

This document contains text Automatically extracted from a PDF or image file. Formatting May Have Been Lost and not all text May Have Been Recognized.

To remove this note, right-click and select "Delete table".

---



**SOGENACTIF**

**Guide du programmeur**

**API Plug-in**

**Version 1.1**  
**Mars 2005**

# SOGENACTIF

## Guide of programmer

### API Plug-in

Version 1.1

March 2005

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

# Atos Worldline





#### **Avertissements :**

- Le fichier Version.txt précise l'environnement dans lequel l'API a été compilée et testée. L'installation de l'API sur tout autre environnement n'est pas garantie.
- Les chemins et les champs présents dans les différents fichiers de l'API ne doivent en aucun cas contenir des espaces. La présence d'espaces provoque des messages d'erreur difficiles à interpréter car la valeur des champs concernés est tronquée.

## Disclaimers :

The File Version.txt precise Environment in which API has was compiled and tested.

Installation of API on all other Environment is not warranty.

The paths and the fields present in the different Files of API not must in no case contain of spaces. The presence spaces causes of Posts Error difficult to interpret because the value of fields concerned is truncated.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Guide of programmer Page : 1



## SOMMAIRE

1. INTRODUCTION.....	3
2. LA REQUÊTE DE PAIEMENT : CALL_REQUEST .....	4
2.1. PARAMETRES DE LA REQUÊTE.....	4
2.2. LE FICHIER PATHFILE .....	4
2.3. LE MODE DEBUG .....	5
2.4. EXECUTABLE REQUEST.....	5
2.5. PARAMETRE TRANSACTION_ID .....	6
3. LA REPONSE A UN PAIEMENT : CALL_RESPONSE .....	7
3.1. CHAMPS DE LA REPONSE.....	7
3.2. EXECUTABLE RESPONSE.....	7
4. LA REPONSE AUTOMATIQUE : CALL_AUTORESPONSE.....	8
4.1. POURQUOI DEUX REPONSES .....	8
4.2. PRINCIPE DE FONCTIONNEMENT .....	8
4.3. IMPLEMENTATION DU SCRIPT CALL_AUTORESPONSE.....	9
5. PROBLEMES DE FONCTIONNEMENT .....	10
5.1. LORSQUE L'ON APPELLE L'API .....	10
5.2. LORSQUE L'API APPELLE LE SERVEUR SOGENACTIF.....	11
5.3. LA REPONSE AUTOMATIQUE NE FONCTIONNE PAS .....	11
5.4. CONTACTS EN CAS DE PROBLEME .....	13
ANNEXE A : MOT-CLES DE LA REQUÊTE .....	14



## SUMMARY

1. INTRODUCTION .....	3
2. THE APPLICATION OF PAYMENT : CALL_REQUEST .....	4
2.1. P	
ARAMETERS OF THE COMPLAINT	
.....	4
2.2. L	
E FILE Pathfile	
.....	4
2.3. L	
E MODE	
DEBUG .....	5
2.4. E	
XECUTIVE REQUEST	
.....	5
2.5. P	
ARAMETERS TRANSACTION	
—	
ID	
.....	6
3. THE ANSWER A A PAYMENT : CALL_RESPONSE .....	7
3.1. C	
HAMP OF THE ANSWER	
.....	7
3.2. E	
XECUTIVE RESPONSE	
.....	7
4. THE ANSWER AUTOMATIC : CALL_AUTORESPONSE .....	8
4.1. PHY TWO ANSWERS .....	8
4.2. PRINCIPLE OF OPERATION.....	8
4.3. IMPLEMENTATION OF SCRIPT CALL_autoresponse.....	9
5. PROBLEMS OF OPERATION .....	10
5.1. WHEN ISIT CALLS TheAPI .....	10
5.2. WHEN TheAPI CALLS THE SERVER SOGENACTIF.....	11
5.3. LA RESPONSE AUTOMATIC NE WORKS NOT .....	11
5.4. CONTACT IN CAS OF PROBLEM .....	13
APPENDIX A : KEY WORD OF THE COMPLAINT .....	
14	

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005





## 1. INTRODUCTION

- Ce document vous décrit les différents scripts et exécutables disponibles dans l'API Plug-in Sogenactif, quel est leur rôle et comment les utiliser. Pour intégrer l'API, vous devrez développer 3 scripts (PHP, PERL ou SHELL) pour traiter les 3 messages échangés entre le serveur de paiement, le site marchand et le navigateur de l'internaute.
- L'API Plug-in Sogenactif présente 2 exécutables principaux :
  - request pour la création de la requête de paiement
  - response pour l'interprétation la réponse du serveur

Note : ce document ne décrit pas comment vous interfacer avec votre système d'information ou votre base de données. Dans les exemples fournis, les variables sont déjà renseignées, vous devrez programmer la lecture et la mise à jour des données de votre système d'information.

Logiciels pré-requis :

- Interpréteur PHP, PERL ou SHELL.
- Serveur http pour exécuter vos scripts.

Conventions d'écriture :

- Les renvois à d'autres documentations seront notés en majuscules et en italique.  
ex : *DICTIONNAIRE DES DONNEES*
- Les exécutables et champs de l'API seront notés en italique.  
ex : *request*, *pathfile*, *transaction\_id*
- Les scripts que vous devez développer seront notés en gras  
ex : **call\_request**

Outre *LE GUIDE DU PROGRAMMEUR*, le kit de paiement est distribué avec les documentations suivantes :

- *LA PRESENTATION GENERALE* (présentation fonctionnelle de l'offre de paiement)
- *LE GUIDE D'INSTALLATION* (manuel présentant les différentes étapes de l'installation de l'API)
- *LE DICTIONNAIRE DES DONNEES* (manuel présentant les différents champs de l'API)
- *LE GUIDE DE PERSONNALISATION DES PAGES* (manuel expliquant les différentes possibilités de personnalisation des pages de paiement)
- *LA DESCRIPTION DES JOURNAUX* (manuel décrivant les différents formats des journaux de fonds).

## 1. INTRODUCTION

- This document you described the different Scripts and executable available in API Plug-in Sogenactif, what is their role and how the use. To integrate IPY you will develop 3 Scripts (PHP PERL or SHELL) for deal the 3 Posts exchanged between the server of payment the Site Dealer and the browser of the user.

- API Plug-in Sogenactif This 2 executable main :

- request for the creation of the query of Payment
- response for interpretation the Reply of server

Note : this document not described not how you interface with your system Information or your base of data. In the Examples provided, the variables are already indicated, you will program the reading and the put to day of data of your system information.

Software Pre-requisites :

- Interpreter PHP PERL or SHELL.
- Server http for run your scripts.

Conventions Writing :

- The References to other documentation will rated in uppercase and in italics.

ex : *DICTIONARY OF DATA*

- The executable and fields of API will rated in italics.

ex : *request, pathfile, transaction\_id*

- The Scripts that you must develop will rated in fatty

ex : **call\_request**

Furthermore *THE GUIDE OF PROGRAMMER* the kit of Payment is distributed with the documentation following :

- *THE PRESENTATION GENERAL* (Presentation functional of supply of payment)
- *THE GUIDE INSTALLATION* (Manual with the different steps of Facility of API)
- *THE DICTIONARY OF DATA* (Manual with the different fields of API)
- *THE GUIDE OF CUSTOMIZATION OF PAGES* (Manual explaining the different opportunities of customization of pages of payment)
- *THE DESCRIPTION OF JOURNALS* (Manual describing the different formats of newspapers of funds).

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005





## 2. LA REQUÊTE DE PAIEMENT : CALL\_REQUEST

### 2.1. PARAMETRES DE LA REQUÊTE

Le *DICTIONNAIRE DES DONNEES* fournit la liste des champs de la requête, leur description et le caractère obligatoire ou facultatif de leur affectation.

Ces champs peuvent être renseignés dans les fichiers parmcom et dans le script *call\_request*. De plus, l'exécutable *request* affectera des valeurs par défaut aux champs *block\_align*, *block\_order*, *currency\_code* et *header\_flag* s'ils ne sont renseignés ni dans les fichiers parmcom, ni dans le script *call\_request*.

Pour renseigner un champ dans le script *call\_request*, il faut créer une chaîne de caractères de la forme *mot-clé=valeur* et passer cette chaîne de caractères en paramètre à l'exécutable *request* (rappel : *valeur* ne doit pas contenir d'espace). Le fichier *call\_request* comporte des exemples de chaînes de caractères pour chacun des champs de la requête. Il suffira de transmettre à l'exécutable *request* toutes les chaînes de caractères que vous souhaitez prendre en compte pour la requête.

#### *Comment se déroulera la création de la requête ?*

Pour créer une requête de paiement, vous devez appeler le script *call\_request* à l'aide d'un navigateur. Ce script appelle l'exécutable *request* avec les champs que vous aurez paramétré. L'exécutable *request* prépare la requête de paiement avec les valeurs qu'il lit dans le fichier parmcom.sogenactif, dans le fichier parmcom.<MERCHANT\_ID> et dans les paramètres fournis par le script *call\_request*.

Si un champ est renseigné dans le fichier parmcom.sogenactif, dans le fichier parmcom.<MERCHANT\_ID> et dans les paramètres fournis par le script *call\_request*, sa valeur sera écrasée au fur et à mesure des lectures. Dans cet exemple, le paramètre aura finalement la valeur transmise par le script *call\_request*.

Ce type d'affectation vous permet de renseigner statiquement des valeurs dans les fichiers parmcom et de les modifier dynamiquement dans le script *call\_request* si nécessaire.

### 2.2. LE FICHIER PATHFILE

Le fichier pathfile est le fichier de configuration principal de l'API. Il contient les chemins des autres fichiers paramètres de l'API et permet donc aux exécutables d'accéder aux fichiers parmcom.sogenactif, parmcom.014213245611111 et certificat. Le fichier pathfile contient également le chemin Internet des logos des moyens de paiement ainsi que le mot clé d'activation du mode DEBUG.

Deux possibilités s'offrent à vous pour la localisation du fichier pathfile sur votre serveur :

- Dans le script *call\_request*, vous précisez en paramètre la chaîne de caractères *pathfile=chemin\_vers\_le\_pathfile*. Alors vous devez positionner le fichier pathfile par rapport au *chemin\_vers\_le\_pathfile*.
- Dans le script *call\_request*, vous ne précisez pas le chemin du pathfile en paramètre. Alors vous devez positionner le fichier pathfile dans le même répertoire que le script *call\_request*.

Le chemin du fichier pathfile fourni à l'exécutable doit être un chemin physique (en aucun cas un chemin Internet). On recommande un chemin absolu, comme c'est le cas dans les exemples de scripts livrés.

## 2. THE APPLICATION OF PAYMENT :

CALL

REQUEST

22..11..PPAARRAAMMEETTRREESS DDEE LLAA RREEQQUUEETTEE

The *DICTIONARY OF DATA* provides the list of fields of the query their Description and the character Mandatory or Optional of their assignment.

These fields can be shown in the Files parmcom and in the script call\_request. Of more Executable *request* affect of values by default to fields *block\_align*, *block\_order*, *currency\_code* and *header\_flag* if not are shown or in the Files parmcom, or in the script call\_request.

To inquire a field in the script call\_request, it must create a chain of characters of the form *keyword = value* and pass this chain of characters in parameter to Executable *request* (Recall : *value* not must not contain space). The File call\_request has of Examples of chains of characters for each of fields of the query. It suffice of transmit to Executable *request* all the chains of characters that you want take in account for the query.

*How is place the creation of the query ?*

To create a query of payment you must call the script call\_request to help a browser. This script calls Executable *request* with the fields that you will set. The executable *request* prepares the query of Payment with the values it bed in the File parmcom.sogenactif, in the File parmcom. <MERCHAND\_ID> and in the parameters provided by the script call\_request.

If a field is informed in the File parmcom.sogenactif, in the File parmcom. <MERCHAND\_ID> and in the parameters provided by the script call\_request, its value will crushed at as and to measure of readings. In this example, the parameter will finally the value transmitted by the script call\_request.

This type Trust you allows of inquire statically of values in the Files parmcom and of the change Dynamically in the script call\_request if necessary.

22..22..LL

EE FFII CCHH IIEERR PPAATT HHEF I I LEE

The File pathfile is the File of configuration Main of API. It contains the paths of Other Files parameters of API and allows So to executable access to Files parmcom.sogenactif, parmcom.014213245611111 and certificate. The File pathfile contains also the Road Internet of Logos of means of Payment and that the word key activation of fashion DEBUG.

Two opportunities available to you for the location of File pathfile on your server :

In the script call\_request, you specify in parameter the chain of characters *pathfile* =chemin\_vers\_le\_pathfile. So you must position the File pathfile by report at *chemin\_vers\_le\_pathfile*.

In the script call\_request, you not specify not the Road of pathfile in parameter. So you must position the File pathfile in the same

Directory that the script call\_request.

The Road of File pathfile provided to Executable must be a Road physical (In no case a Road Internet). We recommends a Road absolute as is the case in the Examples of Scripts delivered.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005

Guide of programmer Page : 4







La syntaxe des lignes de ce fichier est la suivante :

- Chaque ligne commençant par # est une ligne de commentaires.
- Les chemins sont paramétrés par des lignes « mot-cléchemin! »

Les trois paramètres essentiels sont les trois variables F\_CERTIFICATE, F\_PARAM, et F\_DEFAULT qui doivent contenir respectivement les chemins complets des fichiers certif.fr.014213245611111, parmcom.014213245611111, et parmcom.sogenactif. Seul le fichier parmcom.sogenactif doit être paramétré avec son extension .sogenactif. Les extensions des autres fichiers seront ajoutées par l'API d'après les paramètres renseignés dans le script call\_request.

Il est recommandé de placer des chemins complets dans ces variables. De plus, ces chemins doivent être des chemins physiques du serveur, et non pas des chemins internet.

Le paramètre D\_LOGO sert à l'affichage des moyens de paiement. Lui seul doit être renseigné avec une portion de chemin Internet, soit l'URL du répertoire des logos de l'API sur le site web, amputée du nom de domaine de ce dernier. (cf. *GUIDE D'INSTALLATION*).

Le dernier paramètre DEBUG sert à l'affichage des moyens de paiement en mode DEBUG.

### **2.3. LE MODE DEBUG**

Le mode DEBUG permet d'afficher (en format HTML) l'environnement complet de l'api, ainsi que tous les paramètres de la requête et de la réponse d'un paiement.

Pour activer ce mode, le paramètre DEBUG du fichier pathfile doit être renseigné à YES.

Le mode DEBUG est constitué de différentes rubriques indiquant les ressources utilisées par l'api. Il permet ainsi de vérifier, par exemple, le contenu du fichier pathfile, l'URL du serveur de paiement ou encore la valorisation des champs de la requête de paiement.

En phase de production, ce mode devra être désactivé (renseignement de la valeur du paramètre DEBUG autre que YES).

### **2.4. EXECUTABLE REQUEST**

Utilisation : `request var1 var2 var3...`

avec  
`varx` une chaîne de caractère du type *mot-clé=valeur*

La liste des différents mot-clés est précisée dans [l'annexe A](#).

Rappel : *valeur* ne doit pas contenir d'espace.

Retour :

L'exécutable *request* retourne une chaîne de caractères composée de trois chaînes de caractères séparées par un « ! » : `!code!error!buffer!`

avec *code*, le code retour du traitement. Deux valeurs sont possibles :

- 0 : l'exécutable génère un formulaire HTML contenue dans la variable *buffer*
- -1 : l'exécutable retourne un message d'erreur dans la variable *error*. (cf. rubrique [problèmes de fonctionnement](#)).

Ce document est propriété de Atos Worldline. Il ne peut être reproduit ou divulgué sans autorisation écrite préalable.

Réf : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005

The Syntax of lines of this File is the Next :

Each Online beginning by # is a Online of comments.

The paths are parameterized by of lines " tag! way! "

The three parameters essential are the three variables F\_CERTIFICATE, F\_PARAM, and F\_DEFAULT that must contain respectively the paths complete of Files certif.fr.014213245611111, parmcom.014213245611111, and parmcom.sogenactif. Only the File parmcom.sogenactif must be set with its extension . Sogenactif. The Extensions of Other Files will added by API according the parameters shown in the script call\_request.

It is recommended of place of paths complete in these variables. Of more these paths must be of paths physical of server, and not not of paths Internet.

The parameter D\_LOGO serves to Display of means of payment. Him single must be informed with a portion of Road Internet or URL of Directory of Logos of API on the Site Web amputated of name of field of this last. (Cf. *GUIDE INSTALLATION*).

The last parameter DEBUG serves to Display of means of Payment in fashion DEBUG.

22..33..LL

EE MM OOD DEE D DEEB BUUGG

The fashion DEBUG allows display (In format HTML) Environment Full of api, and that all the parameters of the query and of the Reply a payment.

To activate this fashion the parameter DEBUG of File pathfile must be informed to YES.

The fashion DEBUG is formed of different topics indicating the Resources used by api. It allows and of check by example, the content of File pathfile, URL of server of Payment or yet the recovery of fields of the query of payment.

In phase of production this fashion will be disabled (Intelligence of the value of parameter DEBUG other that YES).

22..44..EE

XXEECCUUTTAABLL EE RREQQUUESSTT

Use : *request var*

1

*var*

2

*var*

3

...

with

*var*

X

a chain of character of type *keyword* = *value*

The list of different keywords is specified in Schedule A.

Reminder : *value* not must not contain space.

Back :

The executable *request* returns a chain of characters composed of three chains of

characters separate by a " ! " : ! Code! Error! Buffer!

with *code* the code return of treatment. Two values are possible :

0 : Executable generates a form HTML contained in the Variable *buffer*

-1 : Executable returns a Message Error in the Variable *error*. (Cf. heading problems of operation).

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005





Le formulaire HTML contenue dans la variable *buffer* comporte :

- La liste des logos des moyens de paiement.
- L'URL du serveur Sogenactif.
- Des données cachées à transmettre au serveur Sogenactif.

Cette page doit ensuite être affichée sur le poste de l'internaute. Les blocs d'icônes de paiement proposés par la boutique incluent chacun un en-tête en option (cf. *GUIDE DE PERSONNALISATION*).

L'exécutable *request* construit un message crypté à partir de toutes les données que vous aurez renseignées dans les fichiers *parmcom* et le script *call\_request* (l'activation du mode *DEBUG* permet de s'assurer de la valeur finale des paramètres de la requête de paiement). Il le signe ensuite par un MAC (Message Authentication Code), un scellement calculé avec un algorithme DES. Ce message crypté apparaît (en variable cachée) dans la page HTML retourné par l'exécutable. Il sera posté sur l'URL du serveur Sogenactif pour permettre l'identification du commerçant et par suite, le traitement de sa demande de transaction.

## **2.5. PARAMETRE TRANSACTION\_ID**

Dans le script *call\_request*, deux possibilités s'offrent à vous pour renseigner le champ *transaction\_id* :

- si vous ne le transmettez pas en paramètre à l'exécutable *request*, une valeur sera générée automatiquement. Cette valeur est basée sur l'heure système (format *hhmmss*), elle est donc conforme au format du champ *transaction\_id* (6 caractères numériques).
- Si vous transmettez le *transaction\_id* à l'exécutable *request*, la valeur fournie sera prise en compte pour la transaction.

**WARNING :** Etant donné que la génération automatique du *transaction\_id* s'appuie sur l'heure système pour générer un numéro de transaction, on notera deux inconvénients : tout d'abord on ne couvre que 86400 nombres (24h\*60mn\*60s) et non pas toute la plage possible de numéros de transaction (000000 à 999999). Ceci peut être gênant pour de très gros sites, le numéro de transaction devant être unique sur une journée.

D'autre part, il subsiste un risque de doublons, si deux internautes distincts souhaitant payer au même instant appellent le script *call\_request* dans la même seconde. Une des deux transactions sera alors nécessairement rejetée comme un doublon, pour la même cause d'unicité que précédemment.

Il sera donc toujours préférable de gérer un numéro de séquence pour le champ *transaction\_id* sur le site web commerçant, plutôt que d'utiliser la génération automatique.

The form HTML contained in the Variable *buffer* has :

- The list of Logos of means of payment.
- The URL of server Sogenactif.
- Of data hidden to transmit at server Sogenactif.

This page must then be displayed on the post of the user. The blocks Icons of Payment proposed by the Shop include each a header in option (Cf. *GUIDE OF CUSTOMIZATION*).

The executable *request* built a Message encrypted to from of all the data that you will indicated in the Files parmcom and the script call\_request (Activation of fashion DEBUG allows of ensure of the value final of parameters of the query of payment). It the sign then by a MAC (Message Authentication Code) a sealing calculated with a algorithm DES. This Message encrypted appears (In Variable hidden) in the page HTML returned by the executable. It will posted on URL of server Sogenactif for allow identification of shopping and by Subsequently, the treatment of its request of transaction.

## 22..55.. PP

AARRAAMMEETTRREE TTRRAANNSSAACCTTIIOONN\_\_  
/IDD

In the script call\_request, two opportunities available to you for inquire the field *transaction\_id* :

if you not the pass not in parameter to Executable *request*, a value will generated automatically. This value is based on Time system ( format hhmmss) it is So consistent at format of field *transaction\_id* (6 characters digital).

If you pass the *transaction\_id* to Executable *request*, the value provided will taken in account for the transaction.

WARNING : As given that the generation Automatic of *transaction\_id* based on Time system for generate a number of transaction we Note two disadvantages : all First we not covers that 86400 numbers (24h \* 60min \* 60s) and not not any the beach possible of numbers of transaction (000000 to 999999). This may be embarrassing for of very Wholesale sites the number of transaction before be single on a day.

On the other hand, it remains a risk of duplicates, if two Internet separate wishing pay at same moment call the script call\_request in the same second. A of two Transactions will then necessarily rejected as a duplicate, for the same cause Uniqueness that previously.

It will So always preferable of manage a number of sequence for the field *transaction\_id* on the Site Web merchant, rather that use the generation automatic.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005







### 3. LA REPONSE A UN PAIEMENT : CALL\_RESPONSE

#### 3.1. CHAMPS DE LA REPONSE

Dans la réponse du paiement, on retrouve les champs généraux de la transaction tels *merchant\_id*, *amount*, et *currency\_code*, mais aussi les champs renseignés par le serveur Sogenactif suite à la demande d'autorisation : *response\_code*, *payment\_certificate*, etc...

Le *DICTIONNAIRE DES DONNEES* fournit la liste des champs de la réponse et leur description.

Le serveur Sogenactif renvoie les données de la transaction en méthode POST et de façon cachée. En fait, elle « poste » une variable DATA cryptée sur les URL *normal\_return\_url* et *cancel\_return\_url* renseignées dans la requête de paiement. L'exécutable *response* décrypte cette variable et renseigne tous les champs de la réponse.

#### 3.2. EXECUTABLE RESPONSE

Après la validation des coordonnées bancaires de l'internaute, le serveur Sogenactif affiche en cas d'acceptation de la transaction une page de réponse. Cette page résume les données de la transaction, et présente un bouton « RETOUR A LA BOUTIQUE » qui reconnecte l'internaute à l'URL *normal\_return\_url*. Dans le cas d'un refus de l'autorisation par les serveurs bancaires, le serveur Sogenactif affiche une page de refus avec un bouton « ANNULATION – RETOUR A LA BOUTIQUE » qui renvoie l'internaute vers l'URL *cancel\_return\_url*.

Le bouton « ANNULATION – RETOUR A LA BOUTIQUE » présent sur la page de saisie des coordonnées carte bancaire à le même effet que son homonyme de la page de refus.

Dans chacun de ces cas, une variable cachée DATA contenant la chaîne de caractères cryptée des données de la transaction est postée sur l'URL *normal\_return\_url* ou *cancel\_return\_url*.

Ce mode permet au commerçant de reconnecter l'internaute à son site, puisque le développeur a une totale liberté de paramétrage de ces URL. Il faut malgré tout s'assurer que le serveur commerçant autorise la réception de données en méthode POST.

Remarque : les URL de retour devront nécessairement pointer vers des CGI. En effet, certains serveurs n'acceptent pas que l'on poste des données sur des pages HTML statiques et retournent alors le message « methode POST not allowed ». Rien ne vous empêche par contre d'écrire une page statique dans votre script *call\_response*.

Le script *call\_response* récupère la variable « DATA » et la transmet en paramètre à l'exécutable *response* qui va ensuite la décrypter et renseigner les différents champs de la réponse (l'activation du mode DEBUG permet d'afficher l'ensemble des paramètres de la réponse du serveur de paiement).

Utilisation : *response message*

avec

*message* une chaîne de caractère du type *message=DATA*

L'exécutable *response* peut prendre un ou deux paramètres. Le second paramètre qui peut être transmis est une chaîne de caractères du type *pathfile=chemin\_vers\_le\_pathfile* pour spécifier la localisation du fichier *pathfile* (cf. paragraphe 2.2).

Ainsi, l'exécutable *response* peut prendre un ou deux paramètres en entrée.

### 3. THE ANSWER A A PAYMENT :

CALL

RESPONSE

33..11..CCHHAAMMPSS DDEE LLAA RREEPPOONNSSEE

In the Reply of payment we found the fields General of the transaction such *merchant\_id*, *Amount*, and *currency\_code*, but also the fields shown by the server Sogenactif Following to the request Authorization : *response\_code*, *payment\_certificate*, etc. ...

The *DICTIONARY OF DATA* provides the list of fields of the Reply and their description.

The server Sogenactif returns the data of the transaction in method POST and of way hidden. In Indeed, it " post " a Variable DATA encrypted on the URL *normal\_return\_url* and *cancel\_return\_url* indicated in the query of payment.

The executable *response* decrypts this Variable and information all the fields of the response.

33..22..EE

XXEECCUUTTAABBLEE RREESSPPOONNSSEE

After the Validation of coordinates Banking of the Internet, the server Sogenactif Poster in case acceptance of the transaction a page of response. This page summarizes the data of the transaction and This a button " BACK A THE SHOP " that reconnects Internet users to URL *normal\_return\_url*. In the case a refusal of authorization by the Servers Banking the server Sogenactif Poster a page of refusal with a button " CANCELLATION - BACK A THE SHOP " that returns Internet users to URL *cancel\_return\_url*.

The button " CANCELLATION - BACK A THE SHOP " This on the page of seizure of coordinates map Banking to the same effect that its homonym of the page of refusal.

In each of these case a Variable hidden DATA containing the chain of characters encrypted of data of the transaction is posted on URL *normal\_return\_url* or *cancel\_return\_url*.

This fashion allows at shopping of reconnect Internet users to its site since the Developer has a Total freedom of parameterization of these URL. It must despite all ensure that the server shopping authorizes the Reception of data in method POST.

Note : the URL of return will necessarily point to of CGI. In Indeed, some Servers accept not that one post of data on of pages HTML static and return then the Message " method POST not Allowed . Nothing not you prevents by cons write a page static in your script *call\_response*.

The script *call\_response* recovers the Variable " DATA " and the transmits in parameter to Executable *response* that will then the decrypt and inquire the different fields of the Reply (Activation of fashion DEBUG allows display all of parameters of the Reply of server of payment).

Use : *response Message*  
with

*Message* a chain of character of type *message = DATA*

The executable *response* may take a or two parameters. The second parameter that

may be transmitted is a chain of characters of type *pathfile = chemin\_vers\_le\_pathfile* for specify the location of File pathfile (Cf. paragraph 2.2).

Thus, Executable *response* may take a or two parameters in entry.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005



**Retour :**

L'exécutable *response* retourne une chaîne de caractères composée de 32 chaînes de caractères séparées par des « ! » :

```
!code!error!merchant_id!merchant_country!amount!transaction_id!payment_means!transmission_date!payment_time!payment_date!response_code!payment_certificate!authorisation_id!currency_code!card_number!cvv_flag!cvv_response_code!bank_response_code!complementary_code!complementary_info!return_context!caddie!receipt_complement!merchant_language!language!customer_id!order_id!customer_email!customer_ip_address!capture_day!capture_mode!data!
```

avec *code*, le code retour du traitement. Deux valeurs sont possibles :

- 0 : l'exécutable renseigne les champs de la réponse
- -1 : l'exécutable retourne un message d'erreur dans la variable *error*. (cf. rubrique [problèmes de fonctionnement](#)).

Tous les champs de la réponse sont donc disponibles au niveau du script *call\_response*.

## 4. LA REPONSE AUTOMATIQUE : CALL\_AUTORESPONSE

### 4.1. POURQUOI DEUX REPONSES

Comme expliqué au paragraphe 3.2, l'appel de l'URL *nomal\_return\_url* (ou *cancel\_return\_url*) est déclenché par l'internaute, lorsqu'il clique sur le bouton « RETOUR A LA BOUTIQUE » du ticket électronique ou « ANNULATION – RETOUR A LA BOUTIQUE » du ticket ou de la page de saisie des coordonnées bancaires. Par conséquent, il se peut qu'aucune de ces URL ne soit déclenchée lors d'un paiement, pour peu que l'internaute perde sa connexion, ou tout simplement ferme son navigateur sans avoir cliqué sur un de ces boutons. Le commerçant risque alors d'avoir une commande initialisée dans sa base de données, mais qui ne serait pas confirmée avant la réception de son journal des transactions le lendemain.

Pour pallier ce cas, le serveur Sogenactif renvoie une réponse automatique de la transaction en même temps qu'il affiche la page de réponse.

Ce mode permet au commerçant d'être informé systématiquement, et en temps réel, de toutes les transactions qui sont effectuées sur son site.

### 4.2. PRINCIPE DE FONCTIONNEMENT

La réponse automatique ressemble en tous points à la réponse manuelle, c'est à dire que le serveur poste la même variable DATA cryptée que celle de la réponse manuelle sur l'URL *automatic\_response\_url*.

La différence majeure est que l'envoi de cette réponse automatique est fait par le serveur Sogenactif, et non pas par le navigateur de l'internaute.

L'absence de navigateur implique des différences du point de vue du traitement de cet appel. Le script *call\_autoreponse* ne peut pas faire d'affichage, ni de redirection (pas de navigateur à l'écoute). On ne peut y effectuer que des traitements automatiques, tels une mise à jour de base de données, un envoi de mail, ou tout simplement une sauvegarde des données dans un fichier comme cela est présenté dans l'exemple.

Back :

The executable *response* returns a chain of characters composed of 32 chains of characters separate by of " ! " :

! Code! Error! Merchant\_id! Merchant\_country! Amount! Transaction\_id! Payment\_means! Work  
nsmission\_date! payment\_time! payment\_date! response\_code! payment\_certificate! aut  
horisation\_id! currency\_code! card\_number! cvv\_flag! cvv\_response\_code! bank\_respon  
se\_code! complementary\_code! complementary\_info! return\_context! cart! receipt\_co  
MPLEMENTATION! merchant\_language! language! customer\_id! order\_id! customer\_email! custo  
mer\_ip\_address! capture\_day! capture\_mode! data!

with *code* the code return of treatment. Two values are possible :

0 : Executable information the fields of the Reply

-1 : Executable returns a Message Error in the Variable *error*. (Cf.  
heading problems of operation).

All the fields of the Reply are So available at level of script  
call\_response.

#### 4. THE ANSWER AUTOMATIC :

CALL

—  
Autoresponse

44..11..PP

COUURRQUUOOI DDEEUUX RREEPPOONNSSEESS

As explained at paragraph 3.2, Call of URL *normal\_return\_url* (Or  
*cancel\_return\_url* ) is triggered by the Internet, when clique on the button " BACK A  
THE SHOP " of ticket e or " CANCELLATION - BACK A THE SHOP "  
of ticket or of the page of seizure of coordinates bank. By Therefore, it is may  
no of these URL not or triggered at a payment for little that Internet users  
lose its connection or all simply farm its browser without have clicked on a of  
these button. The shopping risk then have a control initialized in its base of  
data but that not would not Confirmed before the Reception of its newspaper of  
Transactions the next day.

To palliate this case the server Sogenactif returns a Reply Automatic of the  
transaction in same time it Poster the page of response.

This fashion allows at shopping be informed systematically and in time real  
of all the Transactions that are made on its site.

44..22..PPRRINNCCIIPPEE DDEE FFOONNCCTTIOONNNNEEMMEENNTT

The Reply Automatic looks in all points to the Reply manual, is to say  
that the server post the same Variable DATA encrypted that that of the Reply Manual on  
URL *automatic\_response\_url*.

The difference major is that sending of this Reply Automatic is fact by the  
server Sogenactif, and not not by the browser of the user.

The absence of browser implies of differences of point of view of treatment of this  
appeal. The script call\_autoresponse not may not do display, or of redirect (Not of  
browser to listening). We not may y perform that of treatment automatic such a  
put to day of base of data a sending of mail or all simply a aving of

data in a File as this is presented in Example.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005

Guide of programmer Page : 8







#### **4.3. IMPLEMENTATION DU SCRIPT CALL\_AUTORESPONSE**

Le script `call_autoresponse` est très proche du script `call_response`. Ce script récupère tout d'abord la variable cryptée `DATA` et la transmet à l'exécutable `response`. Le chemin vers le pathfile peut également être transmis en paramètre à l'exécutable `response` (cf. paragraphe 2.2). L'exécutable `response` renverra un message d'erreur ou la liste des champs composant la réponse.

A ce stade, vous vous trouvez en possession de toutes les données de la transaction.

Le traitement qui suit est au libre choix du commerçant, en fonction de son système d'information. (mise à jour de base données, envoi de mail, ...). Dans le script d'exemple `call_autoresponse`, les messages d'erreur et les champs de la réponse sont sauvegardés dans un fichier.

Si votre script `call_autoresponse` ne fonctionne pas, consultez la rubrique [problèmes de fonctionnement](#).

#### 44..33..II

MMPLLEEMMEENNTTAAATTIOONN DDUU SSCRRIPPTT CCAALLL\_\_  
AAUUTTOORREESSPOONNSSEE

The script call\_autoresponse is very close of script call\_response. This script recovers all First the Variable encrypted *DATA* and the transmits to Executable *response*. The Road to the pathfile may also be transmitted in parameter to Executable *response* (Cf. paragraph 2.2). The executable *response* return a Message Error or the list of fields component the response.

A this stage you you find in possession of all the data of the transaction.

The treatment that follows is at Free choice of merchant, in function of its system information. (Setting to day of base data sending of mail ...). In the script example call\_autoresponse, the Posts Error and the fields of the Reply are Saved in a file.

If your script call\_autoresponse not works not see the heading problems of operation.

This document is Property of Atos Worldline. It not may be reproduced or disclosed without authorization written beforehand.

Ref : GuideProgrammeurApiPlugin\_Sogenactif.doc Date : 11/03/2005