



# Maxpho<sup>®</sup> WS

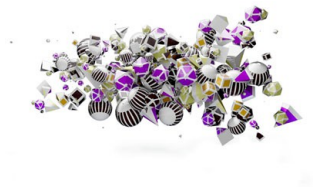


Application Program Interface - API  
Instant Notification Service - INS

---

*Date : 20 / 09 / 2011*  
*Version : 1.2*  
*Auteur: Maxpho Ltd*

---



## Sommaire général

1 - Introduction .....	3
1.1 - Structure d'interaction API.....	4
1.2 - Structure d'interaction INS.....	4
2 - API .....	5
2.1 - Structure d'une requête API.....	5
2.2 - Liste des appels API supportés.....	6
3 - Instant Notification Service .....	7
3.1 - Structure d'interaction INS.....	7
3.2 - Cas d'utilisation.....	8



# 1 Introduction

---

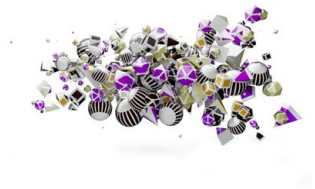
Maxpho API est une interface permettant de relier des logiciels externes à Maxpho Commerce, afin de consentir la configuration et la mise à jour directement d'un logiciel externe.

Maxpho API est un web-service qui offre un service lointain capable de traiter les transactions provenant d'appels externes. Les avantages principaux dérivant de l'utilisation d'appels API dans l'interface avec Maxpho sont:

- **Simplicité d'interaction:** L'avantage principal dérivant de l'utilisation d'appels API est la simplicité. À travers la construction d'un message XML relatif aux types de demandes que vous avez l'intention de faire et en la notifiant à l'application entry-point, Maxpho veillera à poursuivre la transaction et à fournir un résultat (succès / erreur et détails du résultat) par le moyen d'un type d'interaction state-less.
- **Sûreté:** chacun des appels API est composé d'une partie de HeadStack qui prévoit un mécanisme d'authentification de l'expéditeur de la demande. L'authentification est fournie au moyen d'un token d'authentification avec lease d'une durée biennale que le logiciel externe doit utiliser dans chaque interfacement avec l'interface Maxpho API.
- **Indépendance du logiciel d'interfacement:** N'importe quel logiciel externe peut s'interfacer avec Maxpho API, les seules qualités requises sont une bibliothèque XML pour la construction des appels et les connaissances de base dans la structure d'un appel API entre l'application client-server.

Ce système est particulièrement réfléchi pour des scénarios dans lesquels il est nécessaire d'avoir une synchronisation élevée entre le magasin physique et celui virtuel du logiciel Maxpho Commerce. Parmi les services offerts par l'interfacement, nous distinguons deux services fondamentaux:

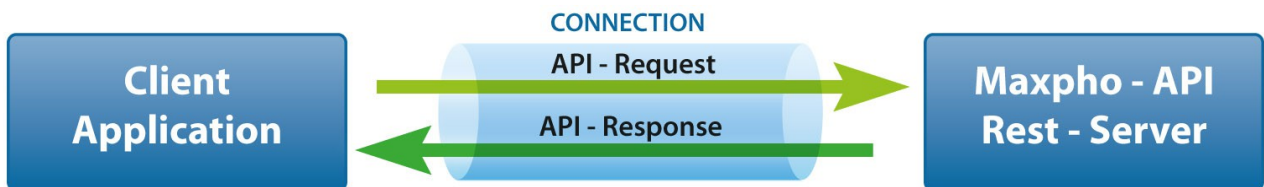
- **Mise à jour du magasin virtuel:** Par le biais des appels API, le logiciel externe peut ajouter des produits, mettre à jour des informations et les supprimer.
- **Mise à jour du magasin physique:** le système d'interfacement prévoit la configuration d'une adresse de notification à qui le système notifiera les variations de la quantité dans le magasin.



## 1.1 Structure d'interaction API

---

La structure de base d'un appel API est la suivante:



L'application client, une fois la demande effectuée, s'occupe de l'envoyer à l'entry-point de l'interface Maxpho API, qui, une fois effectué, informe du résultat de la manoeuvre.

## 1.2 Structure d'interaction INS

---

Le système de notification pour l'historique des commandes et pour la synchronisation du magasin entre le logiciel Maxpho Commerce et le logiciel externe, au choix du client, est structuré de la façon suivante:



Le système notifie les informations relatives à la commande, à l'adresse spécifiée en configuration, en incluant les détails relatifs à la variation sur le magasin des produits vendus.



## 2 API

---

Les appels API sont traités par un REST Server qui traite le message en format XML, envoyé par le logiciel externe.

### 2.1 Structure d'une requête API

---

Chacun des appels API est composé par deux sections:

- **HeadStack**: Chaque requête doit inclure la section d'authentification à l'intérieur de la Request, dans cette section, doivent être incluses certaines informations relatives à:
  - *Version*: version de l'appel API utilisée à l'entrée
  - *Token*: Chaque requête doit utiliser un token d'authentification compatible avec ceux reconnus par l'entry-point
- **Details de la requête**: Chaque requête doit nécessairement inclure tous les champs marqués comme **required** dans la documentation spécifique pour chaque demande.

Un appel classique API aura donc la structure suivante:

```
<?xml version="1.0" encoding="UTF-8"?>
<CallRequest>
  <HeadStack>
    <Version></Version>
    <Token></Token>
    <Action></Action>
  </HeadStack>
  <RequestDetail>
    <!-- Request detail nodes -->
  </RequestDetail>
</CallRequest>
```

### 2.2 Liste des appels API supportés

---

Actuellement, les appels API supportés sont ceux qui permettent de manipuler les produits présents dans le logiciel Maxpho Commerce, c'est à dire:

- *InsertProducts*: permet l'insertion de nouveaux produits [ 1 ... 10 ]
- *UpdateProducts*: permet de mettre à jour toutes les informations sur le produit [ 1 ... 10 ]
- *UpdateStockProducts*: permet de mettre à jour seulement le prix et la quantité de plusieurs



produits [ 1 ... 20 ]

- *OrderList*: permet la réception de toutes les commandes qui ont subi une variation dans une période de temps spécifiée lors de l'appel [1]
- *OrderStatusUpdate*: permet la mise à jour des commandes à partir de la valeur MxpOrderID [1 ... 10]



## 3 Instant Notification Service

---

INS est un système développé pour notifier à l'extérieur du logiciel des événements internes.

### 3.1 Structure d'interaction INS

---

Maxpho Commerce s'occupe de notifier en **POST**, sur un endpoint externe **HTTP/HTTPS** à partir d'un URL, les informations relatives à la variation en cours.

Les événements notifiés sont:

- **Réception commande** [OrderNotify]: Le système, à la réception d'une commande d'un des marketplace configurés sur Maxpho Commerce, procède à l'envoi des détails de la transaction complétée.
- **Mise à jour du statut de la commande** [OrderStatusUpdate]: Pour n'importe quelle variation du statut des commandes, Maxpho Commerce se charge d'en notifier la mise à jour (exemple: paiement reçu, envoyé, annulé etc...).

Pour procéder ou recevoir des notifications du système d'INS, il est nécessaire de développer sur l'endpoint de destination de la notification, un **listener** ou un **handler** qui reste à l'écoute des notifications provenant de Maxpho et qui, à leur réception, implémente une gestion opportune du message.

Le système ne nécessite pas de caractéristiques particulières sur l'endpoint, car les notifications sont envoyées en POST dans le format **XML** avec encoding **UTF8**.

Le mécanisme de notification est **state-less**, c'est à dire qu'il ne prévoit pas une gestion des erreurs reportées par le client en phase de parsing du message reçu. Actuellement, la gestion de l'échec est gérée uniquement sur la base de l'accessibilité de l'endpoint. En cas d'inaccessibilité du server ou de l'entry point (code d'erreur 401, 403, 404 et 500 d'Apache) le système INS se chargera de notifier via email à l'adresse email du magasin, l'impossibilité de communication du système avec l'endpoint externe.



## 3.2 Cas d'utilisation

---

Ci dessous, nous vous présentons certains cas d'utilisation en fournissant une idée de comment pouvoir profiter sur la plateforme du système de notification INS de Maxpho Commerce:

1. Dans une interaction de base, il est possible de profiter de la OrderNotify pour garder synchroniser votre magasin physique avec le magasin virtuel géré par Maxpho, répliqué à son tour sur tous les marketplace gérés par l'acheteur;
2. Avec une gestion plus soignée, en réception d'un événement OrderNotify, il est possible de sauvegarder la commande reçue et synchroniser les quantités vendues en ayant des informations relatives à l'acquéreur et aux détails de la vente (destination de la marchandise, produits vendus, informations sur l'expédition et sur le moyen de paiement sélectionné par l'acheteur). Par le biais de l'appel API OrdersList, ou par l'intermédiaire du service INS OrderStatusUpdate, il est possible de gérer la variation du statut, mise à jour sur l'administration ou par l'un des marketplace;
3. Interfacer le système de notification avec les systèmes BI (Business Intelligence) dans l'objectif de profiler et de geo-localiser de manière détaillée votre eBusiness.

Les cas d'utilisation proposés ont pour objectif d'illustrer quelques simples solutions d'intégrations avec des plateformes externes, mais sont considérés comme des exemples car une exploitation optimale de ce service doit être définie ad hoc sur la base de l'infrastructure du client.