www.zodianet.com

API ZODIANET
« ZAPI 1 »

Description des échanges informatiques entre le système ZiBASE/ZODIANET et des parties tierces (third parties)

Document : Spécification
Source : ZODIANET.
Mail support@zodianet.com
Révision :
- **12 : 15/12/2011 Mise en dispo publique de Zapi.**
- **13 : 30/09/2012 définition des bitmaps de sensors.xml. Effacement d'une alerte.**

CE DOCUMENT EST LA PROPRIETE DE LA SOCIETE ZODIANET

SOMMAIRE

1 PRÉSENTATION 3

2 UTILISATION DE ZIBASE EN COPROCESSEUR RADIO (SCHEMA A) 3

2.1 GENERALITES SUR LES ECHANGES AVEC ZIBASE. 4
2.1.1 Format de l'Entête 4
2.1.2 Paquets envoyés par le système HOST vers ZiBASE 4
2.1.3 Paquets envoyés par ZiBASE vers le système HOST 10

3 GESTION DE TERMINAUX DE TIERCES PARTIES (SCHEMA B) 12

3.1 INFORMATIONS FOURNIES PAR ZIBASE PAR HTTP 13
3.1.1 Page xml des données temps réel de capteurs et sondes 13
3.2 COMMANDES ENVOYEES A ZIBASE PAR HTTP 15
3.2.1 Commandes soumises à l'interpréteur de commandes 15
3.3 GENERALITES SUR LES ECHANGES AVEC LA PLATEFORME ZODIANET. 15
3.4 INFORMATIONS FOURNIES ET SERVICES OFFERTS PAR LA PLATEFORME 15
3.4.1 Identification des requêtes 15
3.4.2 Récupération du token 16
3.4.3 Récupération du fichier .csv d'historique de sondes 16
3.4.4 Récupération du fichier sensors.xml par intermédiation de la plateforme 17
3.4.5 Récupération de la page xml des descripteurs de périphériques et scénarios déclarés 18
3.4.6 Envoi de commandes domotiques 20

1 PRÉSENTATION

L'objectif de ZAPI est de mettre en communication le système ZIBASE/Zodianet avec des équipements tiers (notés ici HOST).
Deux schémas sont présentés dans ce document :
- Utilisation de ZiBASE en coprocesseur radio (Schéma A), (par exemple HomeSeer)
- Gestion de terminaux de tierces parties (Schéma B), avec les smartphones/tablettes….

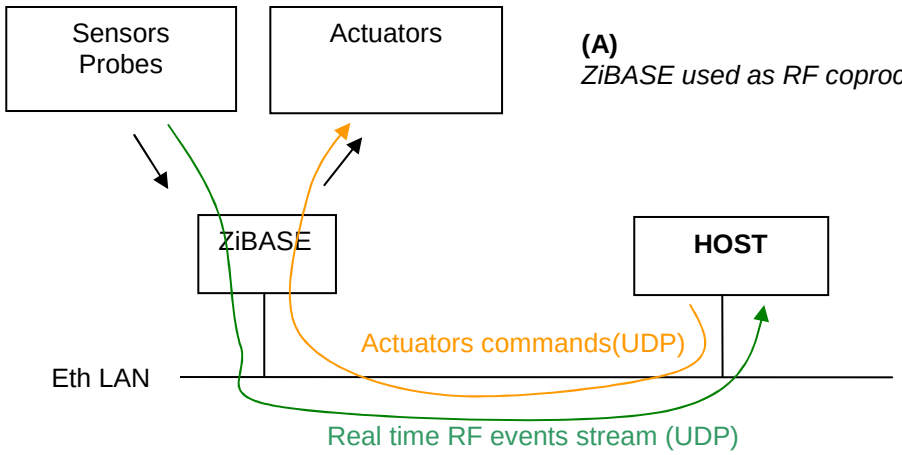
Ils ne sont pas exclusifs, et donc un 'mix' des deux peut être fait par l'utilisateur.
Ils coopèrent avec le fonctionnement normal de ZIBASE et de sa plateforme.

2 UTILISATION DE ZIBASE EN COPROCESSEUR RADIO (SCHEMA A)

Dans un premier mode de fonctionnement (A), les capacités d'entrées sorties/sorties radio de ZiBASE sont mises à profit (sans l'intervention de la plateforme Zodianet) par un gestionnaire domotique externe (HOST).

Mode opératoire:
- le système HOST broadcaste un paquet UDP sur le LAN, auquel les ZiBASEs présentes vont répondre. Le système HOST connaît de cette façon les ZIBASEs présentes et leur adresse IP respectives.
- Si le système HOST souhaite recevoir les événements radio reçus par ZIBASE, il émet un paquet UDP d'enregistrement auprès de ZiBASE en spécifiant son adresse IP (Host IP).
- ZiBASE émet alors les événements radio vers le système HOST.
- Si le système Host veut se dés-enregistrer, il doit émettre un paquet avec Host IP = 0.
- Pour émettre un signal radio, le système HOST émet un paquet UDP vers ZIBASE. La phase d'enregistrement n'est pas nécessaire dans ce sens.

ZIBASE ne peut enregistrer qu'un seul HOST à un instant donné.
Le système HOST peut tester périodiquement la présence de ZIBASE.
Si le système HOST gère plusieurs ZiBASE, il est à sa charge de gérer les événements radio dupliqués qui lui seront certainement retournés et d'envoyer vers la bonne ZiBASE les ordres à transmettre dans l'autre sens.
Il est à noter que ces échanges ne sont pas authentifiés et que par conséquent :
1) ZIBASE sur les ports considérés ne doit pas être visible d'Internet.
2) Le système HOST doit se situer par conséquent sur le LAN de l'utilisateur.



(A)
ZiBASE used as RF coprocessor scheme

2.1 Généralités sur les échanges avec ZiBASE.

Le protocole utilisé est UDP sur IP.
Le format de l'entête du paquet UDP est identique dans les 2 sens (ZiBASE->HOST et HOST->ZiBASE) et d'une longueur de 70 octets.

2.1.1 Format de l'Entête

```
struct ZAPI_packet_header{
unsigned char header[4]; // fixed header 'ZSIG'
unsigned short command; // command number
unsigned char reserved1[16];
unsigned char zibase_id[16];
unsigned char reserved2[12];
unsigned long param1; // generic parameters for commands
unsigned long param2; // generic parameters for commands
unsigned long param3; // generic parameters for commands
unsigned long param4; // generic parameters for commands
unsigned short my_count; // incr.counter at each sent packet (not incr. in burst of same packets)
unsigned short your_count; // last counter value received by the other side
};
```

Les champs short et long sont à présenter en BIG ENDIAN. (p/i les processeurs x86 ont une représentation LITTLE ENDIAN, donc une inversion d'octets est donc à prévoir sur PC).
Le system HOST doit impérativement remplir le champ *header* avec « ZSIG ».
Le champ *command* contient le type de commande et doit naturellement être renseigné.
Les champs *reserved1, reserved2, base_id* et *your_count* doivent être mis à 0 par le system HOST lors de l'émission d'un paquet. En réception, base_id renseigne sur l'identifiant de l'équipement ZIBASE adressé (exemple ZiBASE123456). Ce champ peut être utile dans des installations multi-ZIBASEs.
Les champs *reserved1, reserved2,* et *your_count* doivent être ignorés en réception par le système HOST.
Pour future compatibilité :
Le champ my_count doit être incrémenté de 1 à chaque paquet envoyé à ZiBASE.
Le système HOST doit ignorer quand des paquets successivement reçus possèdent un même champ *my_count*.

☞ *Les instructions **pack** et **unpack** de PHP sont bien utiles pour manipuler cette entête dans ce langage.*
☞ *Un plug-in ZiBASE pour HomeSeer, qui utilise cette API, est disponible sur le WEB. L'utilisateur peut s'en aider en sniffant les échanges avec Wireshark.*

2.1.2 Paquets envoyés par le système HOST vers ZiBASE

Sa longueur totale est égale à l'entête. Les champs *command, param1…param4* portent les informations utiles.
Toutes les commandes sont à envoyer sur le port UDP = 49999 par le système HOST.
Toutes les commandes envoyées par le système HOST sont acquittées par ZIBASE. L'acquittement est normalement très rapide car les requêtes sont mises en file d'attente. Un temps d'attente plus important arrivera à saturation de cette file. Mais après 3 secondes, il peut être considéré qu'un acquittement non reçu, n'arrivera jamais.
Une commande est broadcastée sur l'ensemble des ZIBASEs en mettant 255 sur le LSB de l'adresse IP. (exemple 192.168.1.255 port UDP 49999)

Commande NOP : ZiBASE présente ou découverte de son IP par broadcasting xxx.xxx.xxx.255

Cette commande est sans effet sur ZIBASE sauf de la forcer à répondre pour savoir si elle est présente ou connaître son adresse IP si le paquet était broadcasté.
command = decimal 8
param1..4 = -

Commande HOST_REGISTERING.

Cette commande permet l'enregistrement du système HOST auprès de ZIBASE.
command = decimal 13
param1 = adresse IP du système HOST à enregistrer.
param2 = port UDP du système HOST à enregistrer
param3 = 0 si non utilisé. Si utilisé (installation sans Internet), spécifier l'heure GMT courante imposée (format Unix http://www.onlineconversion.com/unix_time.htm). Utile si ZiBASE n'a pas d'accès Internet pour caler son horloge auprès d'un serveur ntp ou de la plateforme Zodianet. Attention! ZiBASE ne lance pas un scénario lié au temps tant que l'heure GMT n'est pas acquise d'une manière ou d'une autre.
param4 = 0 si non utilisé. Si utilisé, spécifie une fréquence particulière de communication. (avec sauvegarde EEPROM)

433 :	433,92Mhz (EU)	(device RF LOW) // factory default	
310 :	310Mhz (US)	(device RF LOW) // interdit en France !!!	
418 :	418Mhz (GB)	(device RF LOW) // interdit en France !!!	
950 :	868,950Mhz (EU)	(device RF HIGH) // factory default	
350 :	868,350Mhz (EU)	(device RF HIGH)	

PS :
- La modification de fréquence est à lancer au démarrage. Elle n'a pas vertu à changer sans cesse. C'est une donnée rémanente qui reste après extinction de ZIBASE. Le script de commande *FREQ 433 ; FREQ 950* permet de revenir en configuration par défaut à partir du configurateur Zodianet.
- param3 et param4 sont bien uniquement des options ! Mettre 0 par défaut.

Exemple de paquet de commande HOST_REGISTERING envoyé en broadcast pour s'enregistrer auprès de toutes ZiBASEs présentes sur le LAN:
Le label Zapinit n'est plus nécessaire. L'adresse IP d'enregistrement est hexa C0.A8.01.C9 soit 192.168.1.201 et le port UDP hexa 42CC soit 17100.

Trace Wireshark :

```
⊞ Frame 55 (112 bytes on wire, 112 bytes captured)
⊞ Ethernet II, Src: D-Link_c5:77:7f (00:50:ba:c5:77:7f), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol, Src: 192.168.1.201 (192.168.1.201), Dst: 192.168.1.255 (192.168.1.255)
⊞ User Datagram Protocol, Src Port: kyoceranetdev (1063), Dst Port: 49999 (49999)
⊟ Data (70 bytes)
    Data: 5A534947000D5A617069496E6974000000000000000000000...

0000  ff ff ff ff ff ff 00 50  ba c5 77 7f 08 00 45 00   .......P ..w...E.
0010  00 62 22 6c 00 00 80 11  93 06 c0 a8 01 c9 c0 a8   .b"l.... ........
0020  01 ff 04 27 c3 4f 00 4e  8c 2f 5a 53 49 47 00 0d   ...'.O.N ./ZSIG..
0030  5a 61 70 69 49 6e 69 74  00 00 00 00 00 00 00 00   ZapiInit ........
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0050  00 00 00 00 00 00 00 00  00 00 00 00 c0 a8 01 c9   ........ ........
0060  00 00 42 cc 00 00 00 00  00 00 00 00 00 00 00 00   ..B..... ........
```

Commande HOST_UNREGISTERING.

Cette commande permet le dés-enregistrement du système HOST auprès de ZIBASE.
command = decimal 22
param1 = adresse IP du système HOST à dés-enregistrer.
param2 = port UDP du système HOST à dés-enregistrer.
param3 = -
param4 = -

Commande RF_FRAME_SENDING.

Cette commande permet au système HOST d'envoyer des ordres radio qui sont alors émis par ZIBASE.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.

command = decimal 11
param1 = 0
param2 =
D7..0 : ACTION ,
D15:8 : PROTOCOL,
D23:16 : DIM/BRIGHT,
D31:24 : NB BURST

- ACTION :
OFF: 0
ON: 1
DIM/BRIGHT: 2 (DIM/BRIGHT, CHACON/ZWAVE ONLY)
ALL_LIGHTS_ON: 4 // X10, not implemented
ALL_LIGHTS_OFF: 5 // X10, not implemented
ALL_OFF: 6 (X10 ONLY) // X10, not implemented
ASSOC : 7 // association mode with broadcasted packet over default protocols when protocol=0, or // association mode targeted to the specified 'protocol" when <>0
UNASSOC : 8 unassociation mode (Zwave Only)

- PROTOCOL
DEFAULT BROADCAST (X10, CHACON) : 0
VISONIC433: 1, (frequency : device RF LOW, 310..418..433Mhz band))
VISONIC868: 2,
CHACON (32bits frame) (ChaconV2/DIO series) : 3
DOMIA (24 bits frame) (Chacon V1 + low-cost devices): 4
RF X10 : 5
ZWAVE : 6
RFS10/TS10 : 7
XDD433 alrm : 8
XDD868 alrm : 9
XDD868 inter/shutter : 10
XDD868 Pilot Wire : 11
XDD868 Boiler/AC : 12

- DIM/BRIGHT (needed only for DIM/BRIGHT):
0...100 : 0..100% (CHACON/ZWAVE ONLY)

- NB BURST (optional,set 0 by default)
0: default : only 1 burst
1..5 : number of RF packets burst.
Typically 1 burst means 5...8 RF packets.
(LBT 'Listen Before Talk' is done between each burst)

param3 = B7..0 : value 0-15 (corresponding to 1-16 device number of X10 format),
X10 format mapped to VISONIC/CHACON addresses (ZiBASE Ethernet MAC address on VISONIC/CHACON MSBs)

param4 = B7..0 : value 0-15 (corresponding to A-P house code of X10 format),
X10 format mapped to VISONIC/CHACON addresses(ZiBASE Ethernet MAC address on VISONIC/CHACON MSBs)

Commande SCENARIO_LAUNCHING.

Cette commande permet au système HOST de lancer l'exécution de scénarios qui ont été préalablement enregistrés sur ZiBASE à partir du configurateur.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.

L'association 'nom du scenario' <-> numéro de scénario est bijective et peut être obtenue par appel sur la plateforme Zodianet (voir Schéma B).

command = decimal 11
param1 = 1
param2 = Scenario number
param3 = -
param4 = -

Commande SET_EVENT.

Cette commande permet au système HOST de positionner des alertes à l'état ON / OFF ou de simuler l'arriver d'un ID détecteur.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.

command = decimal 11
param1 = 4
param2 = action
param3 = ID
param4 = ev_type

action :
- 0 : inactiver une alerte
- 1 : activer une alerte
- 2: simuler l'arrivée d'un ID de détecteur (peut entraîner l'exécution de scénarios)

ev_type :
- AXX_PXX_ON = 4
- AXX_PXX_OFF = 9
- AXX_PXX_ZW_ON = 19
- AXX_PXX_ZW_OFF = 20
  ID : 0-255 : X10 adr ou Zwave adr
- Others values :
  ID: 0-(2**32-1) : detector ID

Commande READ/WRITE_VARIABLE/CALENDAR/X10.

Cette commande permet au système HOST de lire/écrire (READ/WRITE) la valeur d'une variable ou d'un calendrier dynamique.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.

command = decimal 11
param1 = 5
param2 = valeur à écrire (sur WRITE)
param3 = type d'opération
param4 = numéro de variable ou calendrier

valeur à écrire (sur WRITE) :
variable : 16 bits signés
calendrier :
B0…23 : heures pointées
B24…30 : jours pointés
B31 :0

type d'opération (variable=VAR, calendrier = CAL, X10 state=X10):

```
READ_VAR          0
TYPE_WRITE_VAR    1
```

```
READ_CAL          2
WRITE_CAL         3
READ_X10          4
```

numéro de variable ou calendrier dynamique ou identifiant X10 :
variable (V0…V31) : 0…31
calendrier (1…16) : 0…15
X10 (A…P, 1…16) : 0…255 (B7…B4: house code, B3…B0 : device)

Les opérations READ entraînent, comme toute autre commande, la transmission par ZiBASE d'un paquet UDP «Réponse ACK » mais contenant la valeur de la variable ou du calendrier dynamique pointée par la commande initiatrice READ/WRITE_VARIABLE/CALENDAR_X10.
paramètres de retour :
param1 = valeur de la variable ou du calendrier ou état de l'actionneur X10 (OFF:0/ON :1)
param2 = type d'opération (`READ_VAR / READ_CAL / READ_X10`)
param3 = = numéro de variable ou calendrier ou identifiant X10.
param4 = -

Commande VIRTUAL_PROBE_EVENT.

Cette commande permet au système HOST d'envoyer dans ZiBASE une information de sonde *virtuelle* comme si celle-ci était reçue sur la RF.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.
command = decimal 11
param1 = 6
param2 = Sensor ID (e.g. 4196984322)
param3 =
B15...0 : analog value 1
B23..16 : analog value 2
B31..24 : Low batt =1 (B26)

param4 =
17: Scientific Oregon Type (OS)
20: OWL Type (WS)

Commande PLATFORM_CHANGE.

Cette commande permet à ZiBASE de changer de plateforme de référence.
Il n'implique pas l'enregistrement préalable du système HOST auprès de ZiBASE.
command = decimal 11
param1 = 7
param2 = password in (32 bits) // to enter the new platform. Set 0 if no password
param3 = password out (32 bits) // to leave the current platform. Set 0 if no password
param4 = new platform number : (8 bits)
0 : **Zodianet.** Server: zibase.net. Set by default. Passwords in/out : 0/0
1 : reserved
2 : **Planète domotique**. Server: planete-zb.net Passwords in/out : 0/0
3 : **Domadoo**. Server: domadoo-zb.net Passwords in/out : 0/0

NB :
- Clicking RESET button during power-on resets ZiBASE to server 0 (only if no 'password out' on the current ZiBASE server)
- Server Number is memorized in ZiBASE EEPROM. Thus, ZIBASE restarts on the last specified server.
- Password error freezes Zapi during 15 seconds.

Commande COMMAND_SCRIPT_LAUNCHING.

Un champ texte de 96 octets exactement est concaténé à ZAPI_packet_header. Il contient une ligne de commande terminée par un caractère null (0).
command = decimal 16
param1 = -
param2 = -
param3 = -
param4 = -

Contenu du champ texte :
cmd : <script de commande>

exemple :

cmd: lm [toto]	lancement du scénario de label toto *// doit être validé dans le configurateur*
cmd: lm 2 aft 3600	lancement du scénario toto dans une heure
cmd: lm [toto].lm [tata]	lancement du scénario toto puis tata *// doit être validé dans le configurateur*

2.1.3 Paquets envoyés par ZiBASE vers le système HOST

Réponse ACK

Un paquet d'acquittement est envoyé après chaque commande du système host.
L'acquittement est envoyé *à l'expéditeur* de la commande. Cet expéditeur peut parfaitement être différent du système HOST enregistré pour recevoir le flux RF_FRAME_RECEIVING.
command = 14
param1…4 = -
Tous les paquets ACK portent l'identifiant de ZiBASE (zibase_id) et l'adresse IP du paquet ACK révèle bien sûr celle de ZiBASE.

Commande RF_FRAME_RECEIVING

Une fois le système HOST enregistré, ZiBASE lui notifie, « au fil de l'eau », les événements radio via des packets UDP de structure ZAPI_packet_receiving. Ces derniers consistent en la concaténation d'une entête standard avec une chaîne de caractères de taille variable.
command = 3
param1…4 = -
Les émissions de paquets RF_FRAME_RECEIVING forment le flux temps-réel de l'activité radio de ZiBASE.

```
struct ZAPI_packet_receiving {
    struct ZAPI_packet_header ;
    unsigned char text_field[variable size 1..400] ;
};
```

Structure du champ text_field
Le système HOST ne doit décoder que les chaines de caractères débutant par :
"Received radio ID "
Les champs utiles sont balisés comme specifiés ci-dessous.
Différentes balises pouvant être rencontrées (notation printf C):

Cas général :

"%s(<rf>%s</rf> <lev>%u</lev>/5 %s Batt=<bat>%s</bat>): <id>%s</id>"

Sonde non reconnue sur un protocole connu:

"<dev>Unknown %s</dev> "
"Frame=<fra>" …. "</fra>"

Digimax TS10:

"<dev>TS10</dev> T=<tem>%u</tem>℃ TC=<temc>%u</tem c>℃ <sta>%s</sta>"

(sta=ON/OFF)

OWL CM119/CM130:

"<dev>%s%s</dev> Total Energy=<kwh>%u.%u</kwh>kWh Power=<kw>%u.%u</kw>kW"

Oregon pluie:

"<dev>%s%s</dev> Total Rain=<tra>%u</tra>mm Current Rain=<cra>%u</cra>mm/hour"

Oregon UV :

"<dev>%s%s</dev> <uv>UV</uv> Level=<uvl>%u</uvl>"

Oregon Vent :

"<dev>%s%s</dev> Avg.Wind=<awi>%u.%u</awi>m/s Dir.=<dir>%u</dir>°"

Oregon Thermomètre/Hygromètre :

"<dev>%s%s</dev> T=<tem>%c%u.%u</tem>℃ %s"
"Humidity=<hum>%u</hum>%% "

Exemple de paquet de commande RF_FRAME_RECEIVING:
Le paquet est émis par ZiBASE000e12 d'adresse IP 192.168.1.192 vers le système HOST d'adresse IP 192.168.1.201 et de port 17100 comme spécifié dans l'exemple sur l'enregistrement.

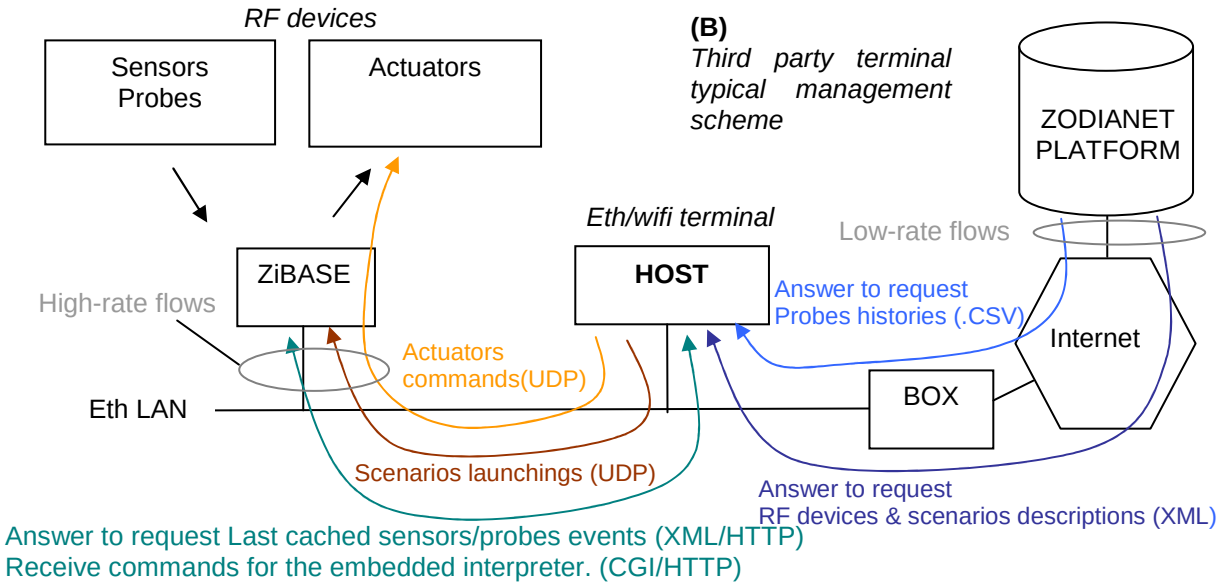
Trace Wireshark :

```
⊞ Frame 94 (248 bytes on wire, 248 bytes captured)
⊞ Ethernet II, Src: Zodianet_00:0e:12 (00:18:7f:00:0e:12), Dst: D-Link_c5:77:7f (00:50:ba:c5:77:7f)
⊞ Internet Protocol, Src: 192.168.1.192 (192.168.1.192), Dst: 192.168.1.201 (192.168.1.201)
⊞ User Datagram Protocol, Src Port: 50387 (50387), Dst Port: 17100 (17100)
⊟ Data (206 bytes)
    Data: 5A534947000354455854 4D534700C83EB7ADEFC3DEAE5A69...

0000  00 50 ba c5 77 7f 00 18  7f 00 0e 12 08 00 45 00   .P..w... ......E.
0010  00 ea 13 fa 00 00 40 11  e1 2f c0 a8 01 c0 c0 a8   ......@. ./......
0020  01 c9 c4 d3 42 cc 00 d6  24 66 5a 53 49 47 00 03   ....B... $fZSIG..
0030  54 45 58 54 4d 53 47 00  c8 3e b7 ad ef c3 de ae   TEXTMSG. .>......
0040  5a 69 42 41 53 45 30 30  30 65 31 32 00 00 00 00   ZiBASE00 0e12....
0050  c0 a8 01 c0 78 aa ac 21  00 00 0e 33 00 00 00 88   ....x..! ...3....
0060  00 00 00 05 00 00 00 01  00 00 00 00 08 46 0b 2f   ........ .....F./
0070  52 65 63 65 69 76 65 64  20 72 61 64 69 6f 20 49   Received  radio I
0080  44 20 28 3c 72 66 3e 34  33 33 4d 68 7a 3c 2f 72   D (<rf>4 33Mhz</r
0090  66 3e 20 3c 6c 65 76 3e  34 3c 2f 6c 65 76 3e 2f   f> <lev> 4</lev>/
00a0  35 20 3c 64 65 76 3e 4f  72 65 67 6f 6e 20 54 48   5 <dev>O regon TH
00b0  20 56 31 2e 30 3c 2f 64  65 76 3e 20 54 3d 3c 74    V1.0</d ev> T=<t
00c0  65 6d 3e 2b 32 32 2e 36  3c 2f 74 65 6d 3e b0 43   em>+22.6 </tem>.C
00d0  20 20 42 61 74 74 3d 3c  62 61 74 3e 4f 6b 3c 2f     Batt=< bat>Ok</
00e0  62 61 74 3e 29 3a 20 3c  69 64 3e 4f 53 36 38 36   bat>): < id>OS686
00f0  31 30 3c 2f 69 64 3e 00                            10</id>.
```

3 GESTION DE TERMINAUX DE TIERCES PARTIES (SCHEMA B)

Le terminal HOST pilote ZiBASE sur le LAN comme le gestionnaire domotique du schéma A. Il en reprend toutefois quelques échanges : La commande d'actionneurs et le lancement de scénarios. Se reporter au schéma A pour en connaître les détails. Mais il n'est pas tenu d'acquérir un flux temps réel d'événements radio pour connaître l'état des appareils RF, il peut donc gérer un état de veille et, à son réveil solliciter ZIBASE pour obtenir une « photo » des états des appareils RF. Pour cela il demande à ZiBASE une table d'états qui lui est retournée sous forme d'un fichier XML.



3.1 Informations fournies par ZIBASE par http

3.1.1 Page xml des données temps réel de capteurs et sondes

La requête s'effectue par l'URL : http://zibase_ip/sensors.xml
Elle doit être faite à partir du LAN, car ZiBASE est susceptible de vérifier que le demandeur se situe bien sur le LAN et la requête peut être rejetée dans le cas contraire.

La balise *x10tab* fournit en hexadécimal un bitmap b0…b255 (BYTE0—BYTE31) des états X10 (A1=bit 0 de l'octet 0, P16=bit7 de l'octet 31).

La balise *vars* fournit les valeurs des variables de ZiBASE.

La balise *cals* vars fournit les valeurs des calendriers dynamiques de ZiBASE (en hexa)

La balise *idx* fournit le nombre de périphériques qui vont être listés dans *evs* .

Balise *evs*
Les données retournées comporte l'heure GMT de réception (Notation UNIX http://www.onlineconversion.com/unix_time.htm) , le type périphérique (normalement redondant avec le protocole), le protocole (OS :oregon,WS :OWL,VS :Visonic,VR :télécommande Visonic,XS : X10 sécurisé,X10_0N : X10 on,X10_OFF : X10 off, CS : Chacon, TS : Digimax TS10), un champ id (à concatener avec le type pour retrouver l'ID dans sa forme habituelle, exemple OS68610), et les grandeurs variables v1 et v2 (appelées I0 et I1 dans le calculateur arithmétique et logique de ZIBASE) significatives pour les sondes. Elles représentes des valeurs entières qui doivent être divisées par 10 sur V1 dans le cas de la température sur Oregon OS et TS10 (précision 0,1℃) , le vent (précision 0,1m/s) etc… L'aide contextuelle du calculateur arithmétique et logique précise ces ajustements.

Exemple de requête

http://192.168.1.192/sensors.xml

Exemple de réponse


```
<?xml version="1.0" encoding="ISO-8859-1"?>
<doc zb="ZiBASE000001">
<x10tab>0000000000000000000000000000000000000000000000000000000000000000</x10tab>
<zwtab>0000000000000000000000000000000000000000000000000000000000000000</zwtab>
<zwpre>0000000000000000000000000000000000000000000000000000000000000000</zwpre>
<zwerr>0000000000000000000000000000000000000000000000000000000000000000</zwerr>
<zwlowbatt>0000000000000000000000000000000000000000000000000000000000000000</zwlowbatt>
<vars>
<var num="0" val="0"/>
<var num="1" val="0"/>
<var num="2" val="0"/>
<var num="3" val="0"/>
<var num="4" val="0"/>
<var num="5" val="0"/>
<var num="6" val="0"/>
<var num="7" val="0"/>
<var num="8" val="0"/>
<var num="9" val="0"/>
<var num="10" val="0"/>
<var num="11" val="0"/>
<var num="12" val="0"/>
<var num="13" val="0"/>
<var num="14" val="0"/>
```

```
<var num="15" val="227"/>
<var num="16" val="179"/>
<var num="17" val="0"/>
<var num="18" val="216"/>
<var num="19" val="190"/>
<var num="20" val="-1"/>
<var num="21" val="-1"/>
<var num="22" val="-1"/>
<var num="23" val="-1"/>
<var num="24" val="-1"/>
<var num="25" val="-1"/>
<var num="26" val="-1"/>
<var num="27" val="-1"/>
<var num="28" val="-1"/>
<var num="29" val="173"/>
<var num="30" val="217"/>
<var num="31" val="16"/>
</vars>
<cals>
<cal num="1" val="0x61AAA555"/>
<cal num="2" val="0x06FC01CF"/>
<cal num="3" val="0x18000FFF"/>
<cal num="4" val="0x63656863"/>
<cal num="5" val="0x786F626B"/>
<cal num="6" val="0xFFFF007D"/>
<cal num="7" val="0xFFFFFFFF"/>
<cal num="8" val="0xFFFFFFFF"/>
<cal num="9" val="0xFFFFFFFF"/>
<cal num="10" val="0xFFFFFFFF"/>
<cal num="11" val="0xFFFFFFFF"/>
<cal num="12" val="0xFFFFFFFF"/>
<cal num="13" val="0x7F000FFF"/>
<cal num="14" val="0x07FF1040"/>
<cal num="15" val="0x108D9CC3"/>
<cal num="16" val="0x68000F01"/>
</cals>
<idx>22</idx>
<evs>
<ev type="7" pro="OS" id="2865505025" gmt="1324232668" v1="0" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="66562" gmt="1324232718" v1="169" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3393760003" gmt="1324232670" v1="197" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="4196985089" gmt="1324232686" v1="207" v2="39" lowbatt="0"/>
<ev type="7" pro="OS" id="1517122816" gmt="1324232722" v1="219" v2="147" lowbatt="0"/>
<ev type="7" pro="OS" id="3391925250" gmt="1324232678" v1="195" v2="38" lowbatt="0"/>
<ev type="5" pro="TS" id="3621322752" gmt="1324232410" v1="210" v2="26" lowbatt="0"/>
<ev type="7" pro="OS" id="2597069569" gmt="1324232720" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="614854176" gmt="1324232394" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="2691855674" gmt="1324232500" v1="0" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="4196967427" gmt="1324232674" v1="198" v2="39" lowbatt="0"/>
<ev type="14" pro="DT" id="2115306752" gmt="1324232598" v1="130" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3670811393" gmt="1324232508" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="382599424" gmt="1324232572" v1="17" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3402375937" gmt="1324232562" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="2115306754" gmt="1324232712" v1="130" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3133940481" gmt="1324232614" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="1904781178" gmt="1324232692" v1="0" v2="0" lowbatt="1"/>
<ev type="14" pro="DT" id="2115306753" gmt="1324231744" v1="130" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="2479940384" gmt="1324232372" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="412385537" gmt="1324231502" v1="1" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="1962588282" gmt="1324231724" v1="0" v2="0" lowbatt="0"/>
</evs>
</doc>
```


<x10tab><zwtab><zwpre><zwerr><zwlowbatt> : ce sont des bitmaps de 256 bits = 256/8 = 32 Octets byte0-31 représentant les adresses pseudo X10 : A1…A16 ---- P1…P16
byte0 = A1(lsb)…A8(msb), byte1 = A9…A16 … byte31 = P9…P16

x10tab = Etat des actionneurs X10/Chacon etc… (1 = ON, 0 =OFF))
zwtab = Etat des périphériques ZWAVE (1 = ON, 0 =OFF))
zwpre = Etat de déclaration dans le contrôleur ZWAVE des périphériques ZWAVE (1= déclaré, 0 non déclaré)
zwerr = Etat des états de transmission sur les périphériques ZWAVE (1 = error)
zwlowbatt = Etat bas de la batterie sur les périphériques ZWAVE (1 = Lowbatt)

3.2 Commandes envoyées à ZIBASE par http

3.2.1 Commandes soumises à l'interpréteur de commandes

La requête s'effectue par l'URL : http://zibase_ip/cgi-bin/domo.cgi?cmd=xxxxxxxxxxxxx
Elle doit être faite à partir du LAN, car ZiBASE est susceptible de vérifier que le demandeur se situe bien sur le LAN et la requête peut être rejetée dans le cas contraire.
Exemples :
Allumage d'un périphérique d'adresse C1 de protocole par défaut
zibase_ip/cgi-bin/domo.cgi?cmd=ON C1

Allumage d'un périphérique d'adresse C1 de protocole 2
zibase_ip/cgi-bin/domo.cgi?cmd=ON C1 P2

Extinction d'un périphérique d'adresse C2 de protocole 3
zibase_ip/cgi-bin/domo.cgi?cmd=OFF C2 P3

DIM à 25% d'un périphérique d'adresse C3 de protocole 3
zibase_ip/cgi-bin/domo.cgi?cmd=DIM C3 P3 25

Lancement d'un scenario N°5 puis allumage d'un péri phérique
zibase_ip/cgi-bin/domo.cgi?cmd=LM 5.ON C1 P2

3.3 Généralités sur les échanges avec la plateforme Zodianet.

Le terminal HOST s'authentifie auprès de la plateforme par l'identifiant principal de ZiBASE (ex ZiBASE000123456) et un numéro de token.
Ce numéro est dérivé du mot de passe utilisateur. Token = f(mot de passe). La fonction 'f' est non bijective et le numéro de token ne permet donc pas de retrouver le mot de passe. Le numéro de token est fourni dans le configurateur dans la rubrique EXPERT/ZiBASE/SYSTEM.
Il est important que le terminal Host enregistre ou « mette en cache » autant que possible les informations fournies par la plateforme pour minimiser le flux échangé.
Dans le cas contraire :

- Zodianet sera contraint de mettre en place une police pour restreinte la périodicité d'accès à la plateforme (ce qui entraînera de facto le dysfonctionnement des terminaux HOST qui sollicitaient trop la plateforme) ou,
- De rendre payant l'accès au service.

De manière générale, il est acceptable que l'action d'un utilisateur soit à l'origine d'une requête (car il est probable que son activité demeure limitée et apériodique) et que les données soient « cachées » dans le terminal (fonction d'antémémoire). Il n'est pas acceptable que cela soit un processus informatique qui sollicite de manière périodique et systématique 24H/24 la plateforme Zodianet pour, qui plus est, acquérir des données qui ne sont en rien temps-réel.

3.4 Informations fournies et services offerts par la plateforme

3.4.1 Identification des requêtes

Il y a 2 méthodes :

- par login/password (exemple login=demo&password=toto)
- par device/token (exemple : device=ZiBASE0032581&token=182b090fad)

3.4.2 Récupération du token

Faire :
https://zibase.net/m/get_iphone.php?login=xxxx&password=yyyy
qui renvoie :
device:token

Le token est invariant mais change si l'utilisateur change son mot de passe. Il permet de s'affranchir de l'envoi de mot de passe dans des sessions ultérieures http, moins sécurisées que https.

Exemple :
ZiBASE0e0001:21fa1234

3.4.3 Récupération du fichier .csv d'historique de sondes

Le terminal HOST peut connaître l'historique des mesures de sondes. Un fichier .csv qui lui est retourné, est identique à celui disponible à partir de PCPAD.

La requête s'effectue par l'URL racine 'http://zibase.net/m/temperature_csv.php'

suivi des identifiants '?device=yyyyyyyy&token=xxxxxxx'

Exemple de requête

http://zibase.net/m/temperature_csv.php?device=ZiBASE0032581&token=182b090fad

Exemple de réponse

HTTP/1.1 200 OK
Server: Apache
Content-Disposition: attachment; filename=zibase_2010_07_18_18H_02M_18S.csv
Cache-Control: max-age=1209600
Expires: Sun, 01 Aug 2010 16:02:18 GMT
Connection: close
Transfer-Encoding: chunked
Content-Type: text/csv

2558
Data per day
Id,date,val1_average,val1_delta,val2_average,val2_delta
OS3930873602,15,23,0.200001,0,0
OS3930873602,16,22.3955,2.3,0,0
OS3930873602,17,21.8913,3.3,0,0
OS3930875138,1,21.1125,3.6,0,0
OS4196930818,1,21.6125,2.8,62.375,18
OS4196962562,15,23.05,0.300001,47.5,1
OS4196962562,16,22.8318,3,55.5909,14
OS4196962562,17,22.387,3.1,49.087,13
OS68610,1,24.2364,5,0,0
OS68610,15,23.8,0,0,0
OS68610,16,23.1455,2.2,0,0
OS68610,17,23.0565,1.1,0,0
WS132195,1,533.136,17.1,0.563636,0.8
WS132195,15,830.85,1.29999,1.2,0.2
WS132195,16,840.536,18.7,0.740909,1
WS132195,17,863.765,21.9,0.878261,0.8

Data per hour
Id,timestamp,time,val1,val2
OS3930873602,1279321197,2010_07_17_0H_59M_57S,22.7,0
OS3930873602,1279324805,2010_07_17_2H_00M_05S,22.4,0
[…]
WS132195,1279443553,2010_07_18_10H_59M_13S,885.9,0.5

WS132195,1279447171,2010_07_18_11H_59M_31S,886.7,0.7
WS132195,1279450771,2010_07_18_12H_59M_31S,887.4,0.5
WS132195,1279454399,2010_07_18_13H_59M_59S,889.5,3.9
WS132195,1279457987,2010_07_18_14H_59M_47S,890.7,0.7
WS132195,1279461598,2010_07_18_15H_59M_58S,891.6,0.9
WS132195,1279465180,2010_07_18_16H_59M_40S,892.7,1
WS132195,1279468767,2010_07_18_17H_59M_27S,893.8,1

0

3.4.4 Récupération du fichier sensors.xml par intermédiation de la plateforme

http://zibase.net/m/get_xml_sensors.php?device=ZiBASExxxxxx&token=xxxxxx
ou
https://zibase.net/m/get_xml_sensors.php?device=ZiBASExxxxxx&token=xxxxxx

Exemple de requête

http://zibase.net/m/temperature_csv.php?device=ZiBASE000001&token=98765432

Exemple de réponse


```
<doc text="ok" zb="ZiBASE000001">
<x10tab>0000000000000000000000000000000000000000000000000000000000000000000000</x10tab>
<zwtab>0000000000000000000000000000000000000000000000000000000000000000000000</zwtab>
<zwpre>0000000000000000000000000000000000000000000000000000000000000000000000</zwpre>
<zwerr>0000000000000000000000000000000000000000000000000000000000000000000000</zwerr>
<zwlowbatt>0000000000000000000000000000000000000000000000000000000000000000000000</zwlowbatt>
<vars>
<var num="0" val="0"/>
<var num="1" val="0"/>
<var num="2" val="0"/>
<var num="3" val="0"/>
<var num="4" val="0"/>
<var num="5" val="0"/>
<var num="6" val="0"/>
<var num="7" val="0"/>
<var num="8" val="0"/>
<var num="9" val="0"/>
<var num="10" val="0"/>
<var num="11" val="0"/>
<var num="12" val="0"/>
<var num="13" val="0"/>
<var num="14" val="0"/>
<var num="15" val="227"/>
<var num="16" val="179"/>
<var num="17" val="0"/>
<var num="18" val="216"/>
<var num="19" val="190"/>
<var num="20" val="-1"/>
<var num="21" val="-1"/>
<var num="22" val="-1"/>
<var num="23" val="-1"/>
<var num="24" val="-1"/>
<var num="25" val="-1"/>
<var num="26" val="-1"/>
<var num="27" val="-1"/>
<var num="28" val="-1"/>
<var num="29" val="173"/>
<var num="30" val="217"/>
<var num="31" val="16"/>
</vars>
<cals>
<cal num="1" val="0x61AAA555"/>
<cal num="2" val="0x06FC01CF"/>
<cal num="3" val="0x18000FFF"/>
<cal num="4" val="0x63656863"/>
<cal num="5" val="0x786F626B"/>
<cal num="6" val="0xFFFF007D"/>
<cal num="7" val="0xFFFFFFFF"/>
<cal num="8" val="0xFFFFFFFF"/>
```

```
<cal num="9" val="0xFFFFFFFF"/>
<cal num="10" val="0xFFFFFFFF"/>
<cal num="11" val="0xFFFFFFFF"/>
<cal num="12" val="0xFFFFFFFF"/>
<cal num="13" val="0x7F000FFF"/>
<cal num="14" val="0x07FF1040"/>
<cal num="15" val="0x108D9CC3"/>
<cal num="16" val="0x68000F01"/>
</cals>
<idx>22</idx>
<evs>
<ev type="7" pro="OS" id="2865505025" gmt="1324232668" v1="0" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="66562" gmt="1324232718" v1="169" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3393760003" gmt="1324232670" v1="197" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="4196985089" gmt="1324232686" v1="207" v2="39" lowbatt="0"/>
<ev type="7" pro="OS" id="1517122816" gmt="1324232722" v1="219" v2="147" lowbatt="0"/>
<ev type="7" pro="OS" id="3391925250" gmt="1324232678" v1="195" v2="38" lowbatt="0"/>
<ev type="5" pro="TS" id="3621322752" gmt="1324232410" v1="210" v2="26" lowbatt="0"/>
<ev type="7" pro="OS" id="2597069569" gmt="1324232720" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="614854176" gmt="1324232394" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="2691855674" gmt="1324232500" v1="0" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="4196967427" gmt="1324232674" v1="198" v2="39" lowbatt="0"/>
<ev type="14" pro="DT" id="2115306752" gmt="1324232598" v1="130" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3670811393" gmt="1324232508" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="382599424" gmt="1324232572" v1="17" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3402375937" gmt="1324232562" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="2115306754" gmt="1324232712" v1="130" v2="0" lowbatt="0"/>
<ev type="7" pro="OS" id="3133940481" gmt="1324232614" v1="0" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="1904781178" gmt="1324232692" v1="0" v2="0" lowbatt="1"/>
<ev type="14" pro="DT" id="2115306753" gmt="1324231744" v1="130" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="2479940384" gmt="1324232372" v1="0" v2="0" lowbatt="0"/>
<ev type="14" pro="DT" id="412385537" gmt="1324231502" v1="1" v2="0" lowbatt="0"/>
<ev type="2" pro="VS" id="1962588282" gmt="1324231724" v1="0" v2="0" lowbatt="0"/>
</evs>
</doc>
```


3.4.5 Récupération de la page xml des descripteurs de périphériques et scénarios déclarés

Récupération des descripteurs

Le terminal HOST peut connaître la description des périphériques RF et des scénarios déclarés comme « visibles ».
Pour chaque périphérique RF, il s'agit essentiellement :

- du label du périphérique qui a été choisi par l'utilisateur,
- de l'association avec son l'identifiant système (ex : OS1234567)
- du logo applicatif associé (ex : vent, thermomètre, détection de mouvement…)
- du répertoire éventuel 'f'

Pour chaque scénario, il s'agit essentiellement :

- du label du scénario qui a été choisi par l'utilisateur,
- d'un numéro représentatif du scenario
- du répertoire éventuel

La requête s'effectue par l'URL racine 'http://zibase.net/m/get_xml.php',
suivi des identifiants '?device=yyyyyyyy&token=xxxxxxx'

Exemple de requête

http://zibase.net/m/get_xml.php?device=ZiBASE0032581&token=182b090fad
https://zibase.net/m/get_xml.php?device=ZiBASE0032581&token=182b090fad

Exemple de réponse (compte demo/demo)


```
<r>
–
<e t="rain" i="logotype_rain.png" c="OS706339840" f="Extérieur">
<n>Pluie Orsay</n>
</e>
–
<e t="transmitter" i="logotype_Presence.png" c="VS614725410" f="Intrusions">
<n>Intrusion</n>
</e>
–
<e t="transmitter" i="logotype_Fumee.png" c="XS3643391298" f="Alertes domestiques">
<n>Détection fumée </n>
</e>
–
<e t="transmitter" i="logotype_Portes.png" c="XS1234" f="Intrusions">
<n>Ouverture porte2</n>
</e>
–
<e t="transmitter" i="logotype_Gaz.png" c="XS3643391553" f="Alertes domestiques">
<n>Fuite de gaz cuisine</n>
</e>
–
<e t="transmitter" i="logotype_Eau.png" c="XS3643390788" f="Alertes domestiques">
<n>Inondation cave</n>
</e>
–
<e t="receiverXDom" i="logotype_LampesPied.png" c="O2">
<n>Lampe rouge</n>
</e>
–
<e t="receiverXDom" i="logotype_LampesPied.png" c="O3">
<n>Lampe verte</n>
</e>
–
<e t="receiverXDom" i="logotype_LampesPied.png" c="O1">
<n>Lampe bleue</n>
</e>
–
<e t="transmitter" i="logotype_Portes.png" c="VS1961418098" f="Intrusions">
<n>Porte principale</n>
</e>
–
<e t="temperature" i="logotype_temperature.png" c="OS439190273">
<n>Intérieur</n>
</e>
–
<e t="temperature" i="logotype_temperature.png" c="OS66048" f="Extérieur">
<n>Temp. extérieure</n>
</e>
–
<e t="power" i="logotype_power.png" c="WS135122">
<n>Conso électrique</n>
</e>
–
<e t="wind" i="logotype_wind.png" c="OS445195008" f="Extérieur">
<n>Vent à Orsay</n>
</e>
–
<e t="light" i="logotype_light.png" c="OS3665311745" f="Extérieur">
<n>UltraViolets</n>
</e>
–
<m id="1">
<n>Ouverture porte2</n>
</m>
–
<m id="2">
<n>Tout éteindre</n>
</m>
–
```

```
<m id="3">
<n>Tout allumer 60s</n>
</m>
</r>
```


Exemple 2 (avec spécification du protocole p et d'un variateur) :


```
<?xml version="1.0" encoding="UTF-8" ?>
- <r>
- <e t="receiverXDom" i="logotype_general.png" c="P16" p="12">
  <n>boil</n>
  </e>
- <e t="receiverXDom" i="logotype_LampesPlafond.png" c="C3" p="3">
  <n>test dim</n>
  </e>
- <e t="temperature" i="logotype_temperature.png" c="OS439190273" f="thermostat" >
  <n>thermostat_in</n>
  </e>
- <e t="receiverXDom" i="logotype_general.png" c="A8">
  <n>A8</n>
  </e>
- <e t="receiverXDom" i="logotype_general.png" c="G1" m2="Chacon_dim">
  <n>test ge</n>
  </e>
- <m id="1" icon="logoMacro_Scenario.png">
  <n>s chacon</n>
  </m>
- <m id="2" icon="logoMacro_Scenario.png">
  <n>k</n>
  </m>
- <m id="3" icon="logoMacro_Scenario.png">
  <n>karotz</n>
  </m>
  </r>
```


Chemins d'accès aux logos en différentes tailles :
http://zibase.net/zodianet/zodiaWeb/icon1/120pix_gray/
http://zibase.net/zodianet/zodiaWeb/icon1/120pix/
http://zibase.net/zodianet/zodiaWeb/icon1/60pix/
http://zibase.net/zodianet/zodiaWeb/icon1/60pix_gray/

exemple :
http://zibase.net/zodianet/zodiaWeb/icon1/120pix/logotype_wind.png

3.4.6 Envoi de commandes domotiques

Codage de ActionModule :
B0-B7 : fonction OFF (0) / ON (1) / DIM (2)
B8-B15 N°de protocole.
B23-B16 : Valeur de DIM

Voir les exemple ci-dessous.

FONCTION « ON » sur le module d'adresse XX (exemple XX=A1, A2.. P16)

Par la méthode du login/password :

https://zibase.net/m/set_iphone.php?login=demo&password=toto&action=comfort&actionComfort=module&idModule=XX&actionModule=1

par http :
http://zibase.net/m/set_iphone.php?login=demo&password=toto&action=comfort&actionComfort=module&idModule=XX&actionModule=1

Par la méthode du token :
https://zibase.net/m/set_iphone.php?device=zodiabase000123&token=574e369a01&action=comfort&actionComfort=module&idModule=O1&actionModule=1

FONCTION « OFF »
https://zibase.net/m/set_iphone.php?login=demo&password=toto&action=comfort&actionComfort=module&idModule=XX&actionModule=0

FONCTION « DIM »
https://zibase.net/m/set_iphone.php?login=demo&password=toto&action=comfort&actionComfort=module&actionType=receiverXDom&idModule=XX&actionModule=327682

valeur de actionModule =2+ (valeur_du_dim * 65536)
exemple : DIM à 5% : actionModule = 2+ (65536*5) = 327682

Lancer le scénario N°Y
https://zibase.net/m/set_iphone.php?login=demo&password=toto&action=comfort&actionComfort=macro&idMacro=Y

Effacer une alerte d'id XS1234

https://zibase.net/m/set_iphone.php?login=demo&password=toto&action=alert&idAlert=XS1234

FIN.