



# La ToIP Open Source

Degré de difficulté



Le monde de la téléphonie sur IP se porte bien, toujours en plein boum depuis ces dernières années où cette technologie a révolutionné le monde des télécoms. En effet, la téléphonie sur IP offre de nouvelles perspectives encore impossibles avec les systèmes télécoms traditionnels.

La téléphonie IP offre une simplicité d'exploitation par le biais d'interfaces graphiques intuitives que vous avez peut-être déjà pu observer sur les nouvelles solutions ToIP Open Source, je vous propose que l'on voit certaines de ces solutions afin de vous présenter un panel de solutions gratuites en matière de téléphonie sur IP.

De leurs côtés, les opérateurs télécoms migrent tous leurs systèmes vers l'IP et certains opérateurs DATA étendent leurs catalogues avec de multiples offres VOIX pour la plus grande joie de tout le monde. Les professionnels ont rapidement pris conscience de l'importance de la téléphonie sur IP et peuvent maintenant proposer des solutions simples et à moindre coût pour tous. L'exemple typique que tous le monde connaît correspond à toutes les offres \*BOX (LiveBox, FreeBox, NeufBox...) qui vous proposent maintenant des offres *Triple-play*, *quadruple-play* (Internet + TV + Téléphone + GSM Wifi) et bientôt des offres *Penta-play* qui vous permettront de regarder la TV sur votre mobile par le biais d'une simple connexion UMTS/WiMax comme le propose aujourd'hui SFR par exemple.

Du côté des particuliers, la révolution du monde libre suit son cours et nous fait découvrir les premières solutions de ToIP open source permettant d'avoir son propre IPBX à la maison et de bénéficier des services avancés en matière de télécommunication comme dans votre entreprise, voir mieux pour les utilisateurs avancés.

## La VoIP / ToIP

La première remarque à ce sujet consiste à ne pas faire l'amalgame entre la VoIP (*Voice over Internet Protocol*) et la ToIP (*Telephony over Internet Protocol*).

La VoIP correspond à la mise en paquets de la voix en datagramme IP afin de transporter les flux de la voix en temps réel par le biais de la technologie IP, principalement des connexions point à point entre deux interlocuteurs ou plusieurs IPBX qui communiquent entre eux. (<http://www.frameip.com/voip>) La VoIP transportant la voix par le biais de la commutation par paquets possède aussi la contrainte du délai de transit identique à la technologie par circuit dite traditionnelle qui correspond au temps qui s'écoule entre le moment où une personne parle et le moment où son interlocuteur entend la phrase. Le transit de votre voix au travers de multiples équipements, d'internet et à destination de l'étranger ou d'un mobile ne doit pas excéder 300ms, au-delà, votre conversation sera de mauvaise qualité, hachée... C'est pourquoi, les opérateurs s'efforcent de réserver une bande passante minimum pour votre téléphone connecté sur votre \*box par le biais d'une boucle locale ADSL, bien que cette technologie ne propose pas en théorie une qualité de service garantie.

Les avantages de la VoIP sont multiples mais le principal avantage reste le prix car les opérateurs sont aujourd'hui en mesure de vous

## CET ARTICLE EXPLIQUE...

Cet article vous présentera un panel de solutions gratuites en matière de téléphonie sur IP.

## CE QU'IL FAUT SAVOIR...

Les notions sur la téléphonie IP.

proposer des forfaits illimités vers tous les fixes en France et même pour certaines, illimités vers certains pays étrangers.

Lorsqu'on parle de ToIP, cela concerne le périmètre du concept, de l'architecture et des équipements qui permettent d'utiliser la VoIP afin de bénéficier de tous ces avantages et des ces nouvelles fonctionnalités associées à cette technologie.

## Les Principaux protocoles

En termes de téléphonie sur IP, il faut distinguer plusieurs types de protocoles :

- Les protocoles de signalisation,
- Les protocoles de transport de la voix.

Les protocoles signalétiques, ont la charge de régir les communications, de déterminer les appelés, de signaler les appelants, de gérer les absences, les sonneries etc... Mais aussi de négocier quel codec pourra être utilisé.

Les protocoles de transport quand à eux, transportent l'information sur un réseau IP. Ce type de protocoles est spécifique à la voix sur IP et aux applications nécessitant le transit de l'information en temps réel comme par exemple, la vidéo conférence.

Voici une liste des principaux protocoles de VoIP :

- SIP,
- IAX,
- MGCP,
- H.323,
- SCCP, propriétaire Cisco Systems anciennement Skinny,
- UA/NOE, propriétaire Alcatel,
- UNISTIM, propriétaire Nortel.

Le SIP est certainement aujourd'hui le protocole le plus connu par le grand public principalement dû à l'émergence de multiples solutions IPBX Open Source dont nous parlerons un peu plus loin dans cet article.

Cependant, un petit nouveau a fait son entrée sur le marché de la VoIP. L'IAX encore inconnu il y a quelques années, il rattrape son retard en s'affranchissant des problèmes de NAT, il n'est pas une menace pour le SIP qui est maintenant très bien implanté mais il est en train de devenir une alternative fiable.

Les protocoles propriétaires, sont des protocoles développés directement par les constructeurs et qui ne sont pas ouverts (non compatible avec d'autres), ils fonctionnent donc uniquement avec les équipements de leurs fabricants. Chacun des grands constructeurs a créé le sien afin d'adapter leurs équipements et de fidéliser le client sur leurs solutions propriétaires.

## Les Protocoles de signalisation

**H323** : Premier protocole VoIP, issue des technologies téléphoniques, il est développé par l'UIT-T, il est couramment considéré lourd à mettre en place et ayant comme conséquence un prix du marché fort. Ce protocole référencé H323 s'appelle en réalité *Système de communication Multimédia Fonctionnant en mode Paquet* est apparue dans sa version 1 en 1996, il est aujourd'hui disponible en version 6.

H 323 permet la mise en place de réseaux téléphoniques IP étendus. Sa prétendue complexité le réserve à un usage professionnel. Les majeures parties des FAI grand public en France utilisent MGCP et/ou SIP.

**SIP** : Le protocole SIP (*Session Initiation Protocol*) a été publié pour la première fois par l'IETF en 1997, par la RFC 2543. Cette dernière un peu vague a été complétée au cours du temps pour finalement arriver en 2002 à la RFC 3261.

SIP à l'origine était conçue pour être un protocole simple, mais afin d'offrir le même niveau de service que H323, Le protocole s'est enrichie, le rendant plus complexe, même si le principe de base reste, beaucoup plus simple que H323. Le SIP a longtemps été considéré comme le futur de la ToIP, il est en tout cas un des protocoles majeurs sur le marché.

**MGCP** : Le MGCP (*Media Gateway Control Protocol*) est un protocole de VoIP issu des premiers développements du projet *TIPHON (Telecommunication and Internet Protocol Harmonization Over Network)* par Telcordia et Level 3 Communications ensuite validé en 1998 par l'IETF par la RFC 2705. Ce protocole a été créé afin d'offrir une couverture plus large en fédérant toutes les signalisations, qu'elles soient IP ou RTC entre autres. (SS7, H323, SIP...) car les passerelles d'interconnexions en H323

des FAI étaient trop complexes et trop coûteuses.

Ce protocole est fortement utilisé sur le marché par les ISP (*Internet Service Provider*) afin d'assurer le contrôle et l'administration à distance des boîtiers \*Box et il constitue la base des offres IP Centrex des opérateurs.

**IAX/IAX2** : Le protocole IAX (*Inter-Astérisik eXchange*) est issu du projet d'IPBX open source et de la communauté Astérisik, tout comme le SIP, c'est un protocole de VoIP qui permet de communiquer entre client/serveur ou serveur/serveur.

La différence avec le SIP et le point fort de l'IAX, se base sur l'utilisation d'un port UDP unique qui est le port 4569 qui permet de s'affranchir des problématiques de NAT dans votre système télécom.

Lorsque l'on parle d'IAX, on parle maintenant de la version 2 puisque la première version de ce protocole n'est quasiment plus utilisé (IAX 1= UDP 5036). Le protocole IAX n'est pas encore standardisé. L'IAX2 support maintenant l'authentification par certificat (PKI), le mode *Trunking* et possède aussi un *JitterBuffer* plus évolué que SIP, plus adapté aux boucles locales ADSL des particuliers, ce qui permet d'obtenir une légère augmentation du niveau de qualité que le SIP dans ces conditions.

Le seul Bémol de ce protocole est qu'il n'est pas encore standardisé donc le nombre d'équipement supportant ce protocole reste faible.

## Protocole de transport

Aujourd'hui, le couple RTP/RTCP, s'utilise systématiquement dans les applications multimédias interactives, pour la téléphonie, la vidéo, les jeux vidéo et même les premiers simulateurs de réalité virtuelle. Ces protocoles applicatifs sont chargés de transporter une information multimédia en temps réel au travers d'un réseau IP. Cependant, ce couple de protocole n'est pas utilisé pour la réservation des ressources réseaux, ni pour fiabiliser les échanges, ni pour garantir les délais de transit puisque certains paquets peuvent être retardés.

**RTP** : Le RTP (*Real Time Protocol*) créé en 1996 dans la RFC 1889, rendu obsolète et standardisé par l'IETF par la

RFC 3550 en 2003. Le RTP a été conçu pour transporter des flux IP ayant de fortes contraintes temporelles, typiquement, des flux multimédias, il permet de reconstituer les flux IP multimédia en temps réel en agissant à deux niveaux :

- La synchronisation des flux,
- La reconstitution de l'ordre des paquets.

Il est important de noter que les protocoles de signalisation comme les protocoles de transport peuvent être utilisés entre les terminaux IP et l'IPBX, mais aussi directement entre les terminaux, si ces derniers peuvent communiquer en IP, ces solutions présentent des avantages et inconvénients qui sortent du cadre de cet article.

*RTCP* : Le *RTCP (Real Time Control Protocol)* lui aussi publié par l'IETF dans la RFC 3550 en 2003, permet un contrôle des flux RTP afin de garantir leurs intégrités ainsi qu'une supervision du réseau en agissant comme une sonde qui informe l'utilisateur de l'état du réseau en temps réel. Les protocoles *RTP* et *RTCP* sont indépendants mais néanmoins, leur association apporte une cohérence dans le traitement de l'information en temps réel afin d'optimiser les conditions de transport des flux IP multimédia ainsi que la qualité de service générale.

## Les solutions IPBX open source

Un *IPBX* ou *PABX IP (Private Branch Exchange)* est un équipement (autocommutateur) téléphonique capable d'acheminer les appels sur un réseau via l'utilisation du protocole IP. *Internet Protocol* permet à partir de l'identifiant d'une hôte (PC, téléphone) un identifiant appelé une *adresse IP* d'acheminer des flux voix, données et images au sein d'un réseau informatique. Il gère l'établissement des communications entre plusieurs postes à l'intérieur d'une entreprise, ainsi que vers l'extérieur (réseau de public : *PSTN*). Pour établir une communication, les utilisateurs pourront utiliser un softphone (logiciel installé sur un PC) où un hardphone (téléphone « matériel » qui utilise les protocoles de voix sur IP) compatibles avec l'IPBX utilisé. L'IPBX en fonction du

niveau d'intelligence, permettra de fournir l'ensemble des services lié à la gestion de la téléphonie de l'entreprise.

Les *IPBX* offrent les mêmes fonctionnalités que les *PABX* dit traditionnels :

- Serveur vocal interactif (SVI ou IVR),
- Transfert d'appel,
- Numérotation interne,
- Voice-mail,
- Identification du numéro,
- Rejet d'appel,
- Blocage des appels des utilisateurs,
- Historique des appels,
- Mise en attente,
- Renvoi d'appels (sans réponse – occupés),
- Messagerie unifiée,
- Groupement d'appel,
- Musique d'attente,
- Etc...

Le concept est simple, il vous suffit d'un pc tout ce qu'il y a de plus classique sur lequel vous installez une solution de ToIP open source. Vous branchez ensuite votre IPBX maison directement à la place de votre téléphone IP sur votre \*\*\*\*BOX sur le port dédié à la téléphonie qui correspond à un *TrunkSip*. Aujourd'hui, très peu d'opérateur (excepté les offres \*\*\*\*BOX) vous propose des solutions *TrunkSip*, encore hors process et offre non commercialisée pour l'instant dans leurs services, demain vous n'aurez plus de ligne analogique chez vous mais bel et bien un ou plusieurs téléphone IP ainsi qu'un Fax IP.

Le *TrunkSip* n'est pas obligatoire pour utiliser un IPBX, vous pouvez aussi l'interconnecté avec le PSTN par le biais d'une ligne T0 ou T2 mais il vous en coûtera une carte Hardware T0 ou T2 ainsi que sa configuration qui reste assez complexe...

Cette configuration vous permet ainsi de personnaliser vos services de téléphonie sur IP et d'ajouter un grand nombre de poste IP supplémentaire chez vous et par exemple de leur attribuer à tous un numéro interne et des règles de filtrage définit.

Ces règles de filtrage vous permettront par exemple d'empêcher un poste IP précis de ne pas appeler des numéros spéciaux et/ou de mobile pour vos enfants par exemple. Vous pouvez autoriser les

appels internationaux uniquement sur votre poste IP afin d'éviter l'explosion de vos communications.

## Asterisk

Asterisk est la solution IPBX Open source la plus utilisée sur le marché de la téléphonie sur IP à l'heure actuelle. Cette solution a notamment donnée naissance à plusieurs projets dont le but était la conception d'IPBX Open source, même les leaders des télécommunications se sont mis à développer des solutions autour de ce produit. Il est possible de citer certains projets Open Source dont Asterisk à permis de donner naissance comme Bayonne, Callweaver, Freeswitch et d'autres encore.

Ce produit doit sa conception au fondateur de la société Digium Mark Spencer qui en 1999 alors qu'il était étudiant à l'université d'Auburn aux États-Unis (Alabama) décide de concevoir son propre PBX sur la base d'une distribution linux.

Le résultat de cette conception s'appelle Asterisk, outil qui connaît aujourd'hui un grand succès dans son domaine. Le projet initial a été développé autour d'une GNU/Linux sur une plate-forme de type x/86, actuellement cette solution est désormais disponible sur une plus grande quantité de plates-formes. Asterisk est un PBX Open Source qui est basé actuellement sur la licence GPL 2.0. Ci-dessous plusieurs déclinaisons d'Asterisk :

### AsteriskGUI et Free PBX

*AsteriskGUI*, GUI pour *Graphique User Interface (Interface utilisateur graphique)* se trouve être une interface graphique et l'outil d'administration d'*asteriskNOW*. *Free PBX* est aussi une interface d'administration créée pour la gestion des serveurs Asterisk. Ces interfaces permettent à chacun de simplifier l'utilisation



Figure 2. Le logo d'Asterisk

et l'administration de votre IPBX en le rendant plus accessible.

### Asterisk at Home

Cette déclinaison d'Asterisk est destinée à être intégrée au sein de très petites structures comme les réseaux domestiques. L'objectif de cette distribution est de simplifier l'intégration d'un serveur de téléphonie sur IP et de proposer une version *light* d'Asterisk sous forme de package.

### Asterisk for Windows

Cette plate-forme d'Asterisk est une déclinaison destinée à permettre la mise

en œuvre de la solution Asterisk sous Windows pour les allergiques à Linux. Cette déclinaison dispose des mêmes fonctionnalisées que ses homologues sous linux. Asterisk permet de fournir les fonctionnalités classiques d'un PBX. L'administration de la plate-forme Asterisk peut se faire via une interface graphique de type WEB tel qu'AsteriskGUI, l'édition des fichiers de configuration en mode texte ainsi qu'un mode d'administration via une console CLI (*Interface en ligne de commande*).

Concernant la VoIP en elle-même, l'outil est capable de gérer un certain nombre de protocoles de signalisation de téléphonie



Figure 5. Asterisk Win32

sur IP, comme IAX, SIP, H323, MGCP, SCCP. Il est également possible de raccorder des téléphones IP de différents constructeurs à un IPBX Asterisk.

Asterisk a été écrit avec le langage C. Cette solution fonctionne sur des systèmes d'exploitation tels que Linux, OpenBSD, Windows, Mac OS X Jaguar. Asterisk fournit quatre API (*Interface de programmations d'application*) modulaires permettant la modification liée aux interfaces matériels, applications téléphoniques, traitement de format de fichier et des codecs. Asterisk permet de mettre en œuvre un certain nombre de fonctionnalités dont certaines sont citées ci-dessous.

- Fournit un grand nombre de fonctionnalités lié à la gestion des appels comme les transferts, interception, mise en attente des appels, musique d'attente...
- Permet de réaliser des conférences vocales à plusieurs,
- Service d'identification de l'appelant,
- Fournit des services vocaux interactifs (IVR *Interactive Voice Response*),
- Interconnexion au réseau téléphonique traditionnel (FXS/FXO, ISDN, PRI, E1, T1) Cartes Digium,
- Supporte une large gamme de protocoles TDM,
- Gère plusieurs protocoles de Voix sur IP (H.323, IAX, IAX2, MGCP and SIP and SCCP),
- Compatibilité avec plusieurs modèles de téléphones IP (Cisco, Nortel, Polycom...),
- Utilisation de l'*Inter-Asterisk eXchange* (IAX) pour la communication entre deux serveurs Asterisk,
- Ne nécessite pas de plate-forme matérielle spécifique (RJ45), sauf pour l'interconnexion au réseau TDM,
- Possibilité d'intégrer du développement JAVA.

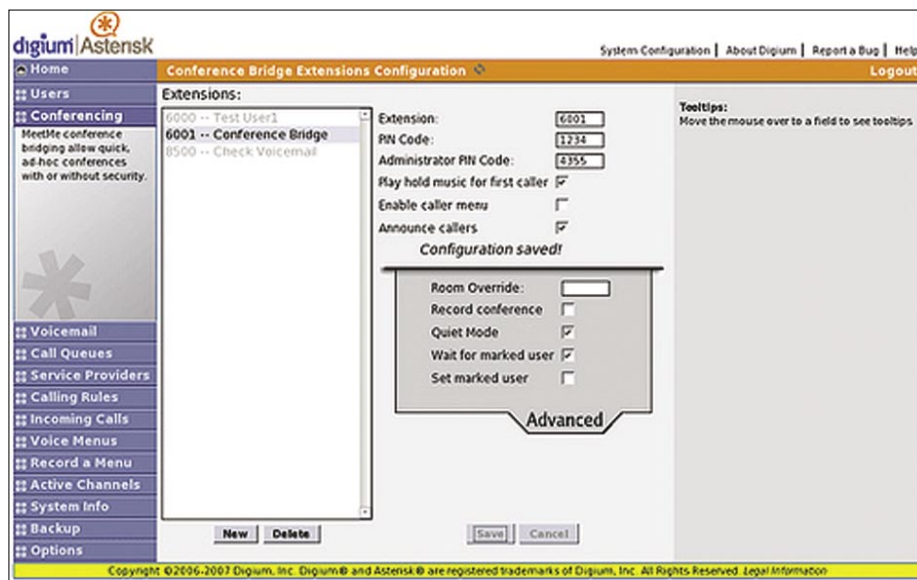


Figure 3. Digium Asterisk

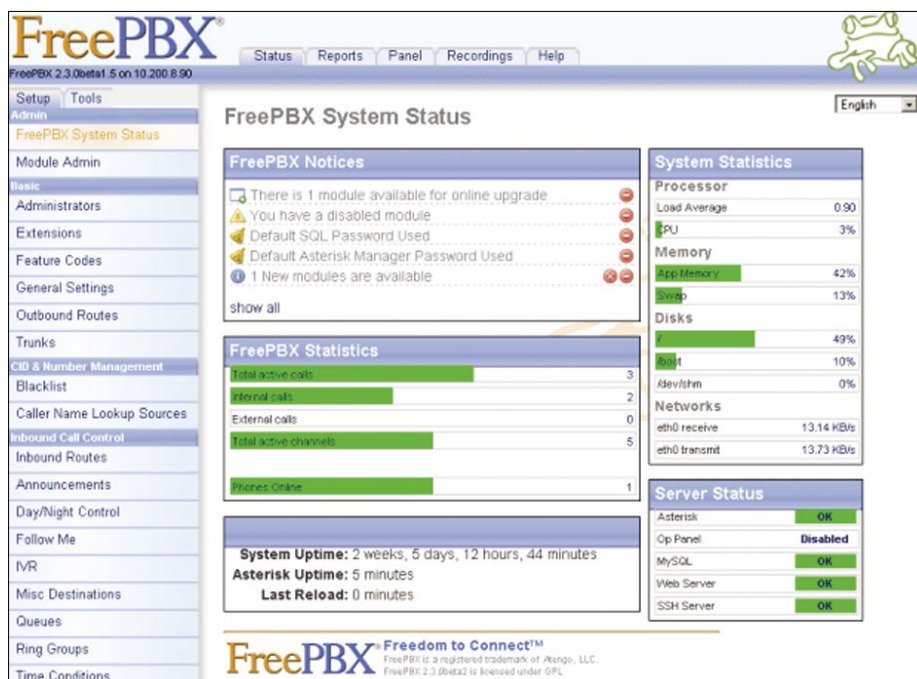


Figure 4. Free PBX

Site officiel du projet Asterisk :

<http://www.asterisk.org>

Site officiel de DIGIUM :

<http://www.digium.com/en/>



Figure 6. SIPfoundry

## SipXecs

SipXecs est une solution IPBX gratuite pouvant être mise en œuvre au sein d'infrastructures de différentes tailles. Elle peut être intégrée dans des infrastructures de très petites tailles à des infrastructures allant jusqu'à 6000 d'après les développeurs de la communauté. Ce produit a pour particularité de supporter uniquement le protocole SIP. À titre d'exemple Nortel Networks un des leaders du marché des télécommunications a annoncé que sa dernière solution de communication unifiée à destination des PME (SCS500) est totalement basée sur le code source de SipXecs. Nortel a également annoncé par la même occasion son entrée dans la communauté du SIPfoundry en tant que contributeur du projet.

Ce produit est prévu pour fournir des services de téléphonie sur IP même au sein de structure de petite taille, avec une utilisation personnelle ou bien même au sein de très petites entreprises souhaitant disposer d'une solution de messagerie unifiée et de téléphonie sur IP.

SipXecs doit sa création à la société *Pingtel Corp* qui réalisa le développement du produit en 1999. Sa création avait un objectif commercial avant d'aboutir en 2004 sur un modèle basé sur l'Open source en participant à la création de la communauté SIPfoundry. SIPfoundry est une communauté Open Source qui

s'est basé sur le standard SIP (*Session Initiation Protocol*) pour développer la solution IPBX SipXecs ayant comme fondement le respect du standard SIP. La solution est basée sur la licence GNU Lesser General Public License 2.1.

Ce produit a été développé à partir des langages C/C++ et basé sur une interface d'administration WEB afin de réaliser la gestion des différents services offerts par le produit tel que le plan de numérotation, les utilisateurs ou bien les téléphones. Ce serveur peut être intégré sur des serveurs standards basés sur des plates-formes Intel, AMD ou power PC, le support de la technologie 64 bit devrait suivre prochainement. Ce produit est capable de réaliser dynamiquement la découverte des équipements comme les téléphones IP. Il permet une intégration complète d'un système de messagerie unifié pour Microsoft Outlook.

Ce produit est prévu pour être couplé à un service d'annuaire de type OpenLDAP ou Active Directory en y ajoutant Microsoft Exchange 2007. SipXecs est conçu pour fonctionner uniquement sur des plates-formes basées sur le système d'exploitation Linux (Centos, Debian, Fedora, Suse, Gentoo et FreeBSD). Actuellement la dernière version stable en ligne de SipXecs est la version 3.10.

Exemple de services que SipXecs est capable de fournir :

- Messagerie vocale,
- Distribution des appels automatisés,
- Messagerie unifiée,
- Trunk RTC,
- Trunk SIP,
- Call Center Server.

Intègre des mécanismes de haute disponibilité et de tolérance de panne.

Interopérabilité avec tous les équipements respectant le standard SIP.

Le site officiel du projet SIPfoundry :

<http://www.sipfoundry.org>  
La liste des équipements compatibles avec la solution (passerelles, Hard phones Softphones...) :  
[http://sipx-wiki.calivia.com/index.php/Phones\\_%26\\_Gateways](http://sipx-wiki.calivia.com/index.php/Phones_%26_Gateways)



Figure 7. SIPXconfig

## CallWeaver

Callweaver est un IPBX qui a été développé autour du projet Asterisk. Ce produit est basé sur une licence de type GPL. Callweaver est capable de s'interfacer sur plusieurs types de réseaux, tel que le raccordement à un réseau téléphonique traditionnel ou IP. Ce produit a été conçu de sorte à ce qu'il puisse gérer un ensemble de protocoles de signalisation de Voix sur IP (H323, IAX2, MGCP, SIP...).

Anciennement connu sous le nom de projet openPBX, celui-ci est devenu le projet Ce projet a été renommé Callweaver étant donné que le nom d'un produit développé par la société Voicetronix portait le même intitulé. Ce produit est né en 2005, Callweaver est dérivé d'une autre plate-forme Open Source, pour cela la solution avec laquelle *le Fork* a été réalisé, fût la version 1.2 d'Asterisk. La raison de la création de cette communauté est liée au fait que les développeurs avaient une vision différente de celle de Digium.

Cette solution étant dérivée d'Asterisk, les fichiers permettant la configuration sont similaires. Un autre moyen permet d'administrer le serveur, comme une interface WEB plus simple pour les adeptes de l'interface graphique. Cette solution utilise la librairie SpanDSP pour la gestion des codecs ainsi que la gestion de la réception et de l'émission des FAX. L'une des plus grosses difficultés liées aux communications SIP par rapport à la NAT est liée à la modification des informations d'adressage durant le transport des informations lors du passage à travers l'équipement réalisant la translation d'adresse (NAT). Le support de *STUN* (*Simple Traversal of UDP trough NATs*) a été développé par un des principaux organismes de standardisation (IETF) afin de permettre la réalisation des communications SIP à travers une NAT (translation d'adresse d'un réseau privé vers un réseau public).

Cette plate-forme est prise en charge par plusieurs types de systèmes d'exploitation (Linux, FreeBSD, NetBSD,



Figure 8. callweaver

OpenBSD, MacOS X/Darwin, Open/Solaris). La version stable actuellement en ligne de Callweaver est la version 1.2.0.1. Ci-dessous une présentation des principales caractéristiques de l'outil :

- Interconnexion au réseau *RTC* (FXS/FXO, ISDN, PRI, E1, T1),
- Gère plusieurs protocoles de Voix sur IP (H.323, IAX2, MGCP and SIP and SCCP),
- Supporte le protocole STUN pour les communications SIP,
- Support du FAX via T.38. (Fax over IP),
- Serveur vocal interactif,
- Gestion des conférences,
- Gestion des fils d'attentes.

Le site officiel du projet Callweaver : <http://www.callweaver.org/wiki/CallWeaver>  
 Pour télécharger Callweaver, vous pouvez le faire via l'adresse ci-dessous : <http://www.callweaver.org/browse>

Pourquoi CallWeaver Forks Asterisk :

- La communauté à le contrôle du projet *Callweaver*, de sorte que pas une seule personne ou entreprise peut arrêter le progrès,
- Utilisation des meilleures bibliothèques disponibles. (par ex SpanDSP, Unicall, Sofia SIP),
- Basé sur Asterisk 1.2, pas de remise en question des acquis du projet d'origine, principalement axé sur l'ajout de modules complémentaire dans le but de développer de nouveaux services avancés,
- Pas de double licence, uniquement GPL,
- Pas d'intérêts commerciaux, communauté de développement libre.

Liste non exhaustive des principales différences avec Asterisk :

- Plus d'utilisation des dépendances Zaptel pour le timing des conférences,
- Séparation du serveur de daemon de la command-line interface (CLI),
- Support STUN (pour SIP NAT traversal & SIP communications),
- Remplacement de Asterisk's core DSP par l'utilisation de *SpanDSP*, plus efficace et supporte T.38 Fax sur IP (STUN),

**Listing 1.** *Freeswitch.xml* standart

```
<?xml version="1.0"?>
<document type="freeswitch/xml">
  <X-PRE-PROCESS cmd="include" data="vars.xml"/>
  <section name="configuration" description="Various Configuration">
    <X-PRE-PROCESS cmd="include" data="autoload_configs/*.xml"/>
  </section>
  <section name="dialplan" description="Regex/XML Dialplan">
    <X-PRE-PROCESS cmd="include" data="dialplan/*.xml"/>
  </section>
  <!-- mod_dingaling is reliant on the vcard data in the "directory" section. -->
  <!-- mod_sofia is reliant on the user data for authorization -->
  <section name="directory" description="User Directory">
    <X-PRE-PROCESS cmd="include" data="directory/*.xml"/>
  </section>
  <!-- phrases section (under development still) -->
  <section name="phrases" description="Speech Phrase Management">
    <macros>
      <X-PRE-PROCESS cmd="include" data="lang/*.xml"/>
    </macros>
  </section>
</document>
```

- Utilisation de SQLite pour sa base de données interne,
- Utilise un *jitterbuffer* universel pour être utilisé avec tous types de canaux,
- Utilisation du *Haching* pour le plan de numérotation donc plus rapide et plus efficace,
- Les variables et des applications dans *extensions.conf* sont sensibles à la casse, script de conversion fournit,
- Renommage de certaines fonctions natives, script de conversion fournit,
- Fournit des scripts de conversion pour convertir les fichiers de configuration chaque fois que des modifications sont apportées à la syntaxe ou le format,
- Support for PostgreSQL en mode temps réel.

**FreeSwitch**

FreeSwitch est une solution open source de téléphonie sur IP, sous une licence *MPL* (*Mozilla Public License*), développé en C. Elle permet la mise en place de communications vers un téléphone virtuel via un commutateur virtuel. FreeSwitch peut être utilisé comme un simple commutateur, un PBX, une passerelle ou un serveur d'applications *IVR* (*Interactive Voice Response*) en utilisant des scripts ou des fichiers XML permettant d'automatiser certaines tâches et de développer de nouveaux services.

FreeSwitch fonctionne sur plusieurs systèmes d'exploitation, notamment Windows, Mac OS X, Linux, BSD et sur les

deux plates-formes Solaris (32 bits et 64 bits). Une Interface Web pour FreeSwitch est disponible sous le nom *Wiki PBX*.

FreeSwitch supporte les caractéristiques standards et avancées du protocole SIP, permettant de mettre en place un serveur de conférence, un serveur de Voicemail... Il utilise aussi les protocoles IAX2, Jingle et H323, et supporte le Fax via T.38.

FreeSwitch nécessite 50 Mo d'espace disque, 512 Mo de Ram et une sortie téléphonique (RJ 45).

Les langages de programmation supportés par cette solution sont :

- Javascript,
- Python,
- Perl,
- Lua (langage de script libre).

La configuration de FreeSwitch peut s'effectuer de deux manières :

- En ligne de commande (CLI),
- En Interface graphique (Web).

Exemple d'un fichier *Freeswitch.xml* standard (voir Listing 1).  
<http://www.wikipbx.org/>  
<http://www.freeswitch.org/>



**Figure 9.** *Freeswitch*

## GNU Bayonne

GNU Bayonne est le serveur d'applications téléphoniques du projet GNU, c'est-à-dire orienté open source basé sur une licence libre. Cette solution offre un environnement gratuit permettant aux petites et grandes infrastructures de développer, de déployer et de gérer des solutions de téléphonie intégrées à leur réseau informatique afin d'exploiter une ou plusieurs lignes téléphoniques. GNU Bayonne propose sur le marché des PBX Open source, un serveur applicatif distribué complet, utilisable avec des cartes téléphoniques multiports de plusieurs fabricants et fonctionnant sur des systèmes d'exploitation libres.

GNU Bayonne2 permet de développer des applications IVR (*Interactive Voice Response*) grâce à un simple langage de script. La version 1.x gère la VoIP grâce au couplage avec le logiciel GNU oSIP Stack.

GNU Bayonne se décline sous deux versions :

- Bayonne 1 : version développée en 1998 pour succéder au système ACS,
- Bayonne 2 : version développée en 2005 avec un accent particulier sur l'utilisation du protocole SIP.

Bayonne est basé sur le projet ACS (*Adjunct Communication Server*). Le projet ACS a été repris par le projet GNU qui œuvre pour développer un système d'exploitation et des logiciels complètement gratuits basés sur Unix. Le nom Bayonne vient du nom du célèbre pont qui relie la ville de Bayonne dans le New Jersey avec l'île de Staten Island dans l'état de New York. L'auteur a ainsi voulu montrer que son logiciel était un pont entre le monde de l'informatique et le monde de la téléphonie.

Bayonne ne possède pas de fonction IP-PBX dans sa version 1. La version 2, prend en compte cette fonctionnalité. Ce projet étant peu suivi par la communauté Internet, il est très difficile donc de trouver



Figure 1. Le logo de Bayonne

de la documentation. Actuellement GNU Bayonne est un projet de petite envergure mais il a le mérite d'avoir fait partie des précurseurs dès 1998.

Le fonctionnement de GNU Bayonne repose sur plusieurs composants que l'on doit lui associer pour pouvoir l'exploiter et le mettre en place.

- GNU Common C++,
- GNU ccScript (Machine virtuelle),
- GNU ccAudio (Gestion des flux audio),
- GNU oSIP Stack (Pile SIP),
- Libhoard (bibliothèque additionnelle),
- Voicetronix PCI (pilote).

Bayonne dispose d'un interpréteur de scripts qui peut être étendu grâce à des applications TGI (*Telephony Gateway Interface*), c'est-à-dire une *Passerelle d'Interfaçage Téléphonique* permettant de simplifier l'intégration de GNU Bayonne.

La solution peut être utilisée aujourd'hui complètement sous GNU/Linux avec une variété grandissante de matériels téléphoniques compatibles. Bayonne est portable et peut être compilé sur la plupart des systèmes d'exploitation.

GNU Bayonne se caractérise par la multitude de services qu'il offre, notamment GNU Bayonne2 qui utilise le protocole SIP et H323 offrant des services avancés IP, bien connu des solutions PBX Open source. Certains utilisateurs de Bayonne avouent que sa configuration est difficile à mettre en place.

La première étape consiste à disposer d'un système d'exploitation open source, par exemple Debian.

La seconde étape consiste à compiler et installer GNU Bayonne2 puis, les modules et enfin les fichiers de configuration pour les différents services que vous souhaitez mettre en place pour configurer vos services.

Voici un exemple de configuration du serveur et du pilote : (voir Listing 2).

<http://www.gnu.org/software/bayonne/>  
<http://www.linuxfocus.org/Francais/April2005/article372.shtml>

## YATE

YATE est un logiciel créé par une communauté originaire de Roumanie, le nom donné à la solution est un acronyme signifiant *Yet Another Telephony Engine*.



Figure 10. Yate

Il a été développé en C++ par la société Null Team qui a été fondée en 2004, après quelques années d'expérience dans le domaine de la téléphonie et de la création de logiciel. Yate se distingue sous deux versions : YATE 1 et YATE 2.

Le développement de YATE est subventionné par la société Sangoma Tech. Cette solution est intégralement compatible avec les cartes d'interface développées par Sangoma Tech, avec une particularité pour YATE 1 qui prend en charge uniquement les cartes de type Sangoma Tech.

YATE peut être utilisé à la fois en tant que client ou serveur. Le mode client est un softphone développé en java. Il peut réaliser la fonction de passerelle entre le réseau public et le réseau IP ou entre un PC et un téléphone, afin de réaliser l'acheminement des communications vers le réseau de l'opérateur.

Yate a été conçu de sorte à permettre son exploitation sur les OS Windows, Linux et Unix. Ce logiciel accepte les scripts dans différents langages de programmation, tel que le PHP, Perl, Python.

Voici les éléments applicatifs que l'on retrouve dans la solution Yate :

- Un outil de supervision : *YateGUI*,
- Une interface Web pour le client (disponible sur Windows et Linux) : *YateClient*,
- Une interface d'administration : *YateAdmin*,
- La possibilité de mettre en place un proxy : *SIP - H323*.

Yate supporte les protocoles :

- H323 (basé sur OpenH323),
- SIP (basé sur une pile SIP propre à YATE),
- IAX et IAX2 (basé sur une partie de libIAX),

· RTP.

Se décline sous différentes fonctionnements comme :

- Serveur de téléphonie sur IP,
- Client VoIP,
- PSTN VoIP Gateway,
- PC2Phone et Phone2PC Gateway,
- Gardien H.323,
- Proxy H.323 ↔ SIP,
- Routeur SIP,
- Serveur d'enregistrement SIP,
- Serveur et/ou client IAX,
- Call center serveur.

On notera une faible documentation existante sur Yate : <http://yate.null.ro/pmwiki/>.

**Synthèse**

Regardez le Tableau 1. Des solutions ont également été développées autour de la plate-forme Asterisk, des équipementiers

tels que ASTRA MATRA ou encore DELL qui s'est lancé dans la conception d'Appliance à destination des TPE et PME.

**Listing 2.** Exemple de configuration du serveur et du pilote

```

/etc/bayonne/server.conf
[server]
user = bayonne
group = bayonne
language = en_us
voice = en/deborah
[engine]
driver = sip
/etc/bayonne/driver.conf
[sip]
inband = true
dtmf = 101
interface = 10.0.1.116:5070 <- replace this with your server ip address
;rtsp = 5074
    
```

**Tableau 1.** Synthèse

	Asterisk	SipXecs	Call Weaver	Freeswitch	Bayonne	YATE
License	GPL 2	GPL 2.1	GPL 2	MPL	GPL	GPL
H323	OUI	NON	OUI	OUI	OUI	OUI
SIP	OUI	OUI	OUI	OUI	OUI	OUI
MGCP	OUI	NON	OUI	NON	NON	NON
IAX	OUI	NON	OUI	OUI	NON	OUI
Interface Web	OUI	OUI	OUI	OUI	OUI	OUI
Configuration	Scripting, CLI, interface WEB, fichier de configuration texte	Scripting, interface WEB, fichier de configuration texte	Scripting, CLI, interface WEB, fichier de configuration texte	Scripting, CLI, Interface Web, fichier de configuration XML	Scripting, CLI, Interface Web	Scripting, CLI, interface WEB, fichier de configuration texte
Codecs	ADPCM,G.711.a (A-Law) et G.711.u (µ-Law), G.722, G.723.1, G.726, G.729, GSM, iLBC, Linear, LPC-10, Speex	G.711, G.722, G.723.1, G.726, G.729, GSM, iLBC, Linear, LPC-10, Speex	G.729, G.723.1	G.722, G.711, G.726, G.723.1, G.729, AMR, iLBC, speex, LPC 10, DVI 14	G.711, G.729	G.729
Version (dernière)	1.4 ou 1.6 BETA	03/10/08	3.10	1.0.0	1.2.11 (Bayonne2)	2.0.0
Plateforme (OS)	Linux,FreeBSD,Mac OS, Windows	Linux, FreeBSD	Linux, FreeBSD, NetBSD, OpenBSD, MacOS X/ Darwin, Open/ Solaris	Linux, Windows, BSD, Mac OS X, Solaris	Solaris, Unix, Linux	Windows, Linux, Unix
origine	Asterisk	Asterisk	Asterisk 1.2 > OpenPBX	Asterisk	N/C	N/C
Dimensionnement (Nombres postes)	2 à 6000	4 à 10 000	2 à 6000	N/C	2 à 120	N/C
Haute disponibilité	OUI	OUI	N/C	N/C	N/C	N/C
Popularité sur Internet (Google)	~ 26 500 000 pages	~142 000 pages	~248 000 pages	~102 000 pages	~172 000 pages	~310 000 pages





**Figure 11.** Solutions VoIP d'Asterisk & Cisco



**Figure 12.** Tribox

Le logiciel *TriBox* (anciennement appelé *Asterisk@home*) est un ensemble de logiciels *open source* qui permet de compléter le logiciel *FreePbx* (ex *Asterisk*) pour fournir l'ensemble des



**Figure 13.** Dell



**Figure 14.** AASTRA

fonctions nécessaires à un central téléphonique haut de gamme. *TriBox* est sous Linux. Il est possible d'utiliser le logiciel *TriBox* sans connaissance de Linux à l'aide de nombreux guides qui existent sur Internet.

## Terminologie

- *TDM* : *Time Division Multiplexing* ou *Multiplexage* à division temporelle est la technique utilisée sur les réseaux de téléphonie traditionnels afin d'allouer des ressources nécessaires à l'acheminement de la voix.
- *Codecs* : Un codec est un algorithme permettant de coder et décoder un signal (la voix), à titre d'exemple celui-ci permet de modifier le format d'un échantillon sonore dans le cadre de la voix sur IP. Le codec est l'élément permettant de définir la qualité de la voix dans un fonctionnement normal (environnement non perturbé) dans le domaine de la téléphonie. Ils permettent la synthèse vocale (numérisation de la voix).
- *Open Source* : Ce terme source ouverte en référence au monde libre, les outils *Open Source* ont la particularité d'être libre d'utilisation à condition de respecter les règles (exemple licences GPL) faisant office de garde fou quand à l'utilisation des codes sources.
- *C/C++* : Langage de programmation pour la conception de logiciel très utilisé par les développeurs d'application.
- *OpenLDAP* : *OpenLDAP* est une solution basée sur le protocole *LDAP* (*Lightweight Directory Access Protocol*) afin de permettre l'interrogation d'un service d'annuaire.
- *Trunk PSTN ou RTC* : Service permettant l'acheminement via un PABX des communications à travers une liaison fournie par l'opérateur.
- *Trunk SIP* : Service permettant l'acheminement via un IPBX des communications à travers une liaison fournie par l'opérateur.
- *STUN* : Mécanismes permettant de s'affranchir des problèmes liés à la translation d'adresse (NAT).
- *NAT* : Mécanisme créé afin de permettre de faire la correspondance entre une adresse IP privée et une adresse IP publique. Ceci permet d'attribuer des adresses privées pour un ensemble de machines d'un réseau domestique ou d'entreprise et un équipement va réaliser la correspondance entre ces adresses privées et l'adresse publique correspondant à une adresse Internet. Ce mécanisme a été créé afin d'améliorer la sécurité, les adresses IP privées ne sont pas accessibles d'Internet ainsi que de permettre d'économiser des adresses IP publiques.
- *E1* : Techniques de transmission multiplexée utilisée en Europe permettant d'acheminer des informations et permettre de transmettre des informations (voix, données) pour un débit de l'ordre de 2.048 Mbit/s (30 canaux + 2 de signalisation).
- *T1* : Techniques de transmission multiplexée utilisée en Amérique du Nord permettant d'acheminer des informations et permettre de transmettre des informations (voix, données) pour un débit de l'ordre de 1.544 Mbit/s (24 canaux + 8 de signalisation).
- *FXO* : *Foreign eXchange Office*, une passerelle permettant d'établir une connexion entre les lignes téléphoniques analogues et un PABX-IP.
- *FXS* : *Foreign eXchange Subscriber*, une passerelle FXS s'utilise pour connecter une ou plusieurs lignes d'autocommutateur traditionnel au réseau téléphonique VoIP ou au fournisseur.
- *RTC* : acronyme définissant Réseaux Téléphonique Commuté.
- *T.38* : T.38 est le protocole de transmission de fax en temps réel normalisé par le groupe de travail 16 de l'UIT.
- *Fork* : Terme utilisé pour définir la création d'un logiciel créé à partir d'un logiciel déjà existant. La particularité est que le noyau du logiciel créé est identique à l'original.

## En aparté sur le système telecom Asterisk

Aujourd'hui Asterisk a fait ces preuves mais il doit continuer à évoluer et à développer des solutions *Grand compte* qui doivent posséder une tolérance de panne, une répartition de charge, des modules correspondant à des solutions métier (Call Center, Taxation...), etc...

De plus, il doit encore conquérir des parts de marché pour entrer véritablement comme une alternative reconnue dans le monde des entreprises et pour cela, plusieurs sociétés commercialisent des solutions de type Asterisk, même les plus grands ci mettent :

- Alliance Cisco – Asterisk,
- Par le biais de leurs solutions *Infoclip* qui utilise un IPBX Asterisk couplé avec les IP phones Cisco.

Appliance destinée aux des PME de moins de 150 personnes à petit prix. Cette Appliance regroupe un logiciel d'origine *Fonality* (*tribox Pro*).

Il a annoncé la mise à disposition d'une gamme de quatre postes *SIP*, d'un téléphone Wifi et d'une extension DECT-ToIP, dont il garantit l'interopérabilité avec Asterisk.

## Société FrameIP, SII

L'expert de votre infrastructure (Réseaux, Sécurité & Téléphonie sur IP). Pôle gestion de projet & marché, pôle expertise technique.  
<http://www.frameip.fr>