



Catalyseur d'intelligence marketing

## Webhooks

Guide d'utilisation

8 Nov. 2017



Canada • France • Russie

[dialoginsight.com](http://dialoginsight.com)

## Table des matières

Bonnes pratiques .....	2
Le choix des abonnements .....	2
Le traitement des notifications.....	2
Tentatives d'envois .....	3
Erreurs.....	3
Désactivation automatique .....	3
Alertes courriels .....	4
Informations sur les notifications .....	4
Type de notifications supportées.....	5
Messages .....	5
Erreurs de livraison.....	5
Erreurs de production .....	6
Contacts.....	8
Création.....	8
Mise à jour.....	9
Abonnement .....	10
Désabonnement.....	11
Quarantaine .....	12
Gérer les notifications reçues .....	15



## Bonnes pratiques

### Le choix des abonnements

Il existe plusieurs abonnements possibles dans Dialog Insight et chacun d'eux peut conduire à un très grand nombre d'envois de notification. Il est important de bien choisir les données vous avez besoin afin de ne vous abonner qu'aux notifications pertinentes, et non à toutes.

### Le traitement des notifications

Une fois un Webhook activé, Dialog Insight va envoyer toutes les modifications demandées. Selon le nombre d'interactions effectuées par les contacts, ou selon le nombre d'envois en erreur, la quantité de notifications envoyées peut être très élevée. Il n'est donc pas recommandé de traiter ces notifications au fur et à mesure où elles arrivent car toute erreur ou délai de traitement peut vous faire perdre des notifications.

L'objectif de votre système qui reçoit les appels de webhooks devrait plutôt se limiter à *accepter la responsabilité d'une notification le plus rapidement possible et de confirmer que la notification a été reçue.*

Nous recommandons donc un système asynchrone qui enregistre directement toutes les notifications que Dialog Insight envoie, dans une base de données ou dans un fichier. Ensuite, un autre processus automatisé pourra traiter une à une les notifications enregistrées afin de produire le résultat voulu, soit l'ajout de contacts dans votre CRM, la mise à jour d'un contact selon vos règles, etc.



## Tentatives d'envois

Une notification est envoyée le plus rapidement possible lorsqu'une action ciblée se produit dans la plateforme Dialog Insight via un POST (http / https). Ce POST expire après un délai de 15 secondes. Et tant que nous ne recevons pas de réponse HTTP dont le statut est «OK» (code = 200), nous réessayerons quatre autres fois pendant les deux prochaines heures, selon ce qui suit :

- 1<sup>er</sup> essai : immédiat
- 2<sup>e</sup> essai : après 1 minute
- 3<sup>ème</sup> essai : après 15 minutes (donc 16 minutes après que l'action se soit produite)
- 4<sup>me</sup> essai : après 30 minutes (donc 46 minutes après que l'action se soit produite)
- 5<sup>ème</sup> essai : après 60 minutes (donc 106 minutes après que l'action se soit produite)

## Erreurs

Si la notification n'a pas pu être livrée après ces tentatives, elle est mise en erreur et abandonnée.

## Désactivation automatique

Les notifications se désactivent automatiquement si votre système tombe en panne, pour éviter un grand volume de messages bloqués en essais multiples.

Votre système sera déclaré *en panne* lorsqu'au moins 100 erreurs consécutives se seront produites, et ce sur une période minimale de 8 heures.

Ce webhook sera alors désactivé et une alerte transmise aux utilisateurs configurés pour recevoir les alertes associées à ce webhook.

Pour réactiver le flot des notifications, vous devrez les activer à nouveau via la plateforme. Notez que les notifications qui auraient eu lieu pendant la période de désactivation seront perdues (de même que les notifications qui étaient en attente lors de la désactivation).



## Alertes courriels

Des alertes sont envoyées par courriel dans les situations suivantes :

- À la 1<sup>e</sup>, 10<sup>e</sup>, 100<sup>e</sup>, 1000<sup>e</sup>, 10 000<sup>e</sup> erreur de chaque journée (basé sur l'heure des serveurs Dialog Insight).
- Lorsque votre webhook se fait désactiver automatiquement.

## Informations sur les notifications

Toutes les notifications sont envoyées au format JSON. Ces notifications commencent toujours par les paramètres l'identifiant :

- « *type* » : contient une chaîne de caractères représentant le type de notification envoyée. Ce paramètre peut donc contenir différentes valeurs, comme:
  - « *sending\_Bounce* » pour une erreur de livraison
  - « *sending\_ProductionError* » pour une erreur de production
  - « *contact\_modified* » pour une modification de contact
  - « *contact\_optout* » pour un désabonnement de contact
  - « *contact\_complaint* » pour une plainte spam formulée par un contact
  - Etc. (voir la documentation pour connaître la liste exacte)
- « *EventUniqueID* » contient une chaîne de caractères qui constitue l'identifiant unique de cette notification
- « *dtExecution* » date et heure exactes de l'envoi de la notification
- « *idCompany* » contient l'identifiant de la compagnie
- « *idProject* » contient l'identifiant du projet dans lequel la notification est configurée
- « *isTest* » est présent et contient « *true* » s'il s'agit d'une notification de test effectué depuis les interfaces de configurations des webhooks, et n'est pas présent dans le cas où il s'agit d'un vrai envoi de notification.
- « *ContactID* » contient un dictionnaire de valeurs identifiant le contact, soit :
  - « *idContact* », l'identifiant du contact dans la base de données de Dialog Insight.

Ainsi que l'ensemble des champs faisant partie de la clé primaire dans le projet.

Notez que votre système doit honorer le paramètre *isTest* de façon appropriée : si ce paramètre contient « *true* », vous ne devez pas tenter de désactiver un contact réel ou de modifier son profil lors de la réception d'un message test.



## Type de notifications supportées

### Messages

Les notifications d'erreur de message contiennent les paramètres inclus dans toutes les notifications du système, ainsi que :

- les informations sur l'envoi en erreur (dans le dictionnaire «*SendLog*») :
  - « *EventType* » contient l'un des deux types de l'envoi en erreur :
    - « *Batch* » pour les envois en lot ou les campagnes
    - « *Event* » pour les envois unitaires (confirmations, Web Services, tests)
  - « *idSendLog* » contient l'identifiant de l'envoi en erreur
  - « *idMessage* » contient l'identifiant du message utilisé pour l'envoi
  - « *idMessageCategory* » contient l'identifiant du type de communication du message
  - « *dtProcessed* » contient la date et l'heure à laquelle l'envoi est entré en production

### Erreurs de livraison

Permet d'être avisé lorsqu'une erreur de livraison de message se produit, y compris lorsqu'un contact est mis en quarantaine.

Ces notifications contiennent les paramètres de bases de l'identifiant, ainsi que des informations sur l'envoi et la livraison en erreur (dans le dictionnaire « *DeliveryErrorInfo* » :

- « *dsnMTA* » contient le nom de domaine du serveur MX avec lequel l'échange a eu lieu
- « *dsnDiag* » contient le code d'erreur obtenu du serveur, qui a causé la mise en quarantaine
- « *BounceCode* » contient le code d'erreur SMTP
- « *isInvalidMailbox* » indique si l'adresse courriel est invalide ou non.
- « *BounceType* » contient le type d'erreur « *BounceBack* » ou « *Bounce* »



Exemple de notifications envoyées :

```
[
  {
    "type": "sending_Bounce",
    "EventUniqueID": "77cb9126-661a-43b9-9915-1c8f9e826f93",
    "dtExecution": "2016.09.19 10:54:49-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "SendLog": {
      "EventType": "Batch",
      "idSendLog": 1,
      "idMessage": 1,
      "idMessageCategory": 456,
      "dtProcessed": "2016.09.19 10:54:49-04:00"
    },
    "DeliveryErrorInfo": {
      "dsnMTA": "mail.exemple.com",
      "dsnDiag": "smtp;550 5.1.1 The email account that you
      tried to reach does not exist.",
      "BounceCode": "5.1.1",
      "isInvalidMailbox": false,
      "BounceType": "BounceBack"
    }
  }
]
```

## Erreurs de production

Permet d'être avisé lorsqu'une erreur se produit lors de la préparation d'un message, avant même sa livraison.

Ces notifications contiennent les paramètres de bases l'identifiant, ainsi que des informations sur l'envoi et la production en erreur (dans le dictionnaire « *ProductionErrorInfo* ») :

- « *ErrorCode* » contient un nombre entier représentant le code d'erreur ci-dessous :
- « *ErrorMessage* » contient la description de l'erreur



ErrorCode	Erreur	ErrorMessage
<i>Erreur du système</i>		
0	NoError	No error
1	RecipientEmailSyntax	Recipient email syntax is invalid.
2	Compilation	There was a compilation error in the message.
3	Execution	There was an execution error in the message.
4	SenderEmailSyntax	Sender email syntax is invalid.
5	Submission	An unknown internal error occurred.
6	RequiredDataMissing	An unknown internal error occurred.
7	EligibilityError	The contact was not eligible to receive the message.
99	Undocumented	An unknown internal error occurred.
<i>Erreur provoquée par le client</i>		
100	Generic	An unknown internal error occurred.
101	MissingData	An unknown internal error occurred.
102	InvalidData	An unknown internal error occurred.
103	NoContent	An unknown internal error occurred.

Exemple de notifications envoyées :

```
[
  {
    "type": "sending_ProductionError",
    "EventUniqueID": "941a8d77-13f0-48f8-be49-9056f71d266d",
    "dtExecution": "2016.09.19 11:23:29-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "SendLog": {
      "EventType": "Batch",
      "idSendLog": 1,
      "idMessage": 1,
      "idMessageCategory": 6098,
      "dtProcessed": "2016.09.19 11:23:29-04:00"
    },
    "ProductionErrorInfo": {
      "ErrorCode": "NoContent",
      "ErrorMessage": "An unknown internal error occurred."
    }
  }
]
```





## Contacts

### Création

Permet d'être avisé lorsqu'un nouveau contact est créé dans Dialog Insight, soit manuellement, par formulaire. Aucun avis n'est envoyé pour les contacts créés par import.

Les notifications sur la création d'un contact contiennent les paramètres inclus dans toutes les notifications du système, et tous les champs du projet avec leurs valeurs (dans le dictionnaire « ContactData »). Ce dernier contient les valeurs de tous les champs du projet créés par le client (où chaque champ aura un code préfixé par «f\_»), ainsi que tous les champs administratifs (source de création du contact, date de création, date d'activation, etc.). Voici un exemple de notifications envoyées :

```
[
  {
    "type": "contact_created",
    "EventUniqueID": "b221df6e-3d8a-485f-84f0-c9ff8e259f45",
    "dtExecution": "2016.09.19 11:24:15-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "ContactData": {
      "f_FirstName": "FirstName",
      "f_LastName": "LastName",
      "dtActivated": null,
      "dtDeactivated": null,
      "SourceIndex_Creation": null,
      "idSource_Creation": null,
      "idSource_Modification": null,
      "idSource_Activation": null,
      "idSource_Deactivation": null,
      "isActive": null,
      "clKey": "xxxxxx",
      "dtCreated": null,
      "dtModified": null,
      "dtInvalidMailbox": null,
      "isInvalidMailbox": null,
      "Domain": null
    }
  }
]
```



## Mise à jour

Permet d'être avisé chaque fois qu'un contact est mis à jour dans Dialog Insight, soit manuellement, par formulaire. Aucun avis n'est envoyé pour les contacts modifiés par import.

Tout comme la création d'un contact, les notifications sur la modification d'un contact contiennent les paramètres inclus dans toutes les notifications du système, ainsi que tous les champs du projet avec leurs valeurs après modification du contact (dans le dictionnaire « ContactData »).

Exemple de notifications envoyées :

```
[
  {
    "type": "contact_modified",
    "EventUniqueID": "2cfd919-5c96-4ecf-8fbb-0e0eefa4196e",
    "dtExecution": "2016.09.19 11:24:40-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_Email": "EMail",
      "idContact": 1
    },
    "ContactData": {
      "f_FirstName": "FirstName",
      "f_LastName": "LastName",
      "dtActivated": null,
      "dtDeactivated": null,
      "SourceIndex_Creation": null,
      "idSource_Creation": null,
      "idSource_Modification": null,
      "idSource_Activation": null,
      "idSource_Deactivation": null,
      "isActive": null,
      "clKey": "xxxxxx",
      "dtCreated": null,
      "dtModified": null,
      "dtInvalidMailbox": null,
      "isInvalidMailbox": null,
      "Domain": null
    }
  }
]
```



## Abonnement

Permet d'être avisé lorsqu'un contact s'abonne à une communication et que son état passe à Abonné, ou à Actif si le projet ne comporte pas de centre de consentements (Loi C-28).

Les notifications sur l'abonnement d'un contact contiennent les paramètres inclut dans toutes les notifications du système, la valeur du champ *isActive*, ainsi que la liste des champs d'abonnements. Cette liste « OptinInfo » contient 2 objets :

- « OptinField » contient le code du champ d'abonnement.
- « OptinDate » contient la date à laquelle le contact a été abonné

Exemple de notifications envoyées :

```
[
  {
    "type": "contact_optin",
    "EventUniqueID": "408922e7-4e8b-456b-a1a0-2956be9a3a69",
    "dtExecution": "2016.09.22 11:00:19-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "OptinInfo": {
      "Fields": [
        {
          "OptinField": "infolettre",
          "OptinDate": "2016.09.22 11:00:19-04:00"
        },
        {
          "OptinField": "infolettre2",
          "OptinDate": "2016.09.22 11:00:19-04:00"
        }
      ]
    },
    "isActive": true
  }
]
```



## Désabonnement

Permet d'être avisé lorsqu'un contact se désabonne d'une communication et que son état passe à Désabonné, ou à Inactif si le projet ne comporte pas de centre de consentements (Loi C-28).

Les notifications sur le désabonnement d'un contact contiennent les paramètres inclut dans toutes les notifications du système, la valeur du champ isActive, ainsi que la liste des champs d'abonnements. Cette liste « OptinInfo » contient 2 objets :

- « OptinField » contient le code du champ d'abonnement.
- « OptinDate » contient la date à laquelle le contact a été désabonné

Exemple de notifications envoyées :

```
[
  {
    "type": "contact_optout",
    "EventUniqueID": "e92268e6-0b0c-4062-8b59-09b445133cd7",
    "dtExecution": "2016.09.22 11:30:30-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "OptinInfo": {
      "Fields": [
        {
          "OptinField": "infolettre",
          "OptinDate": "2016.09.22 11:30:30-04:00"
        },
        {
          "OptinField": "infolettre2",
          "OptinDate": "2016.09.22 11:30:30-04:00"
        }
      ]
    },
    "isActive": false
  }
]
```



## Quarantaine

Permet d'être avisé lorsque l'adresse d'un contact est mise en quarantaine suite à une tentative de livraison de message.

Les notifications sur la mise en quarantaine d'un contact contiennent les paramètres inclut dans toutes les notifications du système, ainsi que toute l'information disponible sur cette quarantaine (dans le dictionnaire « QuarantineInfo ») :

- « idSendLog » contient l'identifiant de l'envoi qui a conduit à une mise en quarantaine
- « idMessage » contient l'identifiant du message utilisé pour l'envoi
- « dtProcessed » contient la date et l'heure à laquelle l'envoi est entré en production
- « dtBounce » contient la date et l'heure à laquelle le message a retourné l'erreur de quarantaine
- « dtDelivered » contient la date et l'heure à laquelle le message a été livré
- « dsnMTA » contient le nom de domaine du serveur MX avec lequel l'échange a eu lieu
- « dsnDiag » contient le code d'erreur obtenu du serveur, qui a causé la mise en quarantaine

Exemple de notifications envoyées :

```
[
  {
    "type": "contact_quarantine",
    "EventUniqueID": "de0e65c1-fe4d-43ef-b8b9-9a4b217d4ff6",
    "dtExecution": "2016.09.19 11:25:52-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "QuarantineInfo": {
      "idSendLog": 1,
      "idMessage": 1,
      "dtProcessed": "2016.09.19 11:15:52-04:00",
      "dtBounce": "2016.09.19 11:25:52-04:00",
      "dtDelivered": null,
```



```

    "dsnMTA": "mail.exemple.com",
    "dsnDiag": "smtp;550 5.1.1 The email account that you
    tried to reach does not exist."
  }
}
]

```

## Plainte de courrier indésirable

Permet d'être avisé lorsqu'un contact signale un message comme courrier indésirable.

Les notifications sur la plainte de courrier indésirable d'un contact contiennent les paramètres inclus dans toutes les notifications du système, ainsi que l'information principale sur la plainte : le rapport du *feedbackLoop* contenu dans le paramètre «FBL\_Report».

Exemple de notifications envoyées :

```

[
  {
    "type": "contact_complaint",
    "EventUniqueID": "25673f56-8222-4af4-b563-1e967304cf61",
    "dtExecution": "2016.09.19 11:26:11-04:00",
    "idCompany": 1,
    "idProject": 1234,
    "isTest": true,
    "ContactID": {
      "f_EMail": "EMail",
      "idContact": 1
    },
    "ComplaintInfo": {
      "FBL_Report": "\r\nFrom: <abusedesk@example.com>\r\nDate:
      Thu, 8 Mar 2005 17:40:36 EDT\r\nSubject: FW: Earn money\r\nTo:
      <abuse@example.net>\r\nMIME-Version: 1.0\r\nContent-Type:
      multipart/report; report-type=feedback-report;\r\n
      boundary=\"part1_13d.2e68ed54_boundary\"\r\n \r\n--
      part1_13d.2e68ed54_boundary\r\nContent-Type: text/plain;
      charset=\"US-ASCII\"\r\nContent-Transfer-Encoding: 7bit\r\n
      \r\nThis is an email abuse report for an email message received
      from IP\r\n10.67.41.167 on Thu, 8 Mar 2005 14:00:00 EDT. For
      more information\r\nabout this format please see
      http://www.mipassoc.org/arf/.\r\n \r\n--
      part1_13d.2e68ed54_boundary\r\nContent-Type: message/feedback-
      report\r\n \r\nFeedback-Type: abuse\r\nUser-Agent:
      SomeGenerator/1.0\r\nVersion: 1\r\nOriginal-Mail-From:
      <somespammer@example.net>\r\nOriginal-Rcpt-To:
    }
  }
]

```



[illegible]

## Gérer les notifications reçues

Il est conseillé d'accepter toutes notifications envoyées par Dialog Insight afin de les traiter dans un second temps. Voici de exemple de code C# permettant d'accepter ces notifications dans un fichier .ashx :

```
<%@ WebHandler Language="C#" Class="WebHookHandlerExample" %>
```

```
using System;
using System.Web;
using System.IO;
using System.Text;

public class WebHookHandlerExample : IHttpHandler
{
    public void ProcessRequest (HttpContext context)
    {
        // Read the posted JSON body
        string json;
        using (StreamReader stream = new
            System.IO.StreamReader(context.Request.InputStream, Encoding.UTF8))
        {
            json = stream.ReadToEnd();
        }

        // Do not process event contents here : queue the event to be
        // processed asynchronously.
        // Exemple: dumping to a file
        string filename = String.Format(@"C:\Temp\webhook-event-{0}.json",
            Guid.NewGuid().ToString());
        File.WriteAllText(filename, json);
    }

    public bool IsReusable
    {
        get { return false; }
    }
}
```





Et un exemple en PHP :

```
<?php
// Read the posted JSON body
$json = file_get_contents('php://input');

// Do not process event contents here : queue the event to be processed
// asynchronously.
// Exemple: dumping to a file
$myfile = file_put_contents('C:\\temp\\webhook-event-' . uniqid('', true) .
'.json', $json);
?>
```

### Contact

Canada : 1 866 529-6214

France : 01 84 88 40 66

Russie : +7 (495) 226-04-11

Courriel : [info@dialoginsight.com](mailto:info@dialoginsight.com)

Site Web : [www.dialoginsight.com](http://www.dialoginsight.com)

Blogue : [www.dialoginsight.com/fr/ressources/academie](http://www.dialoginsight.com/fr/ressources/academie)



@DialogInsight



Dialog Insight

