélectionnez le fichier à ouvrir.

Avoir l'habitude de

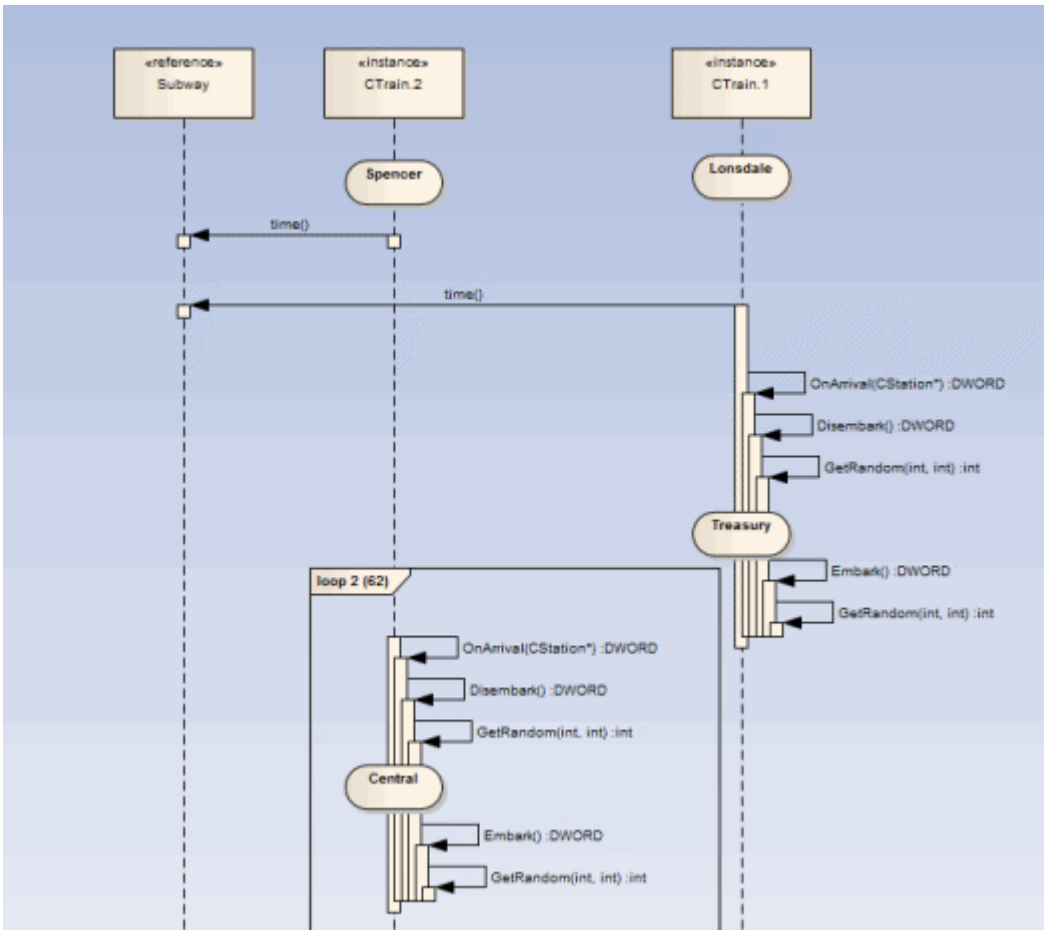
- Générer un diagramme Séquence à partir d'une session d'analyse d'exécution enregistrée, pour :
- toutes les séquences enregistrées ou
- une seule séquence dans la séance
- Enregistrer la séquence enregistrée dans un fichier
- Récupérez l'enregistrement enregistré et chargez-le dans la fenêtre Enregistrer et analyser

Transitions State Rapportage

Cette section décrit comment générer diagrammes Séquence qui montrent les transitions d'état lors de l'exécution d'un programme.

Avoir l'habitude de

Générer diagrammes Séquence qui rapportent les transitions d'état définies par l'utilisateur lors de l'exécution d'un programme (comme indiqué dans l'exemple diagramme généré)



Sujet
Créez une Statemachine sous la classe à signaler.
Définissez les contraintes pour chaque State pour définir le changement d'état à signaler.

Rapportage d'une Statemachine

L' Analyseur d'Exécution peut enregistrer un diagramme Séquence , nous le savons. Ce que vous ne savez peut-être pas, c'est qu'il peut utiliser une Statemachine en même temps pour détecter les transitions State qui pourraient survenir en cours de route. Ces States sont représentés à l'instant présent sur la bouée de sauvetage de l' object . Les transitions ressortent également des lignes de vie. Toute transition invalide ou illégale sera mise en évidence par une bordure rouge. Regarde.

Processus

Tout d'abord, vous modélisez une Statemachine pour l'élément Class approprié.

Vous composez ensuite les expressions qui définissent chaque State à l'aide de l'onglet 'Contraintes' de chaque State .

Ces expressions simples sont formées à l'aide des noms d'attributs du modèle de classe et de la base de code réelle. Ce ne sont pas des déclarations OCL. Chaque expression doit apparaître sur une ligne distincte.

m_strColor == "Bleu"

Vous utilisez ensuite la fenêtre Enregistreur pour lancer le débogueur.

Le bouton Exécuter de la fenêtre Enregistreur est différent du bouton des autres barres d'outils du débogueur.

La fenêtre Enregistreur vous permettra de rechercher une Statemachine si vous ne connaissez pas le nom de la Statemachine . La dialogue « Transition State » présente une liste de Statemachines pour l'ensemble du modèle, dans laquelle vous localisez et sélectionnez le diagramme approprié (voir l'exemple).

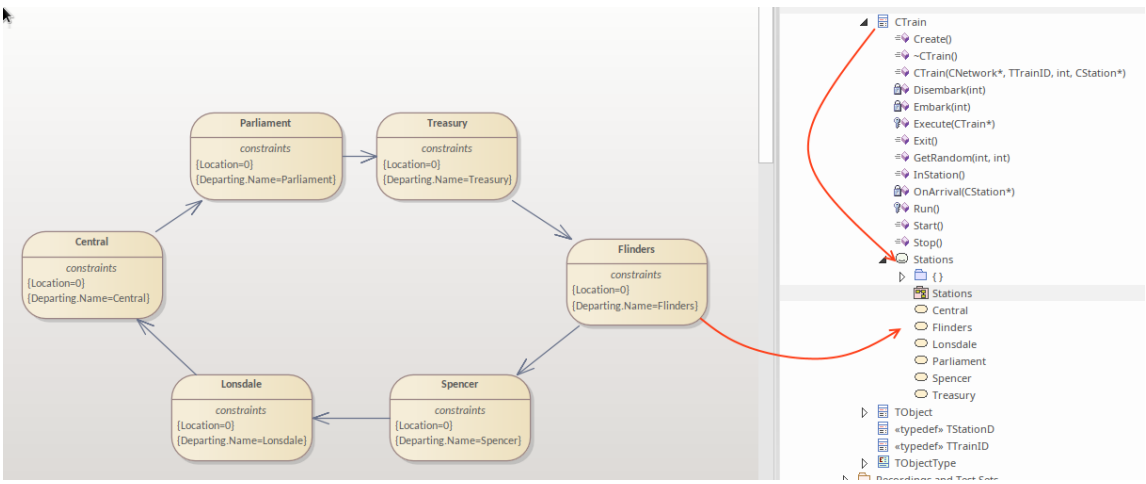
Lorsque vous générez le diagramme Séquence , il représente non seulement la séquence mais aussi les changements d' State aux différents points de la séquence ; chaque instance de classe participant au processus de détection est affichée avec sa propre bouée de sauvetage.

Exemple

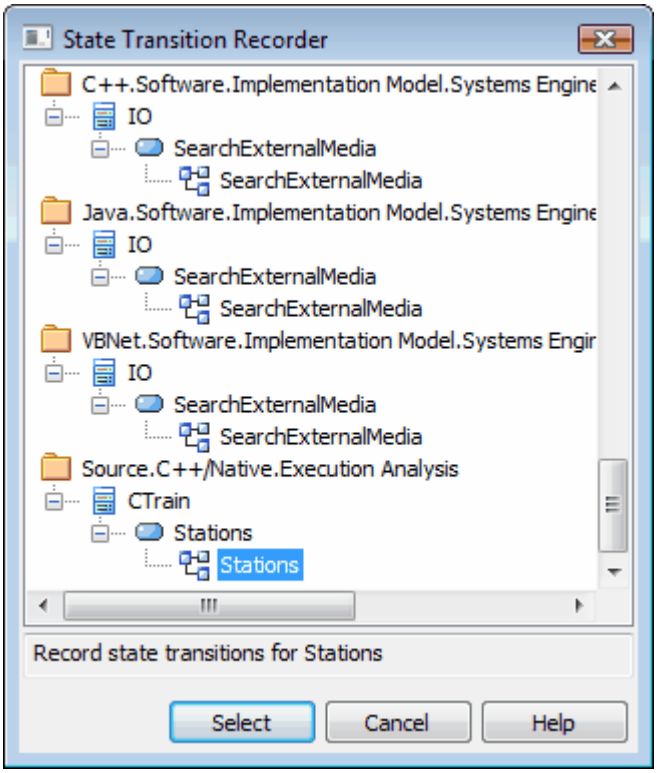
La Stations Statemachine montre les différents States du système de métro Melbourne Underground Loop.

Un train circulant sur le réseau de métro peut être arrêté dans n'importe quelle station représentée sur la Statemachine .

La Stations Statemachine est un enfant de la classe CTrain.



Lorsque vous recherchez le diagramme dans la dialogue ' State Transition Enregistreur ', la hiérarchie affiche uniquement le Paquetage racine, la classe parent, la sous-machine enfant et diagramme ; aucun autre composant du modèle n'est répertorié.


State Transition Recorder
C++.Software.Implementation Model.Systems Engine
IO
SearchExternalMedia
SearchExternalMedia
Java.Software.Implementation Model.Systems Engine
IO
SearchExternalMedia
SearchExternalMedia
VBNet.Software.Implementation Model.Systems Engir
IO
SearchExternalMedia
SearchExternalMedia
Source.C++/Native.Execution Analysis
CTrain
Stations
Stations
Record state transitions for Stations
Select
Cancel
Help

Enregistrement et Mappage Changements de State

Cette rubrique explique comment définir des contraintes sur chaque State de la Statemachine sous une classe, pour définir le changement d'état à enregistrer.

Exemple

Cet exemple de dialogue State « Propriétés » s'adresse à l' State appelé Parlement ; l'onglet 'Contraintes' est ouvert pour montrer comment l' State est lié à la Classe CXTrain.

Un State peut être défini par une seule contrainte ou par plusieurs ; dans l'exemple, l' State Parlement a deux contraintes :

Les valeurs des contraintes ne peuvent être comparées que pour les types élémentaire, enum et string

La classe CXTrain a un membre appelé Location de type int et un membre appelé Departing.Name de type CString ; cette contrainte signifie que cet State est évalué à True lorsque :

- une instance de la classe CXTrain existe et
- sa variable membre Location a la valeur 0 et
- la variable membre Departing.Name a la valeur Parlement

Opérateurs dans les contraintes

Il existe deux types d'opérateurs que vous pouvez utiliser sur les contraintes pour définir un State :

- Les opérateurs logiques AND et OR peuvent être utilisés pour combiner des contraintes
- Les opérateurs d'équivalence {= et != } peuvent être utilisés pour définir les conditions d'une contrainte

Toutes les contraintes pour un State sont soumises à une opération AND sauf indication contraire ; vous pouvez utiliser l'opération OR sur eux à la place, vous pouvez donc réécrire les contraintes dans l'exemple comme :

Emplacement=0 OR

Emplacement=1 AND

Départ.Nom!=Central

Voici quelques exemples d'utilisation des opérateurs d'équivalence :

Départ.Nom!=Central AND

Emplacement !=1

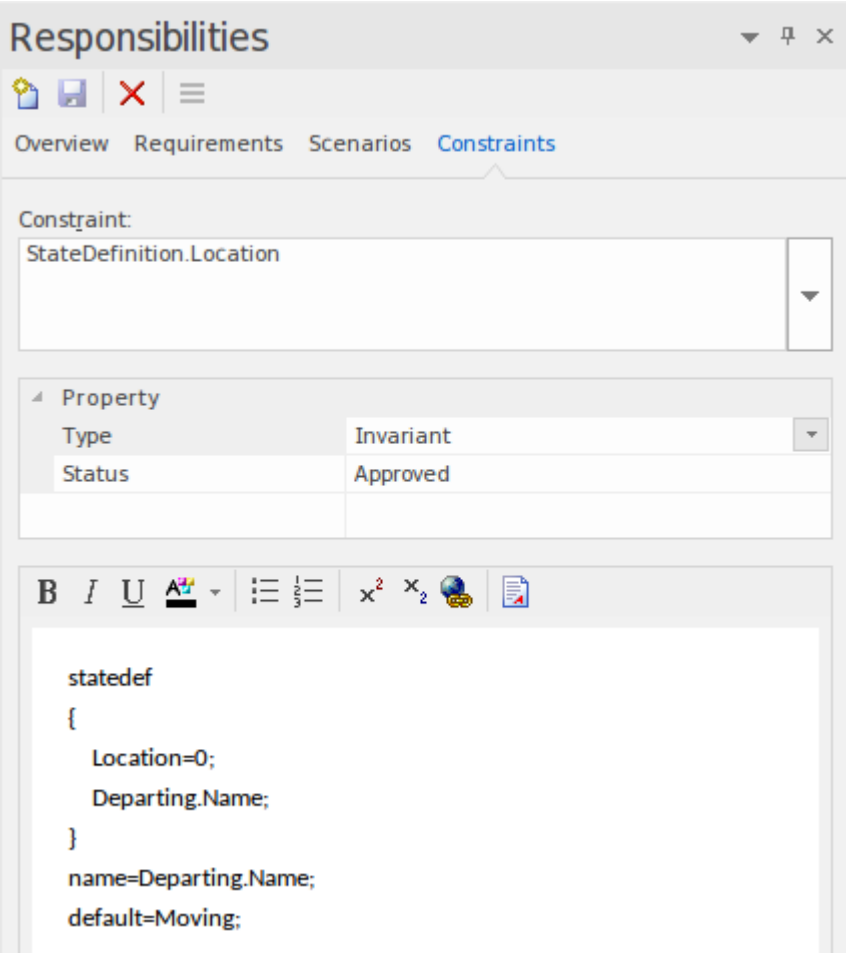
Notes

- Les guillemets autour des chaînes sont facultatifs ; la comparaison des chaînes est toujours sensible à la casse pour déterminer la vérité d'une contrainte

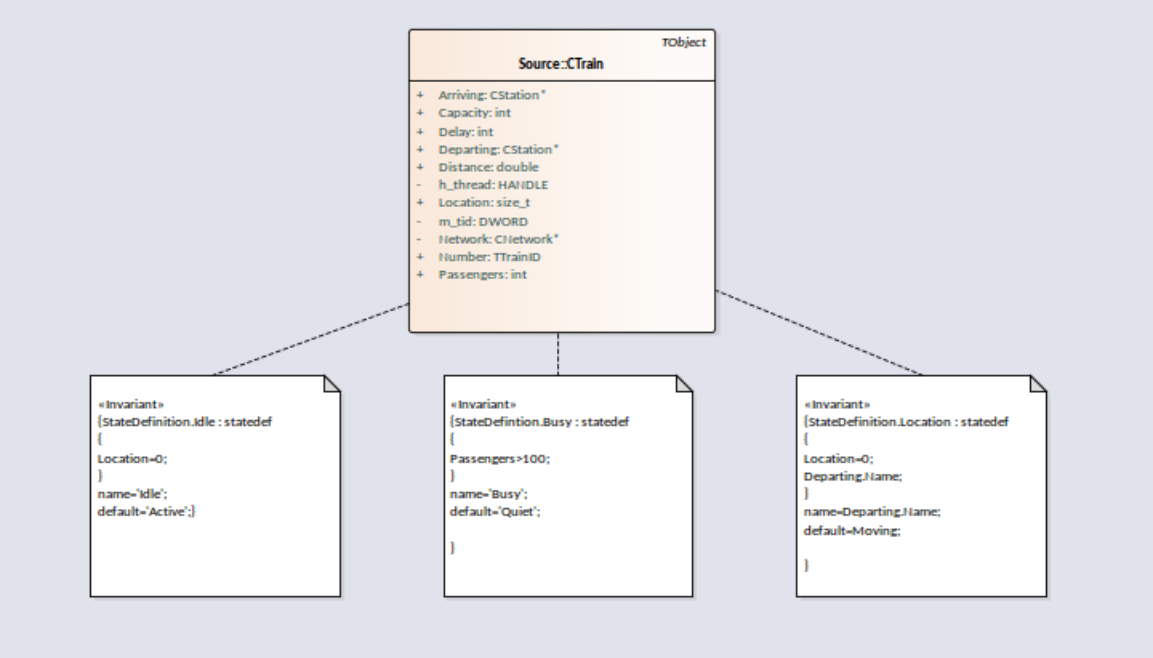
Analyseur de State

L' Analyseur de State est une fonctionnalité qui permet d'analyser, de détecter et d'enregistrer les états des instances d'une classe. La fonctionnalité fonctionne en combinant une définition d'état (définie sur une classe comme contrainte) et des marqueurs appelés points State . Il est disponible pour tous les langages pris en charge par l' Analyseur d'Exécution , notamment Microsoft.NET, Mono, Java et le C++ natif.

Nous commençons par sélectionner une classe et composer notre définition d'état.



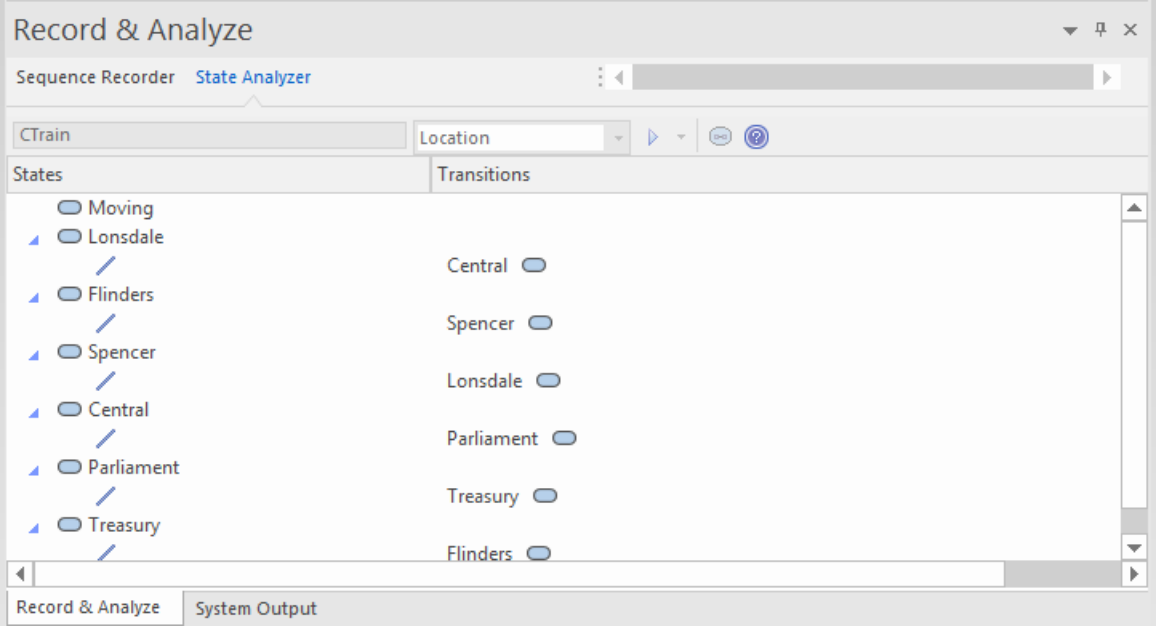
Nous pouvons obtenir une image de toutes les définitions d'état que nous avons définies en plaçant la classe sur un diagramme et en établissant un lien vers les notes de classe qui sont elles-mêmes liées à une contrainte de définition d'état particulière. Nous expliquons comment procéder dans une section ultérieure.



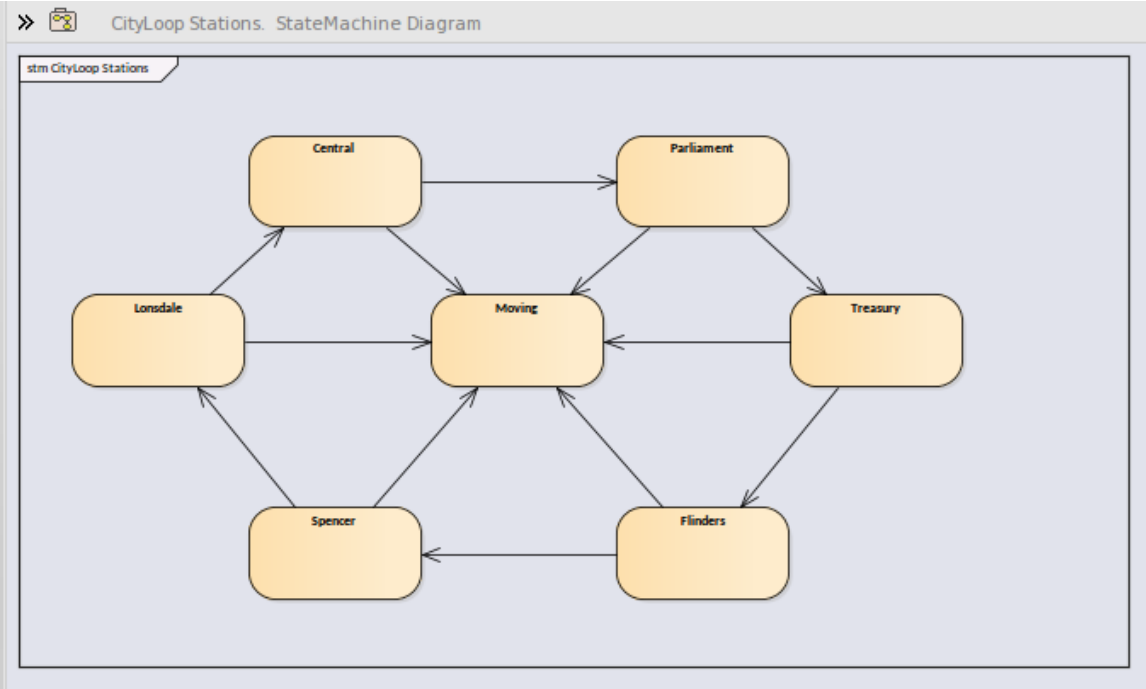
Les points State sont définis en plaçant un ou plusieurs marqueurs dans le code source pertinent.

```
73 DWORD   CTrain::OnArrival( CStation* S)
74 {
75     Departing = S;
76     Location = 0;
77     Delay = (Disembark(GetRandom()) + Embark(GetRandom()));
78     DWORD ScheduleTime = Network->TimeAtStation(Departing);
79     if(Delay > (int)ScheduleTime)
80         return Delay;
81     return ScheduleTime;
82 }
```

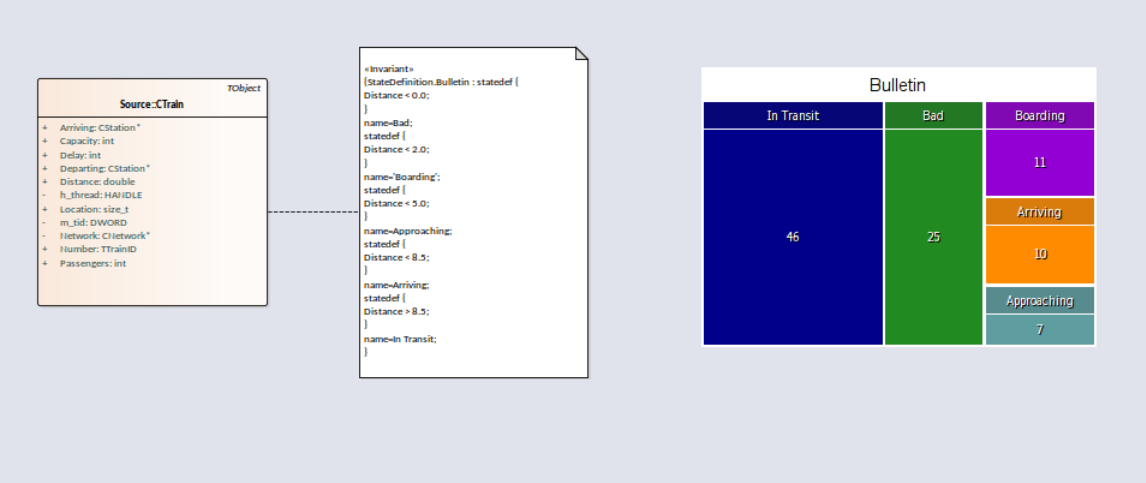
Le programme à analyser est exécuter à l'aide du contrôle Analyseur de State . Lorsque l' Analyseur d'Exécution rencontre un point State , l'instance actuelle de la Classe est analysée. Lorsque le domaine valeur de l'instance correspond à la définition de l'état, un état est enregistré. Chaque fois que l'instance varie, de nouveaux états sont ainsi détectés. Le contrôle répertorie chaque état au fur et à mesure de sa découverte. Sous chaque état, le contrôle répertorie l'ensemble discret des transitions vers d'autres états effectuées par les instances de la classe.



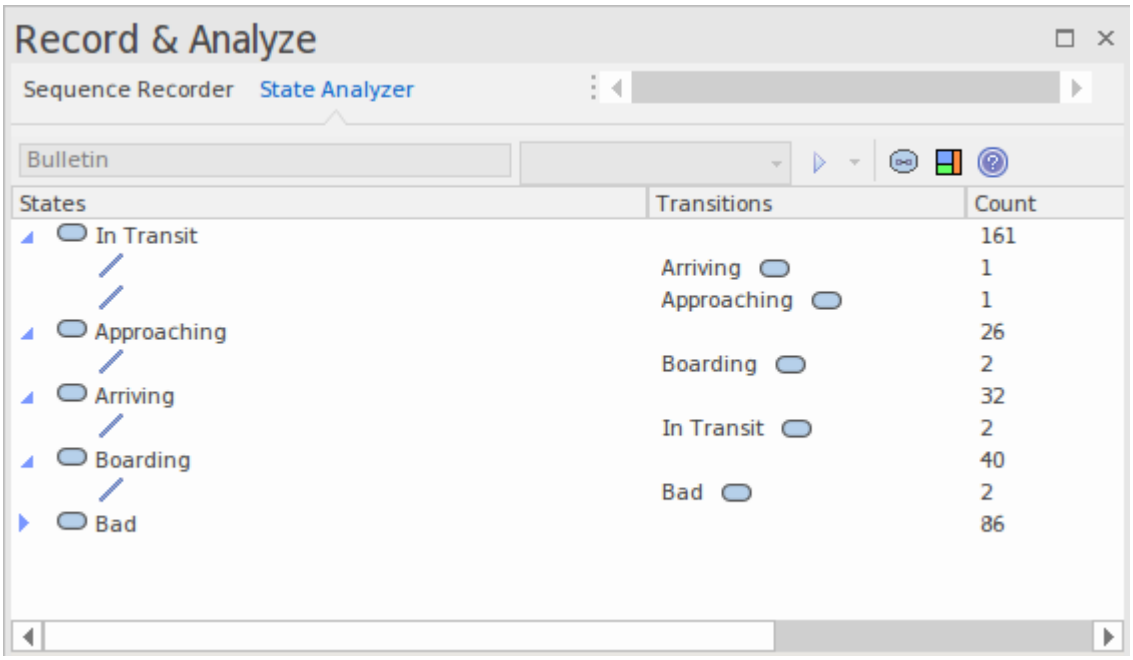
Les informations peuvent être utilisées pour créer un Statemachine .



En utilisant les mêmes informations, nous pouvons facilement produire une Carte de Chaleur . Cet exemple montre une classe « Train », sa définition State « Bulletin » (sous forme de note liée) et la Carte de Chaleur qu'elle a produite. Les chiffres sur la carte sont des pourcentages. Sur la carte, nous pouvons observer que les trains étaient à l'état « En transit » 46 % du temps.



Il s'agit de l'analyse de la définition State du « Bulletin » qui a produit notre Carte de Chaleur .



Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur > Analyseur de State Conception > Élément > Editeurs > Contraintes

Définitions de State

Définitions de State sont composées dans les propriétés Contraintes d'un élément Class. Le type de contrainte doit être nommé *StateDefinition.name* , où « nom » est votre choix de titre pour la définition. Ces titres sont répertoriés dans la combo de l' Analyseur de State chaque fois qu'une Classe est sélectionnée. Vous sélectionnez une seule définition dans cette zone de liste déroulante avant d'exécuter le programme. La définition State dans notre exemple est nommée « StateDefinition.Location ». Il définit les états en fonction de l'emplacement des instances de la classe CTrain.

Définitions de State sont composées d'une ou plusieurs spécifications. Chaque spécification d'état commence par le mot

clé « statedef » qui est ensuite suivi d'une ou plusieurs instructions. Les instructions définissent les contraintes qui décrivent l'état, et éventuellement une variable dont valeur peut être utilisée pour nommer l'état. Les instructions sont placées entre accolades et se terminent par un point-virgule comme indiqué :

```
statedef {
Location=0;
Départ.Nom ;
}
```

Nommer les états à l'aide de variables

Dans cet exemple, « Location » est une constante et « Departing.name » est une variable. Une instruction supplémentaire suit les contraintes et indique le nom de l' State à attribuer à partir de la variable valeur . Voici la définition avec la directive de dénomination.

```
statedef {
Location=0;
Départ.Nom ;
}
name=Departing.Name;
```

Nommer les états à l'aide de littéraux

Dans cet exemple, la définition State ne contient que des constantes et l'état est nommé à l'aide d'un littéral.

```
statedef {
Location=100;
}
nom='Central';
```

Une définition State unique définissant plusieurs spécifications State .

```
statedef {
Passagers > 100 ;
}
nom=Occupé ;
statedef {
Passagers >= 50 ;
}
nom = Calme ;
statedef {
Passagers < 50 ;
}
nom=Très silencieux ;
statedef {
Passagers = 0 ;
```

}
nom = Inactif ;

State par défaut

Une définition State peut spécifier un état « fourre-tout » par défaut qui décrira l'état d'une instance lorsqu'aucun autre état n'est vrai. Vous définissez un état par défaut pour la définition avec une instruction ressemblant à ceci :

statedef {
Location=0;
Départ.Nom ;
}
name=Departing.Name;
par défaut = Déplacement ;

Dans cet exemple, pendant l'exécution, toute instance détectée ayant un attribut « Emplacement » non nul sera enregistrée comme étant dans l'état « En mouvement ».

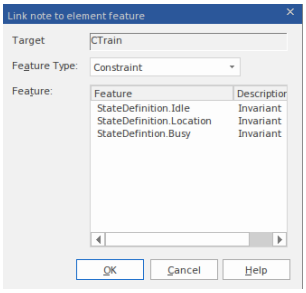
Vous pouvez choisir d'exclure l'enregistrement de l'état par défaut en désactivant l'option 'Inclure l'état par défaut' dans le menu déroulant de la barre d'outils de l' Analyseur de State . Cela exclurait l'enregistrement des transitions vers un état « par défaut ».

Création Notes sur un élément de classe affichant Définitions de State

Cette section décrit comment créer le diagramme de classe qui montre toutes les Définitions de State définies pour la classe.

Actions

Afficher un diagramme de classes	Ouvrez un diagramme de classes existant ou créez-en un nouveau.
Créer un lien vers l'élément Class	Faites glisser la classe qui vous intéresse sur le diagramme sous forme de lien.
Créer un élément note	Créez un élément note sur le diagramme et liez-le à la classe.
Liez la note à la définition State	Sélectionnez le lien entre la Note et la Classe et, à l'aide de son menu contextuel, sélectionnez l'option 'Lier Note à Fonctionnalité de l'Élément'.
Choisissez la définition à	Dans la dialogue de l'élément, choisissez « Contraintes » dans la liste déroulante.

afficher sur la Note  	Toutes Définitions de State définies seront répertoriées parmi lesquelles vous pourrez choisir.
Répéter	Répétez la procédure pour toutes les autres Définitions de State de la classe.

Synchronisation

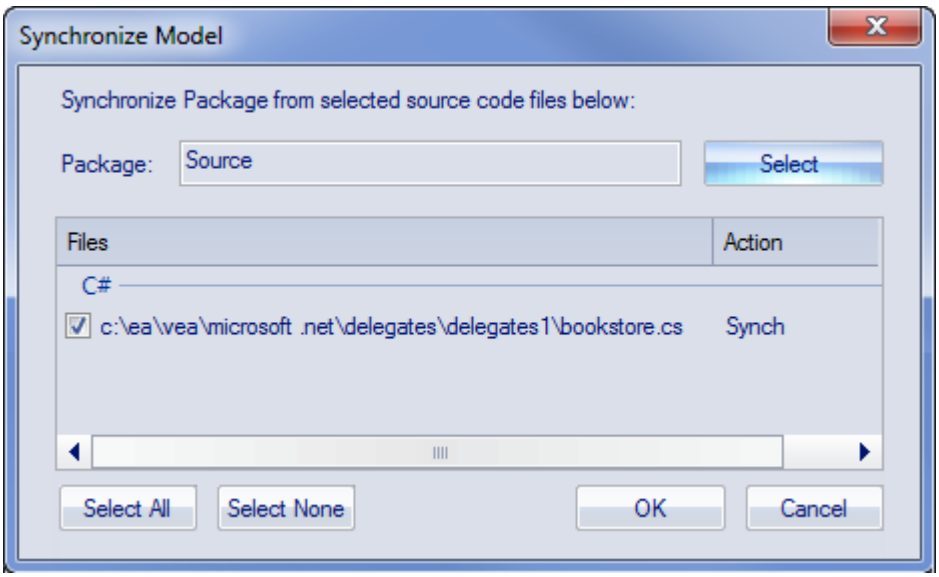
L'enregistrement produit un certain nombre d'actifs, l'historique d'enregistrement étant le principal. Enregistrement identifie également un ensemble de fichiers de code source. Cet ensemble peut être utilisé pour produire diagrammes de classes et de domaines Test , mais peut également être utilisé pour synchroniser votre modèle.

Un modèle synchronisé permet une navigation rapide et précise entre les éléments diagramme et le modèle de classe.

Accéder

Ruban	Exécuter > Outils > Enregistreur > Ouvrir Enregistreur > Bouton de la barre d'outils
Menu Contexte	Cliquez-droit sur la fenêtre Enregistrer & Analyser | Synchroniser Modèle avec Code Source

Synchroniser Modèle



Champ/Bouton	Action
Paquetage	Cliquez sur le bouton Sélectionner et sélectionnez le Paquetage cible dans lequel procéder à l'ingénierie inverse des fichiers de code.
Fichiers/ Action	Répertorie les fichiers identifiés lors d'un ou plusieurs enregistrement(s). L'action appropriée est répertoriée à côté de chaque fichier.
Tout sélectionner	Cliquez sur ce bouton pour cocher la case en regard de chaque fichier de la liste « Fichiers ».
Ne rien sélectionner	Cliquez sur ce bouton pour décocher la case en regard de chaque fichier de la liste

	« Fichiers ».
OK	Cliquez sur ce bouton pour lancer l'opération. La progression de la synchronisation sera affichée.
Annuler	Cliquez sur ce bouton pour abandonner la synchronisation et fermer le dialogue .