



par Luca Bariani
<lucabariani/at/ferrara.linux.it>

Bayonne: le projet opensource pour le téléphone et les services IVR



L'auteur:

Né Ferrara (Italie) en 1977, il obtient un diplôme d'ingénieur en électronique à Ferrara en 2002. Il est membre du groupe d'utilisateur de Linux à Ferrara et du groupe GNU Bayonne. Il développe des services IVR et sms pour les téléphones. Au sein du projet GNU Bayonne, il rédige de la documentation et des articles. Il profite des rencontres opensource pour parler de Bayonne et s'occupe aussi du site internet italien bayonne.it.

Traduit en Français par:
Erwin Kwolek
<kaoul.kae/at/gmail.com>

Résumé:

Bayonne est le serveur IVR et téléphone du projet GNU, de ce fait, c'est un logiciel libre et opensource.

IVR signifie Interactive Voice Response. Un système IVR relie l'informatique aux réseaux téléphoniques (avec le matériel approprié) et agit grâce à des commandes reçues de ces derniers (des signaux de tonalité par pression de touche et parfois, de véritables commandes vocales). Les systèmes IVR permettent la création et la gestion de services téléphoniques complètement automatisés sans opérateur humain.

Systèmes et services IVR

De nos jours, n'importe qui a déjà eu à utiliser un service IVR par téléphone au moins une fois (peut-être même sans s'en être aperçu). En fait ces derniers sont employés par toutes les compagnies de téléphone (mobile ou pas) pour leurs centres d'appel ou pour des services. Réfléchissez, combien de fois avez-vous écouté des voix enregistrées pour choisir un menu dans tel ou tel service, ou pour attendre d'être mis en communication avec un opérateur humain). Les serveurs IVR peuvent agir les uns sur les autres de plusieurs façons et ce, sur des ressources locales ou à distance. Ils peuvent également accéder à des systèmes de fichiers, aux bases de données, aux réseaux et ainsi de suite... Enfin, les systèmes IVR peuvent être facilement intégrés dans les systèmes préexistants ou faire partie de systèmes distribués.

Avec l'IVR beaucoup de services téléphoniques peuvent être créés et utilisés, ci-dessous quelques exemples:

- Service de réponse: un service téléphonique de base, habituellement mis en application avec des systèmes plus simples et meilleur marché que ceux d'IVR. Néanmoins, en utilisant IVR, beaucoup de possibilités et de personnalisations deviennent disponibles:
 - ◆ Contrôler différentes boîtes vocales pour plusieurs utilisateurs;
 - ◆ Authentifier un utilisateur à distance;
 - ◆ Ajouter une notification (courriel ou autre) pour chaque message reçu;
 - ◆ Transférer les messages vocaux à un serveur distant (par Internet et/ou courriel).
- Mises à jour en ligne d'informations dynamiques: Les systèmes IVR peuvent manipuler des données dynamiques qui changent chaque jour, chaque heure. Cela permet la création des services téléphoniques informatifs, par exemple les cours des actions, les dernières nouvelles, la météo, des offres spéciales quotidiennes de magasins...
- Lecture et exploration de catalogues en ligne : Un IVR peut facilement se connecter aux bases de données et à d'autres points d'émission de données (locaux ou à distance), ainsi nous pouvons facilement fournir des services sur le catalogue (par exemple : des livres, des films, CD, des magasins et ainsi de suite). Choissant la catégorie, le coût et de ce dont il a besoin, l'utilisateur peut obtenir des listes et des informations sur tous les articles choisis.
- Achats et réservations : Par l'utilisation des bases de données ou d'autres systèmes dynamiques de données, IVR permet la réservation et les services d'achat de marchandises comme la réservation d'un billet de cinéma ou d'un film dans un magasin de location, l'achat d'un livre ou un DVD. Un catalogue peut être inclus (voir le point précédent), comme le paiement par cartes de crédit ou par d'autres modes de paiement (voir le prochain point).
- Services de paiement: dans chaque pays il y a plusieurs classes de numéros de téléphone avec différents coûts:
 - ◆ « numéros normaux »: le coût des communications téléphoniques dépend du fournisseur, du secteur et du temps d'appel (ce sont les numéros habituels, maison, bureau);
 - ◆ « numéros verts gratuits »: aucun coût pour l'utilisateur (ces numéros sont souvent ceux de centres d'appel de fournisseurs téléphoniques);
 - ◆ « numéros payants »: le coût d'appel est choisi par le fournisseur et habituellement il est plus élevé que qu'un appel normal; la somme peut dépendre du temps de connection ou être fixe.Avec les « numéros payant », l'IVR permet à l'utilisateur le paiement de marchandises et d'informations sans cartes de crédit ou transactions électroniques.
- Notification d'événements: Les systèmes IVR sont bidirectionnels: ils reçoivent et émettent des appels téléphoniques; en utilisant des appels automatiques et programmés, un système IVR peut annoncer au sujet des événements par téléphone (par exemple : rendez-vous, réunions, dates d'expiration, alarmes dans les cas importants...).

IVR fonctionne grâce aux lignes téléphoniques en utilisant le son d'entrée et de sortie à l'utilisateur, permettant une intégration très facile par systèmes vocaux comme TTS (texte à la parole: synthétiseur audio qui convertit le texte écrit en discours oral) et ASR (reconnaissance de la parole asynchrone: analyseur vocal qui convertit

le discours humain en texte). Les premiers permettent à des systèmes IVR de vocaliser oralement du texte écrit et pas seulement des voix pré-enregistrées; les derniers servent à identifier des commandes vocales de voix en entrée utilisateur.

Les systèmes TTS et ASR permettent plus de services téléphoniques IVR:

- Courriel électronique vocal: notre serveur de courriel reçoit un message normal, TTS le convertit en audio et (après authentification appropriée) il peut être écouté par l'utilisateur qui appelle système IVR.
- Service de réponse avancé: ASR peut améliorer le service de réponse décrit ci-dessus. La conversion de la voix reçue par le réseau téléphonique en texte qui peut être facilement envoyé (par l'intermédiaire de courriel ou sur un serveur web, etc.).
- Services téléphoniques pour les personnes sourdes et/ou muettes: nous pouvons utiliser le réseau téléphonique et rendre disponibles des informations aux gens qui habituellement ne peuvent pas y accéder parce qu'ils sont handicapés:
 - ◆ les personnes sourdes-muettes peuvent employer un clavier normal pour écrire un texte converti par un système de TTS et placé sur le réseau téléphonique par IVR;
 - ◆ les personnes sourdes peuvent recevoir un appel téléphonique par IVR, le faire convertir en texte par ASR et être affiché;
 - ◆ Les solutions peuvent être combinées pour les personnes qui sont sourdes-muettes, leur permettant d'émettre et recevoir des appels téléphoniques.

D'une manière générale, IVR offre chaque service et chaque information offerte par un site Web habituel mais sur le réseau téléphonique. La majorité de ces services peuvent donc utiliser les systèmes IVR. Mais IVR permet plus: le téléphone est bidirectionnel, ainsi les systèmes IVR peuvent avoir des services qui appellent des utilisateurs directement (un site web ne peut pas offrir ce dispositif).

Bayonne

Les systèmes IVR ont été pendant longtemps le monopole de solutions fermées et non libres proposées par quelques vendeurs. Chacun d'eux développant des serveurs IVR adaptés à leur propre matériel; Bayonne est la réponse du monde opensource et du projet GNU à cette situation.

Le site officiel de Bayonne est www.gnu.org/software/bayonne où vous pouvez trouver les sources et la documentation (des manuels d'installation et d'administration, un bon nombre d'exemples et des Howtos). Sur bayonne.sourceforge.org, il y a des liens externes vers des ressources et le dernier paquet de développement. Bayonne possède un accès public (en lecture seulement) CVS pour ses sources: la version 1,2 sur savannah.org (la version actuellement stable pour des applications de production) et la version 1,3/2,0 sur sourceforge.org (les versions expérimentales et de développement)

Le but et la principale caractéristique de Bayonne est d'offrir une plateforme IVR virtuelle qui est indépendante du matériel utilisé. Et Bayonne peut fonctionner avec une partie des familles IVR (pas toutes encore) sans pour autant avoir besoin d'une d'entre elles. Pour chaque famille et/ou fournisseur de plateforme IVR, Bayonne comporte un tron commun modulaire qui définit un ensemble d'instructions et un environnement virtuel, pour un usage général et sans aucune référence de matériel. Dans cet environnement virtuel on peut développer des services IVR sans être restreint au matériel utilisé; il est donc possible de changer de plateforme IVR ou la mise en communication d'un service IVR à un autre système avec une plateforme différente ne demande pas de réécrire le code.

Comme indiqué précédemment, Bayonne supporte plusieurs panneaux, même très différents les uns des autres, de 2 à 120 lignes téléphoniques (analogique ou numérique). D'ailleurs, Bayonne peut utiliser encore plus de panneaux de la même famille en même temps (dans la nouvelle version de Bayonne, on peut même

utiliser le panneau de différentes familles en même temps), ainsi des centaines de lignes téléphoniques peuvent être manipulées simultanément. Malheureusement Bayonne ne support pas les modems habituels série/USB (analogique, RNIS ou GSM).

Pour essayer ou tester Bayonne, il ne faut pas de plateforme IVR coûteuse. Actuellement, il suffit d'une carte son car Bayonne utilise un pilote qui émule une plateforme IVR utilisant les entrées/sorties du PC (clavier, microphone et écouteurs) : Bayonne tourne sur un PC de bureau.

Donc pour lancer des services IVR, Bayonne utilise son propre langage, appelé CcScript (un composant externe en version 1.2 intégré dans le paquet principal et pour la version de développement en 1.3/2.0). Ce langage est très simple, avec des actions informatiques complètes et les principales structures de programmation: Blocs conditionnels, boucles, sauts conditionnels ou non, routines.... Une fonction principale de CcScript est d'avoir des instructions « limitées temporellement »: avant qu'une instruction ne soit exécutée, le langage connaît son temps maximum d'exécution. Pour des opérations incluant des appels à des systèmes externes, comme les systèmes de fichiers, les bases de données et les connexions réseau, il y a un paramètre « timeout ». Si l'exécution dure plus longtemps que le timeout, l'opération échoue. Du point de vue du serveur, cette fonction est très importante car elle permet à Bayonne d'offrir des services déterministes et reliés.

Les possibilités informatiques de Bayonne ne sont pas seulement limitées à ceux de CcScript. En fait Bayonne peut interagir avec des langages de haut niveau comme le Perl et le Python. Grâce à cette intégration directe, Bayonne hérite de toutes les bibliothèques et modules disponibles pour ces langages: chaque opération informatique, algorithme habituel ou non, manipulation de données, raccordement à la base de données ou serveur externe (enchaînement, courriel, LDAP, dossier partagés)... Ainsi Bayonne peut traiter tout qu'on peut lui demander.

Même sans Perl et Python, Bayonne peut directement accéder à des bases de données MySQL ou Postgres et aux requêtes SQL sans utiliser CcScript. De la même manière, Bayonne peut accéder au Système de fichiers et se connecter par l'intermédiaire du réseau à des ressources distantes. En outre, Bayonne peut directement interagir avec un synthétiseur de voix TTS comme Flite, thêta (seulement les voix anglaises sont disponibles) et festival (multilingue). Actuellement, Bayonne ne peut se relier aux systèmes de reconnaissance de la parole, mais ce dispositif est au statut de développement.

Pour la plupart des services IVR, Bayonne n'a pas besoin de systèmes externes Texte à Parole car il possède déjà ces propres modules vocaux multilingues (Anglais, Français, Italien et Russe) qui peuvent dire des nombres (ordinaires ou cardinaux), des dates et des temps (ces modules et l'audio sont complètement personnalisables). Ces modules incluent des voies humaines pré-enregistrées et peuvent être utilisés pour la plupart des services téléphones IVR directement dans la architecture de Bayonne.

Bayonne peut gérer des entrées venants de connexions téléphoniques et des entrées provenant du système lui-même dans plusieurs buts : gestion synchronisée, où Bayonne attend pour un signal d'entrée (exemple : attendre qu'une touche soit pressée par l'utilisateur) et de la gestion désynchronisée, où Bayonne reçoit une notification quand un événement arrive (exemple : une touche pressée ou un timeout expiré) sans explicitement les attendre. Ces deux modes de gestion d'entrées permettent à Bayonne de réagir rapidement à l'utilisateur et aux événements système sans aucune perte de temps.

Comme vu précédemment, Bayonne a été développé pour gérer un grand nombre de lignes téléphoniques simultanément, donc il permet des commandes et des instructions « multithreading » (gestion simultanée de plusieurs processus). Chaque processus peut interagir avec les autres par le biais de signaux. Cela autorise des processus de synchronisation et des échanges de données entre les processus et, de plus, en connectant des tâches qui utilisent des lignes différentes (exemple : deux utilisateurs peuvent se parler à travers Bayonne).

Toutes ces caractéristiques rendent Bayonne utile dans un large éventail de situations. Il peut être utilisé pour

de petits services téléphoniques IVR avec quelques lignes, tout aussi bien que pour un grand service central supportant des centaines de lignes. Avec Bayonne on peut facilement avoir des services « simples et rapides » ou des services complexes invoquant par la même des systèmes externes. En conclusion, Bayonne est la bonne solution de serveur IVR pour tous les besoins en services IVR.

<p><u>Site Web maintenu par l'équipe d'édition LinuxFocus</u> © <u>Luca Bariani</u> "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org</p>	<p>Translation information: en --> -- : Luca Bariani <lucabariani/at/ferrara.linux.it> en --> fr: Erwin Kwolek <kaoul.kae/at/gmail.com></p>
---	---

2005-09-06, generated by lfparsr_pdf version 2.51