

Utilisation de déclencheurs de base de données avec agent mixte pour effectuer un traitement spécifique à une application

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Aperçu](#)

[Dialing List et Personal Callback List](#)

[Déclencheurs d'échantillon](#)

[Insérez une ligne dans une base de données externe](#)

[L'information de contact de mise à jour dans une base de données externe](#)

[Supprimez une ligne d'une base de données externe](#)

[Exécutez un exécutable externe d'un déclencheur](#)

[Considérations de représentation](#)

[Méthodes recommandées](#)

[Informations connexes](#)

Introduction

Ce livre blanc décrit comment employer des déclencheurs de base de données SQL Server pour exécuter le traitement spécifique à l'application, quand les enregistrements de contact dans la base de données privée d'agent mixte (BA) sont fermés ou sont mis à jour par le responsable de campagne de BA.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Intelligent Contact Management de Cisco (missile aux performances améliorées)
- Agent mixte de logiciel ICM de Cisco
- Base de données SQL de Microsoft

Composants utilisés

Les informations en ce livre blanc sont basées sur des ces logiciel et versions de matériel :

- Version 4.6.x et ultérieures missile aux performances améliorées de Cisco
- Microsoft SQL 7

Les informations en ce livre blanc ont été créées des périphériques dans un environnement de travaux pratiques spécifique. Tous les périphériques utilisés en ce livre blanc commencé par une configuration (par défaut) effacée. Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à [Conventions relatives aux conseils techniques Cisco](#).

Aperçu

Le BA enregistre les enregistrements de contact sont commuté une base de données de Microsoft SQL Server. L'utilisation d'une application de base de données standard, telle que le Serveur SQL, le facilite pour que des tiers et des utilisateurs intègrent leurs systèmes et applications avec du BA.

Ce livre blanc décrit comment employer des déclencheurs de base de données SQL Server pour exécuter le traitement spécifique à l'application, quand le responsable de campagne de BA en clôture ou met à jour l'état de incluent l'article dans la base de données.

Dialing List et Personal Callback List

Le BA enregistre les enregistrements de contact dans la table de **Dialing_List**. Il enregistre les enregistrements personnels de rappel dans la table de **Personal_Callback_List**. Ces tables sont dans la base de données privée de BA (**customer_inst_baA**). Dans cette table, CallStatus et CallResult identifient l'état d'un enregistrement de contact.

Nom	Type	Taille	Description
Call Status	CAR	1 octet	CallStatus indique l'état de l'enregistrement : p — L'enregistrement de contact est EN SUSPENS pour s'appeler. c — L'enregistrement a été fermé. Référez-vous à ces documents pour une description complète de toutes les valeurs de CallStatus et de leurs significations : <ul style="list-style-type: none">• Agent mixte de logiciel ICM de Cisco installé et guide de configuration (version 4.6.x missile aux performances améliorées de Cisco)• Guide utilisateur d'agent mixte de logiciel ICM de Cisco (version

			missile aux performances améliorées de Cisco 5)
Call Result	SMA LLIN T	2 octets	<p>CallResult indique le résultat de l'appel comme retourné au responsable de campagne de BA par le numéroteur de BA. Ainsi, il indique si la VOIX d'affichages de numéroteur, OCCUPÉ, ou PAS DE RÉPONSE quand le numéro de contact est composé. Référez-vous à ces documents pour des descriptions complètes de toutes les valeurs de CallResult et de leurs significations :</p> <ul style="list-style-type: none"> • Agent mixte de logiciel ICM de Cisco installé et guide de configuration (version 4.6.x missile aux performances améliorées de Cisco) • Guide utilisateur d'agent mixte de logiciel ICM de Cisco (version missile aux performances améliorées de Cisco 5)

Toutes les fois que le responsable de campagne de BA met un enregistrement à jour dans le **Dialing_List** ou dans le **Personal_Callback_List**, les champs de CallStatus et de CallResult sont mis à jour. Pour cette raison, il est simple de créer des déclencheurs de base de données SQL Server pour surveiller ces champs.

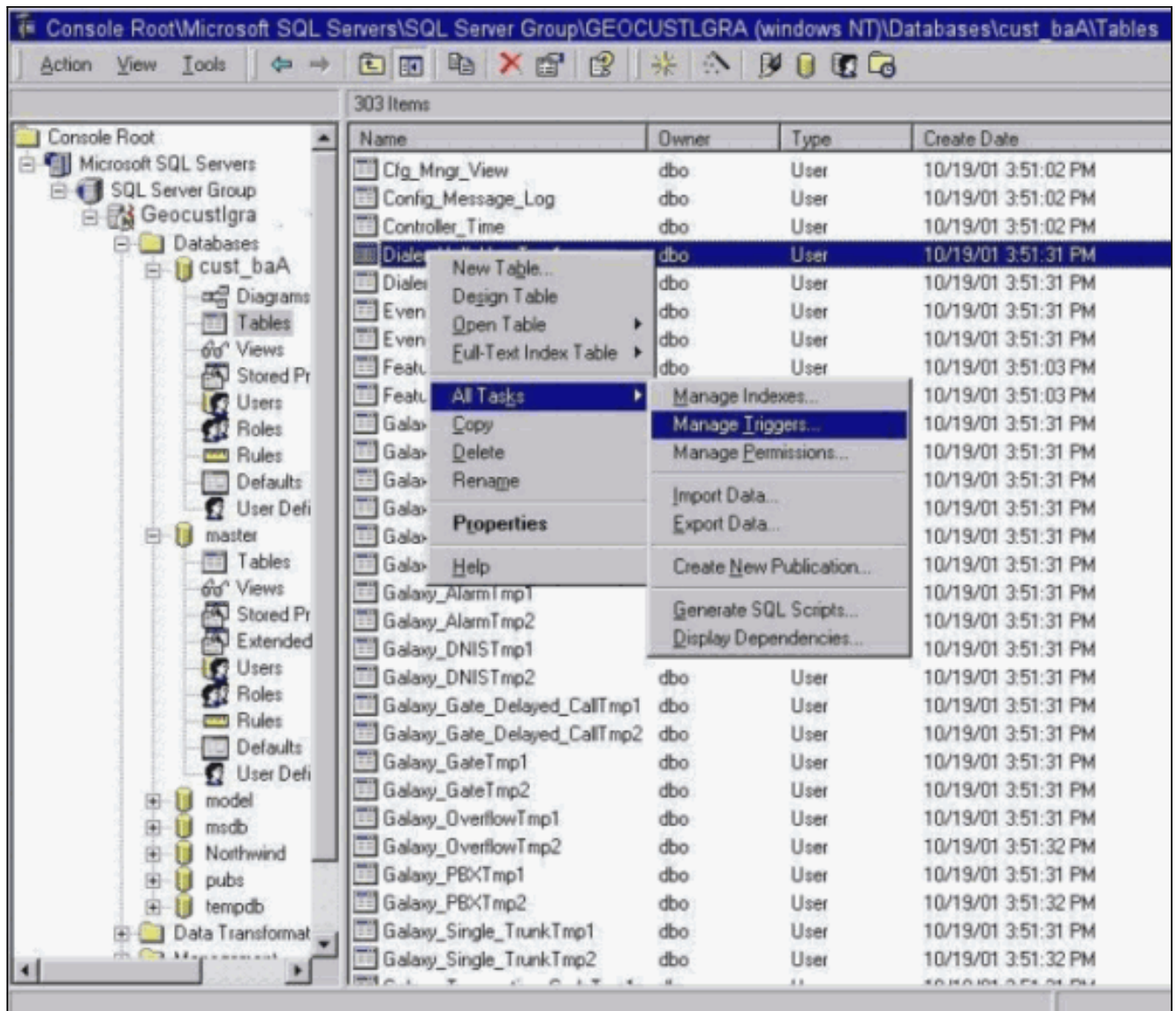
L'utilisation des déclencheurs sur les tables de base de données de BA te permet pour atteindre un nombre illimité d'objectifs spécifiques à l'application. Ce livre blanc explique quelques exemples de scénario où les déclencheurs peuvent être utiles.

[Déclencheurs d'échantillon](#)

Remarque: Tous les déclencheurs d'échantillon en ce livre blanc se rapportent à une table factice, le **CRM_Table** dans la base de données **CRM_DB**.

Terminez-vous ces étapes pour créer des déclencheurs du gestionnaire d'entreprise de Serveur SQL :

1. Approfondissement à la base de données de BA.
2. Sous la base de données, cliquez sur l'icône de **Tableau**. Ceci affiche les tables qui sont associées avec la base de données.
3. Sélectionnez la table sur laquelle vous voulez créer un déclencheur.
4. Cliquez avec le bouton droit la table sélectionnée et sélectionnez **toutes les tâches > gèrent des déclencheurs**.



[Insérez une ligne dans une base de données externe](#)

Le déclencheur d'échantillon dans cette section insère une ligne dans la table, **CRM_Table** (dans la base de données **CRM_DB**) toutes les fois que le responsable de campagne de BA met un enregistrement à jour de contact dans le **Dialing_List**.

CRÉEZ l'insert_crm_data de DÉCLENCHEUR

```
CREATE TRIGGER insert_crm_data ON Dialing_List FOR
UPDATE AS
/* Proceed only if EXACTLY ONE Record is updated */
!--- See next Note for an explanation. IF (SELECT COUNT
(*) FROM INSERTED) = 1 BEGIN INSERT CRM_DB..CRM_Table
SELECT inserted.AccountNumber, GETDATE (),
inserted.CallStatus FROM inserted END
```

Remarque: Autre que des enregistrements de contact de mise à jour quand les contacts s'appellent, le responsable de campagne exécute également quelques mises à jour en vrac. Ceux-ci devraient être filtrés par le déclencheur. Voyez la section de [méthodes recommandées de](#) ce pour en savoir plus de livre blanc.

[L'information de contact de mise à jour dans une base de données externe](#)

Le déclencheur d'échantillon dans cette section met un enregistrement à jour dans la table **CRM_Table** (dans la base de données **CRM_DB**) par une correspondance du numéro de compte du contact. Par exemple, si le responsable de campagne de BA met l'enregistrement à jour dans **Dialing_List** avec le numéro de compte 1A2B3C4, puis l'enregistrement dans **CRM_Table** avec le même numéro de compte est mis à jour simultanément.

CRÉEZ l'updt_customer_by_acctnum de DÉCLENCHEUR

```
CREATE TRIGGER updt_customer_by_acctnum ON Dialing_List
FOR UPDATE AS
/* Proceed only if EXACTLY ONE Record is updated */
IF (SELECT COUNT (*) FROM INSERTED) = 1
BEGIN

    DECLARE @CS CHAR (1)
    DECLARE @ACCTNUM VARCHAR (25)
    SELECT
        @CS = inserted.CallStatus,
        @ACCTNUM = inserted.AccountNumber
    FROM inserted
    UPDATE Northwind..CRM_Table
    SET Status = @CS
    Where Northwind..CRM_Table.AccountNumber = @ACCTNUM

END
```

[Supprimez une ligne d'une base de données externe](#)

Le déclencheur d'échantillon dans cette section supprime un enregistrement de la table **CRM_Table** (dans la base de données **CRM_DB**) par une correspondance du numéro de compte du contact, si cet enregistrement de contact a été fermé dans le **Dialing_List**. Par exemple, si le responsable de campagne de BA ferme l'enregistrement dans **Dialing_List** avec le numéro de compte 1A2B3C4, puis l'enregistrement dans **CRM_Table** avec le même numéro de compte est supprimé.

CRÉEZ le del_customer_by_acctnum de DÉCLENCHEUR

```
CREATE TRIGGER del_customer_by_acctnum ON Dialing_List
FOR UPDATE AS
/* Proceed only if EXACTLY ONE Record is updated */
IF (SELECT COUNT (*) FROM INSERTED) = 1
BEGIN

    DECLARE @CS CHAR (1)
    DECLARE @ACCTNUM VARCHAR (25)
    SELECT
        @CS = inserted.CallStatus,
        @ACCTNUM = inserted.AccountNumber
    FROM inserted
    IF @CS = 'C'
    BEGIN
        DELETE Northwind..CRM_Table
        WHERE Northwind..CRM_Table.AccountNumber =
@ACCTNUM
    END

END
```

Exécutez un exécutable externe d'un déclencheur

Le déclencheur d'échantillon dans cette section lance le programme **C:\MYAPP.EXE**. Le déclencheur passe l'AccountNumber et le CallStatus de l'enregistrement mis à jour comme arguments de la ligne de commande à **MYAPP.EXE**. Ceci se produit toutes les fois que le responsable de campagne de BA met à jour le statut de exactement un enregistrement.

CRÉEZ LE DÉCLENCHEUR run_executable

```
CREATE TRIGGER run_executable ON Dialing_List FOR UPDATE
AS
/* Proceed only if EXACTLY ONE Record is updated */
IF (SELECT COUNT (*) FROM INSERTED) = 1
BEGIN
    DELCARE *CMD VARCHAR (100)
    SELECT
        @CMD = 'C:\MYAPP.EXE ' + inserted.AccountNumber + '
' + inserted.CallStatus
    FROM inserted
    EXEC master ..xp_cmdshell @CMD
END
```

Remarque: Dans la table précédente, plutôt qu'appellent la procédure stockée de **xp_cmdshell** pour commencer un exécutable, le déclencheur peut également exécuter n'importe quelle autre procédure stockée.

Considérations de représentation

- Des déclencheurs de Serveur SQL sont exécutés synchroniquement. Ceci signifie s'il y a un déclencheur de mise à jour sur une table, une commande de mise à jour sur cette table ne retourne pas jusqu'à ce que l'exécution de déclencheur soit de finition. Si les déclencheurs ne sont pas de façon optimale conçus, alors les mises à jour de base de données de BA sont ralenties. En outre, le responsable de campagne de BA pourrait rencontrer des délais d'attente de la Connectivité de base de données ouverte (ODBC) tandis qu'il exécute des commandes SQL pour mettre à jour les tables. En règle générale, un déclencheur devrait faire le traitement possible minimum et devrait être optimisé pour la vitesse d'exécution.
- Le résultat de l'exécution de déclencheur pourrait affecter le résultat de l'exécution d'origine SQL. Par exemple, supposez qu'un déclencheur de mise à jour est défini sur **Dialing_List**. Si le déclencheur de mise à jour échoue, alors l'exécution de mise à jour sur **Dialing_List** pourrait également échouer et l'enregistrement en question peut être laissé dans **Dialing_List** avec un CallStatus incorrect.
- Si un déclencheur est imbriqué, alors l'exécution SQL ne se termine pas jusqu'à ce que tous les déclencheurs imbriqués soient exécutés. Par conséquent, la panne de n'importe quel déclencheur pourrait échouer l'exécution de l'original SQL.
- Si un déclencheur exécute une application externe ou une procédure stockée différente, alors l'exécution de déclencheur n'est pas complète jusqu'à l'application externe ou la procédure stockée est de finition.
- L'utilisation des curseurs dans les déclencheurs n'est pas recommandée, en raison de l'incidence potentiellement négative sur la représentation.
- Les déclencheurs ne devraient pas effectuer la CPU ou les tâches mémoire-intensives.
- Tous les déclencheurs devraient être bien testés et soumis à une contrainte avant déploiement.

Méthodes recommandées

- Vérifiez toujours que seulement un enregistrement est mis à jour avant que vous poursuiviez. Ceci empêche le déclencheur d'agir sur les mises à jour en vrac faites par le responsable de campagne. Ce sont des exemples des mises à jour en vrac qui sont généralement filtrées par le déclencheur : Un changement du CallStatus d'un enregistrement d'**A** (ACTIF) à **U** (UNKNOWN), quand un numéroteur avec l'ACTIVE enregistre des démonter du responsable de campagne. Le responsable de campagne met périodiquement des enregistrements à jour dans **U** CallStatus à **P** (EN ATTENDANT). **Remarque:** Quand un enregistrement de contact est mis à jour en raison d'un appel, les enregistrements sont toujours mis à jour un par un. Par conséquent, la vérification pour voir si exactement un enregistrement était les sauvegardes mises à jour contre des actions non désirées de déclencheur le plus souvent. Une exception dans ce cas se produit quand des mises à jour d'une exécution en vrac seulement une ligne.
- Vérifiez la valeur de CallStatus avant que vous poursuiviez. Par exemple, quand le responsable de campagne envoie un enregistrement au numéroteur de BA, le responsable de campagne place le CallStatus à **A** (ACTIF). Un déclencheur peut vérifier CallStatus. Si CallStatus mis à jour est **A**, il ne poursuit pas pour promouvoir.
- Afin d'optimiser la représentation pendant les mises à jour de base de données de BA, le temps et les exécutions ressource-intensives ne doivent pas être exécutés dans le déclencheur. Vous pouvez créer une table spécifique à l'application dans une autre base de données pour servir de file d'attente. Ceci accélère l'exécution de déclencheur. Vous pouvez employer le déclencheur pour insérer, mettre à jour, ou supprimer des données du fait spécifique à l'application ajoutez. Puis, une autre application peut voter cette table pour des modifications et effectuer d'autres tâches comme nécessaires.

Informations connexes

- [Installation et guide de configuration d'agent mixte de logiciel ICM de Cisco](#)
- [Guide utilisateur d'agent mixte de logiciel ICM de Cisco](#)
- [Support et documentation techniques - Cisco Systems](#)