



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Help

Côté serveur blocs courantes

Côté serveur blocs courantes

Les applications de routage et la voix utilisent les blocs du côté serveur.

- **Dorsal** (voix et routage). Permet d'exécuter personnalisé dorsal Java Server Pages (JSP).
- **Règle métier** (voix et routage). Utilisez cette case pour que les Composer à interroger Genesys Rules Authoring Tool (GRAT). Pour l'ensemble de règles que vous spécifiez Composer interroge GRAT pour les faits associés à l'ensemble de règles. Vous pouvez ensuite définir les valeurs pour les faits, demandez de l'évaluation Genesys Rules Engine et enregistrer les résultats dans une variable.
- **Données de la base de données** (voix et routage). Utilisation de la connexion à une base de données et récupération/manipulation des informations de/dans une base de données. Ce bloc utilise un profil de connexion pour obtenir des informations d'accès à la base de données. Il accepte une requête SQL ou un appel de la procédure, qui peut être défini à l'aide de Query Builder ou l'aide de procédure enregistrée. Il peut aussi utiliser un fichier de script SQL.
- **Entrée de la base de données** (voix uniquement). Accepte un bloc de données de la base de données comme source de données et agit comme un champ de saisie qui accepte une entrée basée sur une grammaire créée à partir de résultats renvoyés depuis la base de données.
- **Service externe** (route uniquement). permet aux applications de routage appeler des méthodes sur des serveurs tiers respecter du protocole Genesys Interaction Server (SIG). Utilisez pour l'échange de données avec des serveurs tiers (non Genesys) qui utilisent le SDK d'Interaction de Genesys ou d'un autre serveur ou application est conforme avec le protocole de communication GIS.
- **Bloc OPM** (voix et routage). Permet aux applications de VXML et SCXML à utiliser les paramètres opérationnels (OPM) qui permettent à un utilisateur d'opérationnelle contrôler le fonctionnement de ces applications en externe. Paramètres opérationnels définis et gérés dans la fonction Operational Parameter Management (OPM) de Genesys Administrator Extension (GAX)
- **Demande de Web** (voix et routage). Permet d'invoquer web reste style Service ou la requête web HTTP pris en charge. Elle prend en charge méthodes mettre, supprimer, obtenir et POST sur HTTPS.
- **Service Web** (voix et routage). Permet d'exécuter des Services Web pour les applications de routage et la voix. Fonction normes de Web Services communes comme XML, SOAP et WSDL au lieu de normes exclusives. Vous pouvez transmettre des paramètres (comme subdialogs) et de stocker les valeurs de retour dans variables. OBtenir, POST et SOAP sur HTTPS sont pris en charge.
- **TLib Block** (route uniquement). Utilisez ce bloc de flux de travail et des sous-colonnes flux de travail qui utilise `<session:fetch>` méthode="tlib". Les propriétés de révèle bloc pour former une demande de TLib pour définir l'état de l'agent n'est pas prêt équivalent à TAgentSetNotReady. Il définit également srcexpr et `<content>` élément pour rendre possible de former générique TLib demandes.

Côté serveur blocs offrent la possibilité d'interagir avec des pages côté serveur internes et externes personnalisées, Web Services et adresses URL. Ces blocs peuvent être utilisés pour échanger des données de variables VoiceXML et SCXML, chaînes JSON entre interprète GVP et personnalisés pages du côté serveur. L'exception de la **bloc de règle métier**, Composer utilise les pages du côté serveur (ASP.NET ou JSP) pour la mise en œuvre les fonctionnalités de bloc côté serveur. Si vous incluez ces blocs dans un diagramme, les pages de côté serveur fournis dans Composer projets sont utilisées en cours d'exécution.

Exemples de scénarios Web

Dans un scénario typique pour le bloc de Service Web ou d'une requête Web, la page du côté serveur Composer fourni est appelée en premier via la plate-forme par le biais de balises de langue approprié (<session : extraction pour SCXML et <données>, <subdialog> pour VXML). Cette page, en fonction des paramètres d'entrée spécifiées dans le bloc, ouvre n'importe quel URL externe pour les blocs de Service Web ou d'une requête Web. En cas d'arrêt du Service Web, il est à la demande SOAP appropriée et envoie des. Ensuite, il analyse la réponse, il reçoit de la demande externe et rend disponible à l'application. La figure ci-dessous montre le flux. [Image:ServerSide.gif](#)

Le besoin de Pages du côté serveur

Composer fournit les blocs de côté serveur en que les utilisateurs sont généralement carte leur callflows ou le flux de travail à leur la logique métier, via ces blocs de prévision. Par exemple, le bloc dorsal offre la possibilité de créer personnalisé dorsal pages serveur pouvant être plus étroitement associé avec la logique métier et en même temps offre davantage de flexibilité, car la logique dorsal est fournie par l'utilisateur. Les différentes fonctions du côté serveur offrent un service proxy qui peut être utilisé pour la requête Web Services, les serveurs de web et les pages serveur dorsal lors de la qui fournit une interface utilisateur qui est suffisamment simple à utiliser, mais également elle contient des fonctionnalités avancées. Concernant la sécurité, les blocs de la requête Web et de Service Web offrent des clients proxy qui prend en charge HTTP et HTTPS, ainsi que les SOAP/HTTPS. Composer prend en charge les pages du côté serveur dans Java et .NET.

- Pages de Java server sont hébergés sur Apache Tomcat, qui est sous et déployé avec Composer.
- Applications .NET sont hébergées sur Microsoft IIS. Ces derniers doivent être déployés par l'utilisateur sur le même serveur comme Composer.

Le choix entre l'utilisation de Java ou .NET dépend principalement les techniques sont disponibles pour l'utilisateur, ainsi que les plates-formes. Ci-dessous est une matrice de décision décrivant les certains situations courantes où le bloc plus approprié du côté serveur est recommandé.

Situation	Recommandé bloc	Commentaires
Un flux de callflow doit utiliser un Service Web, qui comporte une définition de WSDL.	Bloc de Service Web	Le bloc de Service Web fournit utilities pour concevoir la manière dont le Service Web sera utilisé, comme un analyseur WSDL. Pendant l'exécution, les résultats de la sortie également facilement peuvent être attribués qu'à des variables callflow ou de flux de travail.
Un flux de callflow doit interroger un serveur web pour les données	Bloc de requête Web	Le bloc de la requête Web fournit un client proxy pour l'envoi de la requête web, tout en offrant la fonctionnalité de HTTPS, attribution le résultat pour les variables, etc..
Un service web de repos style doit être utilisés par l'application.	Bloc de requête Web	

<p>Un flux de callflow doit accéder à certaines données à l'aide de certains interface spécifique à l'aide de ne pas HTTP ou SOAP</p>	<p>Bloc dorsal</p>	<p>Le bloc dorsal propose un service de proxy pour une application principale est développé par l'utilisateur et personnalisée en conséquence.</p> <p>Le bloc dorsal permet de vous réutiliser personnalisé bocaux et ensembles de .NET rapidement car il fournit un mécanisme simple pour transmettre des paramètres dans la page du serveur dorsal les. Les pages de dorsal fournissent un squelette mise en œuvre, ce qui facilite et rapide pour démarrer l'application logique personnalisée qui permet d'autres bibliothèques fourni par l'utilisateur.</p>
<p>Un flux de callflow doit effectuer un traitement après personnalisé pour les données récupérées</p>	<p>Bloc dorsal</p>	<p>L'application dorsal devront être créé de sorte qu'il récupère les données et après le traite en conséquence.</p>
<p>Mon application ne fonctionne pas avec le Service Web ou les blocs de la requête Web. Que puis-je utiliser ?</p>	<p>Bloc dorsal</p>	<p>Réessayez commençant par le bloc dorsal, étant donné que l'application est ouverte par la nature. L'application dorsal est conçue pour fournir une interface simple à l'application de spécifiques à l'utilisateur en cours.</p> <p>Remarque : La page du côté serveur dorsal appelée depuis le bloc dorsal faire partie de la projet et seront incluse lorsque le projet d'application est déployé.</p>