



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.

Genesys Rules Authoring Tool 8.5.x Help

Importation et exportation de modèles

Importation et exportation de modèles

Importer un projet de modèle

Utilisez ce panneau pour importer des modèles précédemment exportés —, par exemple des exemples de modèles ou des modèles exportés d'un autre système.

1. Sélectionnez le nœud **Modèles** dans l'arborescence de navigation de gauche.
2. Cliquez sur le bouton **Importer les modèles**.
3. Cliquez sur le bouton **Choisir un fichier**, parcourez le système de fichiers et le modèle de projet à importer.
4. Si vous cliquez sur **Remplacer les modèles existants dans le référentiel**, le référentiel de toute version antérieure de chaque modèle est supprimé avant l'importation de nouveaux modèles dans le système cible. Si cette option n'est pas activée et qu'un modèle existant portant le même nom est trouvé dans le référentiel, un message d'erreur s'affiche et il est mis fin à l'importation.
5. Cliquez sur **Terminer**.

Avertissement

Soyez prudent lors de la modification de modèles ou de versions de modèle, car cette opération risque d'affecter les règles existantes. Ainsi, il se peut qu'une règle existante utilise une condition qui n'existe pas dans une autre version du modèle. Contactez le développeur du modèle de règle pour être sûr de choisir les versions et modèles corrects pour votre application.

Exporter un projet de modèle

1. Sélectionnez le nœud **Modèles** dans la navigation de gauche.
2. Sélectionnez les projets de modèles que vous souhaitez exporter.
3. Nommez le fichier d'exportation et l'emplacement d'exportation du projet de modèle.
4. Cliquez sur **Terminer** pour terminer l'exportation.

Important

Il appartient au développeur de modèle de règle de veiller à ce que les modèles qu'il développe ne rencontrent pas de problèmes de conflit de noms. Par exemple, les

noms de fonction, les signatures de méthode Java et les faits doivent porter des noms différents, même s'ils se trouvent dans des modèles différents, car un créateur de règle peut créer des règles basées sur plusieurs modèles. Les noms ne doivent pas être dupliqués et le créateur de règles doit savoir quels modèles/versions doivent être utilisés, et dans quelle combinaison.