



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 Help

Genesys Rules System 9.0.0

Table of Contents

Ajuda do Genesys Rules Authoring Tool	4
Visão geral das regras	7
Criar regras lineares	11
Atualizar regras lineares	13
Copiar regras lineares	14
Importar regras lineares	15
Exportar regras lineares	16
Excluir regras lineares	17
Criar tabelas de decisão	18
Atualizar tabelas de decisão	20
Copiar tabelas de decisão	21
Importar tabelas de decisão	22
Exportar tabelas de decisão	23
Excluir tabelas de decisão	24
Visão geral do pacote de regras	25
Verificar minhas permissões	26
Criar pacotes de regras	27
Editar pacotes de regras	29
Excluir pacotes de regras	30
Implantar/cancelar implantação de pacotes de regras	31
Trilha de auditoria	37
Exibir o histórico de pacotes	38
Importar modelos de regras	39
Importar pacote de regras	41
Exportar modelos de regras	43
Exportar pacote de regras	44
Visão geral de calendários comerciais	46
Criar calendários comerciais	48
Copiar calendário comercial	50
Excluir calendários comerciais	51
Visão geral dos testes de regra	52
Usar Teste de divisão para comparar resultados de regras de negócios	53
Visão geral do desenvolvimento do modelo de regra	54
Importar e exportar modelos	61
Criar e editar modelos	63

Publicar modelos	65
Exemplos de desenvolvimento de modelos	66
Mapeamento da linguagem da regra	80
Usar o Drools 5	82
Editor de ações	84
Editor de condições	86
Editor de enumerações	87
Editor de modelos de fatos	88
Editor de funções	89
Editor de parâmetros	90
Pesquisar	100

Ajuda do Genesys Rules Authoring Tool

O Genesys Rules Authoring Tool (GRAT) é um aplicativo com base no navegador que permite criar e editar regras de negócios e, na versão 9.0, desenvolver os modelos nos quais as regras se fundamentam. (Eles eram desenvolvidos anteriormente no Genesys Rules Development Tool, um componente separado baseado em Eclipse.)

Os desenvolvedores de modelos de regras de negócios usam essa ferramenta para criar os modelos subjacentes a partir dos quais regras e pacotes de regras são criados. Os modelos de regras também podem ser desenvolvidos externamente e importados diretamente para o GRAT.

Os autores de regras de negócios usam essa ferramenta para criar, editar ou excluir regras e implantá-las nos sistemas de produção ou de laboratório.

Todos os recursos descritos aqui estão disponíveis apenas a usuários que têm as permissões pertinentes configuradas usando o controle de acesso baseado em função (RBAC). Consulte os administradores do seu sistema se não tiver as permissões necessárias.

Links rápidos

Visão geral, descrições e resumos

[Regras \(lineares e tabelas de decisão\)](#)

[Pacotes de regras](#)

[Calendários comerciais](#)

[Desenvolvimento de modelo de regra](#)

[Testes de regra](#)

Trabalhar com regras lineares

[Criar regras lineares](#)

[Atualizar regras lineares](#)

[Copiar regras lineares](#)

[Importar regras lineares](#)

[Exportar regras lineares](#)

Trabalhar tabelas de decisão

[Criar tabelas de decisão](#)

Trabalhar com pacotes de regras 1

[Criar pacotes de regras](#)

Atualizar tabelas de decisão

Copiar tabelas de decisão

Editar pacotes de regras

Importar pacote de regras

Trabalhar com pacotes de regras 2

Importar modelos de regras

Exportar modelos de regras

Verificar minhas permissões

Exibir o histórico de pacotes

Trilha de auditoria

Trabalhar com calendários comerciais

Criar calendários comerciais

Copiar calendários comerciais

Excluir calendários comerciais

Trabalhar com modelos

Importar e exportar modelos

Criar e editar modelos

Publicar modelos

Exemplos de desenvolvimento de modelos

Trabalhar com editores de modelo

Editor de ações

Editor de condições

Editor de enumerações

Editor de modelos de fatos

Editor de funções

Editor de parâmetros

Testes de regra

Usar teste de divisão para comparar resultados de negócios

Novo no GRAT versão 9.0

A funcionalidade do Genesys Rules Authoring Tool e do antigo Genesys Rules Development Tool foi combinada em um único componente — GRAT. Ainda é possível usar a versão 8.1 do GRDT para desenvolver modelos e importá-los para o novo componente GRAT, mas o GRDT versão 8.1 não é mais enviado com o novo GRAT.

Os elementos de marca da interface do usuário são alinhados com outros software da Genesys.

Uma nova opção de configuração — **Desabilitar**



serialização do pacote — aparece na guia **Geral** de um pacote de regras. Quando esta opção está habilitada (o valor é definido como `false` [padrão]), os pacotes de regras de baixo uso são serializados e podem ser recarregados rapidamente sob demanda.

- Os botões desfazer/refazer foram adicionados à interface do usuário para ajudar na edição de regras, calendários e cenários de teste.
- Os recursos de controle de acesso baseado em função (RBAC) são estendidos à funcionalidade de desenvolvimento de regras no GRAT que estava anteriormente no GRDT.
- Limitação conhecida — Algumas funcionalidades relacionadas ao processamento de eventos complexos e mapeamento de parâmetros não estão implementadas no lançamento inicial da versão 9.0. O resultado disso é que não há nenhuma provisão para criar ou editar modelos que suportem cenários de teste de regra e teste de divisão. No entanto, você ainda pode importar os modelos criados na versão 8.1 do GRDT e basear regras e pacotes neles, mas não pode editar ou criar os modelos na versão de lançamento inicial.

Visão geral das regras

Uma regra de negócios é uma parte da lógica que define, em uma pequena escala, o que uma empresa faz. Para o Genesys Rules System, uma regra é uma parte externa da lógica que pode ser personalizada por analistas comerciais e invocada por aplicativos. Isso permite ajustar os comportamentos comerciais específicos, conforme necessário.

Tipos de regras

O GRAT permite configurar dois tipos de regras: linear e tabelas de decisão.

Regras Lineares

As regras lineares seguem o seguinte formato básico:

```
WHEN {condition} THEN {action}
```

Quando a condição for verdadeira, a ação ocorrerá. Este formato de regra é melhor para as ações simples, como atribuição de um valor para voltar para o aplicativo. No entanto, observe que regras lineares podem ter várias condições e ações ou apenas ações sem condições. As condições e as ações que estão disponíveis dependem dos modelos de regras incluídos no pacote de regras.

Exemplo

Propósito

Se a idade de um cliente estiver entre 30–40 anos, a interação do cliente será encaminhada para o Grupo de agentes 1.

É possível criar uma nova regra linear no Genesys Rules Authoring Tool. Insira o nome, fase e assim por diante, conforme desejado e, em seguida, adicione uma condição e uma ação. As fases das quais o autor de regras pode selecionar são determinadas pelo modelo de regra que o autor de regras está usando.

Há uma enumeração chamada Fases no fato `_GRS_Environment` que será criada sempre que um novo projeto de modelo de regras for criado na guia Desenvolvimento de regras. Se a enumeração de fases não estiver presente, o autor de regras verá simplesmente * no menu suspenso **Fase**. Nesse caso, a fase não será considerada ao avaliar o pacote de regras.

Importante

O fato `_GRS_Environment` deve ser fornecido para todas as avaliações de regras. Um fato `_GRS_Environment` vazio deve ser fornecido para regras ao nível do pacote que não usam uma Fase (ou seja, a Fase foi definida como *).

As listas suspensas Adicionar Condição e Adicionar Ação são preenchidas com todas as condições e ações que foram criadas nos modelos de regras incluídos no pacote de regras. As listas suspensas contêm as expressões de linguagem que os desenvolvedores de regras usaram durante a criação dos componentes e não o mapeamento da linguagem das regras. Isso possibilita a criação de regras sem conhecer o mapeamento da linguagem da regra ou familiarizar-se com o Drools.

Os parâmetros contidos em cada condição e ação são representados pelos nomes inseridos para eles. O autor de regras de negócios deve substituir esse nome inserindo um valor (como para uma faixa etária) ou selecionando uma opção na lista suspensa (como para um Grupo de agentes).

Portanto, para criar essa regra, o autor de regras selecionará Age Range como a condição e irá inserir 30 como o parâmetro {ageLow} e 40 como o parâmetro {ageHigh}. A ação será Target Agent Group e Agent Group 1 será selecionado na lista suspensa {agentGroup}.

Mais informações

Os tópicos a seguir explicam como trabalhar com regras lineares no GRAT:

- **Criar regras lineares**
- **Atualizar regras lineares**
- **Copiar regras lineares**
- **Importar regras lineares**
- **Exportar regras lineares**
- **Excluir regras lineares**

Tabelas de decisão

As tabelas de decisão permitem que você junte várias Regras lineares com o mesmo conjunto de condições (quando) e ações (então) para serem usadas para um caso de negócios (estruturado) complexo. Use as tabelas de decisão para evitar dezenas de regras lineares com estrutura idêntica no sistema.

Importante

1. As opções nas tabelas de decisão precisam ser mutuamente exclusivas para evitar ambiguidade. Isso garante que haja apenas um resultado por avaliação. Se as opções não forem mutuamente exclusivas, várias linhas poderão ser executadas em nenhuma ordem específica. A última linha executada determinará o resultado final.
2. O número máximo de colunas suportadas nas Tabelas de decisão é 50.
3. Quando você estiver editando regras, tenha cuidado para não excluir os seus dados de cookies, pois isso pode fazer com que a regra seja perdida. Consulte a documentação do navegador que você está usando para obter mais informações sobre como impedir que um usuário limpe os dados de cookies.

Mais informações

Os tópicos a seguir explicam como trabalhar com tabelas de decisões no GRAT:

- **Criar tabelas de decisão**
- **Atualizar tabelas de decisão**
- **Copiar tabelas de decisão**
- **Importar tabelas de decisão**
- **Exportar tabelas de decisão**
- **Excluir tabelas de decisão**

Ordem de execução

É possível configurar regras para vários contextos comerciais (nós representando os diversos elementos na sua hierarquia de estrutura comercial) ou, para regras globais, ao nível de pacote de regras. No painel de navegação, cada contexto comercial na estrutura comercial configurada é representado ao nível de um nó diferente. A ordem de execução das regras em um pacote de regras depende do nível do nó: as regras são executadas primeiro ao nível do pacote/global, depois em cada nível da hierarquia por vez.

Assim, se você tiver definido essa hierarquia:

- Pacote
 - Departamento de vendas
 - Financeiro

e durante a execução, você especifica 'Departamento de vendas'/'Finanças', então a ordem de execução será:

1. Regras ao nível do pacote (de acordo com a prioridade).
2. Regras para o departamento de vendas (de acordo com a prioridade).
3. Regras para o financeiro (de acordo com a prioridade).

Em um determinado nó, é possível modificar a ordem de execução, usando as setas para cima  ou para baixo  em cada regra.

Apenas as regras em um caminho de nó específico são executadas em qualquer execução de determinadas regras. O caminho de execução é determinado pela entrada no Mecanismo de regras na solicitação de execução.

Importante

A estrutura comercial é definida no Gerenciador de configurações ou no Genesys Administrator.

Importante

Os administradores de sistema podem configurar a execução de regras para ser 'de cima para baixo' ou 'de baixo para cima'. O indicador **Ordem de avaliação das regras** na parte inferior da tela mostra qual dessas está selecionada e uma dica de ferramenta está disponível ao focar o cursor do mouse sobre esse indicador. Qualquer alteração feita para essa configuração será aplicada dinamicamente, mas terá efeito somente após uma reinicialização ou atualização do navegador.

Trilha de auditoria

A guia **Trilha de auditoria** permite exibir o histórico de regras individuais, como quando elas foram atualizadas ou implantadas e por quem. Quando acessada em um contexto comercial (um nó na Árvore do Explorer), a guia **Trilha de auditoria** listará as regras que existem para esse contexto comercial.

Criar regras lineares

Siga essas etapas para criar uma regra linear:

1. Navegue até o pacote de regras ao qual a nova regra pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Navegue até o nó correto da estrutura comercial sob o pacote de regras que definirá o nó no qual a sua regra linear será criada. Se você criar a regra linear ao nível de pacote de regras, será uma regra global. Selecione o nó na navegação à esquerda e clique na guia **Regras**.
2. Clique em **Nova regra linear**.
3. Em **Resumo de regras**, o campo **ID** é preenchido automaticamente. Este campo não pode ser editado.
4. Insira um **Nome** para a regra (por exemplo, Gold).
5. Insira uma breve **Descrição** para a regra (por exemplo, se o cliente é um membro Ouro, aumente a prioridade).
6. Selecione a **Fase** em que essa regra será aplicada (classificação, priorização ou arquivamento para iWD. Consulte o Guia de implantação do Genesys Rules System para obter mais informações sobre as fases).
7. Selecione o **Calendário comercial** a ser usado com essa regra (opcional).
8. O campo **Instantâneo pendente** é exibido com um símbolo de seleção, indicando que o conteúdo dessa regra ainda não foi incluído em um instantâneo do pacote. Consulte Implantação para obter detalhes sobre como trabalhar com instantâneos.
9. Insira uma **Data de início** e uma **Data de término** para a regra (opcional). Se a **Data de término** for anterior à data atual, um aviso indicará que a regra está desatualizada.
10. No painel inferior, preencha as linhas **Quando** e **Então**.
 - a. Para adicionar uma Condição (Quando), clique em **Adicionar condição** e selecione da lista (por exemplo, uma condição para este cenário pode ser Quando o cliente for um membro Ouro). A condição da regra inclui o nome do modelo da regra a partir do qual a condição é derivada.
 - b. Para adicionar uma Ação (Então), clique em **Adicionar ação** e selecione na lista (por exemplo, uma ação para esse cenário pode ser Aumentar a prioridade em 100). A ação da regra inclui o nome do modelo da regra a partir do qual a ação é derivada.

Importante

O número máximo de segmentos (texto mais variáveis) em Condições ou Ações em regras lineares é 9. Uma mensagem de erro será exibida se esse limite for ultrapassado.

- c. Insira valores para os parâmetros na tabela sob as colunas **Condição** e **Ação**. Dependendo de como os parâmetros foram configurados pelo desenvolvedor de modelo de regra, pode haver restrições quanto aos valores possíveis de serem inseridos.
4. Clique em **Validar** para validar a sintaxe da regra linear. A opção **Validar** aparece no menu **Mais**.

-
5. Clique em **Salvar** para salvar suas alterações.

Ao editar regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Atualizar regras lineares

Siga estas etapas para atualizar uma regra linear existente:

1. Navegue até o pacote de regras ao qual a regra linear pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto em (no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a regra linear na lista e atualize as informações, conforme necessário. Consulte **Criar regras lineares** para obter detalhes sobre os campos que podem ser atualizados. Quaisquer alterações feitas no **Resumo de regras** são salvas automaticamente. Clique em **Salvar** para salvar quaisquer alterações feitas no corpo da regra. Forneça um comentário de check-in que resuma as alterações feitas. Isso será exibido no **Histórico de auditoria**.

Quando estiver editando regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Copiar regras lineares

É possível copiar uma regra linear e colar essa cópia no mesmo pacote de regras, seja no mesmo nó ou em um nó diferente. Siga estas etapas para copiar uma regra linear:

1. Navegue até o pacote de regras ao qual a regra linear pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto na árvore (o nível no qual a regra foi definida) e clique na guia **Regras**.
2. Localize a regra na lista, selecione-a e, em seguida, selecione **Copiar regra** a partir do menu **Mais**.
3. Se desejar que a cópia esteja no mesmo nó, clique em **Colar regra**.
4. Se desejar que a cópia esteja em um nó diferente, selecione esse nó na árvore, clique na guia **Regras** e, em seguida, clique em **Colar regra**.

Importante

Se desejar mover a regra para outro local, primeiro copie, depois cole e, em seguida, volte e exclua o original. O sistema não permitirá que você cole uma regra após esta ter sido excluída do repositório.

5. Atualize as informações conforme necessário e clique em **Salvar**. Consulte **Criar regras lineares** para obter detalhes sobre os campos que podem ser atualizados.

Ao editar regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Importar regras lineares

É possível importar regras lineares no GRAT.

Importante

É possível importar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende. Consulte **Importar pacote de regras** para obter detalhes.

Siga estas etapas para importar uma regra linear:

1. Navegue até o pacote de regras ao qual a regra linear pertencerá na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra será definida) e clique na guia **Regras**.
2. Clique em **Importar regra**.
3. Navegue até o local do arquivo da regra linear.
4. Insira um comentário.
5. Clique em **Importar**.
6. A regra linear será exibida na lista de regras. Edite os campos conforme necessário. Consulte **Criar regras lineares** para obter detalhes sobre os campos que podem ser atualizados. Clique em **Salvar**.

Ao editar regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Exportar regras lineares

É possível exportar regras lineares exportando o pacote de regras que contém a regra. Talvez você queira exportar as suas regras para fazer backup delas ou movê-las para outro servidor e, em seguida, importá-las novamente.

Você pode exportar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende. Consulte **Exportar pacote de regras** para obter detalhes.

Para exportar uma regra linear:

1. Navegue até o pacote de regras ao qual a regra linear pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra foi definida) e clique na guia **Geral**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a regra na lista, selecione-a e, em seguida, selecione **Exportar pacote de regras** na lista suspensa no painel inferior. Há A opção exportada pode ser salva somente como um arquivo .xml.

Excluir regras lineares

Siga estas etapas para excluir uma regra linear:

1. Navegue até o pacote de regras ao qual a regra linear pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (o nível no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a regra linear na lista e clique no ícone de menos na coluna **Ações** à direita.

Excluir regras criadas por engano

Se você acidentalmente criar uma regra e desejar excluí-la, adicione os dois valores obrigatórios (**Nome** e **Fase**) antes de tentar excluí-la. Não é possível excluir uma regra, a menos que os valores obrigatórios sejam fornecidos.

Criar tabelas de decisão

Siga estas etapas para criar uma nova tabela de decisão:

1. Navegue até o pacote de regras ao qual a nova tabela de decisão pertencerá na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Navegue até o nó correto da estrutura comercial sob o pacote de regras, que definirá o nó no qual a sua tabela de decisão será criada. Se você criar a tabela de decisão ao nível de pacote de regras, será uma regra global. Selecione o nó e clique na guia **Regras**.
2. Clique em **Nova tabela de decisão**.
3. Em **Resumo de regras**, o campo **ID** é preenchido automaticamente. Este campo não pode ser editado.
4. Insira um **Nome** para a tabela de decisão (por exemplo, Status).
5. Insira uma breve **Descrição** para a regra (por exemplo, Ajustar a prioridade, dependendo do status do cliente).
6. Selecione a **Fase** em que essa regra será aplicada (classificação, priorização ou arquivamento para iWD. Consulte o Guia de implantação do Genesys Rules System para obter mais informações sobre as fases).
7. Selecione o **Calendário comercial** a ser usado com essa regra (opcional).
8. Insira uma **Data de início** e uma **Data de término** para a regra (opcional). Se a **Data de término** for anterior à data atual, um aviso é exibido para indicar que a regra está desatualizada.
9. Use as setas para cima e para baixo na última coluna à direita para controlar a ordem das linhas da tabela de decisão. Em alguns casos complexos, as regras podem ser projetadas para que várias linhas sejam avaliadas como verdadeiras. Nesse caso, a ordem das linhas torna-se importante, por isso é possível reordenar as linhas ao criar e editar uma tabela de decisão.

Importante

Os administradores de sistema podem configurar a execução de regras para ser 'de cima para baixo' ou 'de baixo para cima'. O indicador **Ordem de avaliação das regras** na parte inferior da tela mostra qual dessas está selecionada e uma dica de ferramenta está disponível ao focar o cursor do mouse sobre esse indicador. Qualquer alteração feita para essa configuração será aplicada dinamicamente, mas terá efeito somente após uma reinicialização ou atualização do navegador.

10. Adicione **Condições** e **Ações** no painel inferior.

Importante

É possível usar um símbolo de caractere curinga (*) em dados de linha em uma tabela de decisão (se o recurso for configurado por administradores). O caractere curinga indica que, para essa linha, o valor do parâmetro onde ele é usado não é importante e não será avaliado. Uma seleção de caractere curinga agora é exibida no topo de todas as listas, independentemente de serem enumerações, atributos comerciais, Configuration Server, banco de dados etc. No caso de parâmetros numéricos, é necessário digitar o valor do caractere curinga — o GRAT agora aceita isso como um campo de número válido. Para qualquer condição que contenha um ou mais caracteres curingas, sua avaliação não será considerada na lógica da regra. Há algumas restrições:

- Os valores de caracteres curinga funcionarão apenas para cadeias de caracteres e campos numéricos — campos de data, hora e booleanos não são aceitos.
- Os valores de caracteres curinga são 'tudo ou nada' para condições com vários parâmetros. Por exemplo:

Customer age is between 40 and 60

é UMA condição e será excluída desta linha se um ou mais campos contiverem um valor de caractere curinga.

- a. Selecione uma ou mais **Condições** da lista (por exemplo, uma condição para este cenário pode ser nomeada A idade do cliente é...).
 - b. Selecione uma ou mais **Ações** da lista (por exemplo, uma ação para este cenário pode ser nomeada Aumentar prioridade em xxx).
 - c. Insira valores para os parâmetros na tabela sob as colunas **Condição** e **Ação**. Dependendo de como os parâmetros foram configurados pelo desenvolvedor de modelo de regra, pode haver restrições quanto aos valores possíveis de serem inseridos.
 - d. Repita a Etapa C, adicionando mais valores de ação e condição.
 - e. Reordene as linhas conforme o apropriado.
6. Clique em **Validar** para validar a sintaxe da regra linear.
 7. Clique em **Salvar** para salvar suas alterações.

Importante

Ao editar regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Importante

O campo **Instantâneo pendente** indica se algum instantâneo dessa regra foi criado. Consulte Implantar pacotes de regras para obter informações sobre instantâneos.

Atualizar tabelas de decisão

Ao editar ou atualizar uma regra da tabela de decisão, você poderá inserir ou remover as colunas de ações e condições apenas depois que todos os dados na tabela forem validados (nenhuma linha vermelha visível). Essa restrição o impede de perder dados não salvos que acabaram de ser inseridos. Se você inserir acidentalmente a condição ou ação errada, ou se acidentalmente excluir uma coluna de condição ou ação, poderá clicar em **Cancelar** para reverter a regra novamente para a última versão salva.

Importante

Clicar em **Cancelar** resultará na perda de dados que foram inseridos na tabela, mas não foram salvos.

Ao adicionar linhas a uma regra da tabela de decisões, é importante corrigir todos os erros de validação nessa linha antes de tentar adicionar ou excluir colunas de condição ou de ação, para evitar perda de dados na linha em que você está trabalhando.

Para atualizar uma tabela de decisão existente:

1. Navegue até o pacote de regras ao qual a tabela de decisão pertence na Árvore do Explorer (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto na Árvore do Explorer (o nível no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a tabela de decisão na lista e atualize as informações, conforme necessário. Consulte **Criar tabelas de decisão** para obter detalhes sobre os campos que podem ser atualizados. Quaisquer alterações feitas no **Resumo de regras** são salvas automaticamente. Clique em **Salvar** para salvar quaisquer alterações feitas no corpo da regra. Forneça um comentário de check-in que resuma as alterações feitas. Isso será exibido no **Histórico de auditoria**.

Quando estiver editando regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Copiar tabelas de decisão

É possível copiar uma tabela de decisão e colar essa cópia no mesmo pacote de regras ou em um diferente. Siga essas etapas para copiar uma tabela de decisão:

1. Navegue até o pacote de regras ao qual a tabela de decisão pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a tabela de decisão na lista, selecione-a e, em seguida, selecione **Copiar regra** a partir do menu **Mais**.
3. Se desejar que a cópia esteja no mesmo nó, clique em **Colar regra**.
4. Se desejar que a cópia esteja em um nó diferente, selecione esse nó na árvore, clique na guia **Regras** e, em seguida, clique em **Colar regra**.

Importante

Se desejar mover a regra para outro local, primeiro copie, depois cole e, em seguida, volte e exclua o original. O sistema não permitirá que você cole uma regra após esta ter sido excluída do repositório.

5. Atualize as informações conforme necessário e clique em **Salvar**. Consulte **Criar tabelas de decisão** para obter detalhes sobre os campos que podem ser atualizados.

Ao editar regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

Importar tabelas de decisão

É possível importar tabelas de decisões no GRAT.

Importante

É possível importar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende. Consulte **Importar pacote de regras** para obter detalhes.

Siga estas etapas para importar uma tabela de decisão:

1. Navegue até o pacote de regras ao qual a tabela de decisão pertencerá na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra será definida) e clique na guia **Regras**.
2. Clique em **Importar regra**.
3. Navegue até o local do arquivo da tabela de decisão. Os arquivos da tabela de decisão podem estar no formato .xml ou .xls.
4. Insira um comentário.
5. Clique em **Importar**.
6. A tabela de decisão será exibida na lista de regras. Edite os campos, conforme necessário. Consulte **Criar tabelas de decisão** para obter detalhes sobre os campos que podem ser atualizados. Clique em **Salvar**.

Quando estiver editando regras, tenha cuidado para não limpar o seu histórico de navegação ou dados de cookies uma vez que isso pode fazer com que alterações não salvas sejam perdidas.

É possível importar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende. Consulte **Importar pacote de regras** para obter detalhes.

Exportar tabelas de decisão

É possível exportar tabelas de decisões no GRAT. Por exemplo, talvez você queira exportar as suas regras para fazer backup delas ou movê-las para outro servidor e, em seguida, importá-las novamente. Também é possível exportar o arquivo em formato de planilha (.xls). Isso poderá ser útil se você desejar uma tabela de decisões com centenas de linhas poderá ser mais fácil trabalhar no Excel, replicando linhas e fazendo alterações secundárias e, em seguida, importando-as novamente no Rules Authoring Tool, em vez de criar centenas de linhas na ferramenta.

Importante

É possível exportar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende. Consulte **Exportar pacote de regras** para obter detalhes.

Para exportar uma tabela de decisão:

1. Navegue até o pacote de regras ao qual a tabela de decisão pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a tabela de decisão na lista e clique em **Exportar regra**. Há duas opções para exportar tabelas de decisão: .xml ou .xls. Selecione o formato preferido.
3. Você pode **Abrir** ou **Salvar** o arquivo de regra exportado.

Excluir tabelas de decisão

Siga estas etapas para excluir uma tabela de decisão:

1. Navegue até o pacote de regras ao qual a tabela de decisão pertence na navegação à esquerda (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione o nó correto (no qual a regra foi definida) e clique na guia **Regras**.

Dica

Também é possível usar o recurso **Pesquisar** para localizar regras.

2. Localize a tabela de decisões na lista e clique no ícone de menos na coluna **Ações** à direita.

Visão geral do pacote de regras

Pacotes de regras são conjuntos de regras. Os pacotes de regras são usados para agrupar, gerenciar e implantar regras. As regras de um pacote de regras fornecem um conjunto de funcionalidades (como uma solução iWD). O Genesys Rules Authoring Tool (GRAT) permite criar, editar e excluir pacotes de regras.

Os pacotes de regras fornecem os seguintes recursos:

- A capacidade de particionar regras e fatos de forma que eles fiquem menores, bem definidos e sejam aplicados apenas a um aplicativo ou uso específico. Isso facilita a sua depuração e seu entendimento. O modelo de fatos é uma descrição dos dados. Ele contém tipos e nomes de campos que são agrupados em tabelas/classes. Os fatos são entrada/saída para a execução da regra e são instâncias de tabelas/classes definidas no modelo de fatos.
- A capacidade de isolar os pacotes de regras a partir de um outro ao executar regras. Isso também melhora o desempenho pois o Mecanismo de regras tem menos candidatos a serem examinados durante a avaliação.
- A capacidade de atualizar pacotes de regras individuais sem afetar outros pacotes implantados.
- A capacidade de importar e exportar um pacote de regras inteiro contendo as definições de regras, os calendários comerciais e também os modelos dos quais o pacote de regras depende.
- Um pacote de regras contém uma ou mais regras mais o modelo de fatos necessário para suportar as regras. Os pacotes de regras são implantados individualmente no Mecanismo de regras.

Ao selecionar um pacote de regras existente na navegação à esquerda, quatro guias são exibidas:

- A guia **Geral** exibe as informações básicas para o pacote de regras, como nome, tipo e modelos associados.
- A guia **Regras** permite criar, editar e visualizar regras. Quando você clica no nó do pacote de regras e depois na guia Regras, você pode criar, editar e exibir regras no nível global para esse pacote. Clicar em outros nós (que representam vários contextos comerciais) permite modificar as regras definidas para o contexto comercial específico.
- A guia **Trilha de auditoria** permite exibir o histórico de regras individuais, como quando elas foram atualizadas ou implantadas e por quem.
- A guia **Histórico de pacotes** permite visualizar o histórico de um pacote e suas versões e instantâneos, incluindo alterações a regras, modelos, calendários, cenários de teste, importações/exportações e implantações. O histórico de todos os pacotes de um inquilino também pode ser exibido ao nível de Locatário.

Importante

Quando um pacote é excluído, todas as referências a ele também são excluídas do Histórico de pacotes.

Verificar minhas permissões

Para verificar todas as permissões que lhe foram concedidas sob o controle de acesso baseado em regras:

1. Na navegação à esquerda, selecione a guia **Locatário**.
2. Clique no botão **Verificar minhas permissões**.

Essa página mostra-lhe:

- Quais locatários lhe são acessíveis
- Quais permissões de função você possui
- Quais modelos lhe são acessíveis
- Quais permissões de função você possui para o pacote de regras selecionado. Caso não veja o nome do pacote de regras, não poderá visualizá-lo ou editá-lo. Se o pacote de regras não estiver mapeado para uma função, as permissões mostradas aqui corresponderão às ao nível GRAT.

Para ver como as permissões são alocadas e mantidas, clique **aqui** (novo documento).

Criar pacotes de regras

Siga estas etapas para criar um novo pacote de regras:

1. Selecione o nó na hierarquia comercial ao qual o pacote de regras pertencerá a partir da lista suspensa. Os pacotes de regras podem pertencer a qualquer nó na hierarquia.

Importante

Os nomes dos pacotes devem ser únicos entre nós e Locatários. Os nomes dos pacotes devem seguir uma convenção de nomenclatura, como incluir o nome do nó/locatário ou o nome da empresa nos nomes dos pacotes para evitar conflitos.

2. Na navegação à esquerda, selecione **Novo pacote de regras** sob o nó ou a **Solução** adequados. É preciso ter as permissões adequadas para que essa opção seja exibida.
3. No **Painel de detalhes**, insira uma propriedade de nome para o novo pacote de regras.

Importante

Há duas propriedades de nome para um pacote de regras: **Nome do pacote** e **Nome comercial**.

Nome do pacote deve estar em conformidade com as convenções de nomeação do pacote Java. De modo geral, o nome do pacote deve estar totalmente em letras minúsculas, pode conter dígitos, mas não devem iniciar com um dígito e '.' deve ser usado como separador, não os espaços. Por exemplo, **minhas.regras** e **minhasregras1** são nomes válidos, mas **Minhas regras** e **1minhas.regras** não são nomes de pacotes válidos. Cada organização deve estabelecer as suas próprias convenções de nomeação para evitar a colisão de nomes. Além disso, as palavras-chave do Java devem ser evitadas nos nomes de pacotes. Por exemplo, **meu.pacote** ou **novas.regras** não são nomes de pacotes válidos. Uma lista de palavras-chave do Java pode ser encontrada [aqui](#):

Nome comercial permite fornecer um nome simples para o pacote de regras, tal como é exibido na árvore de navegação do GRAT. Por exemplo, **Regras Acme** não é um nome de pacote de regras válido, mas é possível usar **acme** como **Nome do Pacote** e **Regras ACME** como o Nome comercial.

4. Selecione o tipo de pacote de regras que você está criando. A lista suspensa mostra que tipos já estão no repositório para o locatário selecionado. Conforme você altera o tipo, a lista de modelos para esse tipo será exibida.
5. Insira uma descrição para o pacote de regras. Os modelos de regras disponíveis (que foram criados para o nó/Locatário e correspondem ao tipo selecionado na Etapa 4) serão exibidos na tabela. Modelos com o prefixo '(*)' são modelos que foram criados no **Ambiente do locatário** e podem ser usados por todos os **Locatários**. Os desenvolvedores de regras criam modelos de regras e publicam-nos no repositório de regras usando o módulo de Desenvolvimento de modelo.

Importante

As permissões de acesso configuradas no Configuration Server também podem afetar os modelos que são exibidos.

Importante

Os usuários do GRAT podem selecionar entre várias versões de modelos, que são exibidos na caixa de diálogo **Seleção de modelo** aprimorada, juntamente com comentários de versão criados pelo desenvolvedor do modelo para ajudar a identificar diferenças entre as versões. O número de versões de um modelo que são exibidos está configurado no Genesys Administrator.

6. Se necessário, marque a caixa de seleção **Desabilitar serialização do pacote** para desligar a serialização para este pacote de regras, se estiver ativado. (A serialização de pacote pode ser habilitada/desabilitada globalmente no GRE. O cliente iWD deve sempre ter a serialização desligada.)
7. Selecione os modelos que deseja incluir e clique em **Salvar**.
8. O novo pacote de regras será exibido na navegação à esquerda. Expanda o novo pacote de regras e as seguintes opções (sujeitas às permissões definidas para o seu ID de usuário) aparecerão na pasta de pacotes de regras:
 - Configuração de teste de divisão — Use este nó para criar regras que permitem controlar como o teste de divisão se aplica à regra ao nível do pacote de regras.
 - Calendários comerciais
 - Cenários de teste
 - Implantar regras
 - Pesquisar

Será possível observar também a estrutura comercial para as quais você tem permissão de acesso.

9. Agora é possível criar regras para o seu pacote de regras.

Editar pacotes de regras

Siga estas etapas para editar um pacote de regras existente:

1. Selecione o nó ao qual o pacote de regras pertence a partir da lista suspensa.
2. Na navegação à esquerda, selecione o nome do pacote de regras.
3. No **Painel de detalhes**, é possível modificar o campo **Descrição** e atualizar quais modelos devem ser incluídos no pacote de regras. Não é possível editar o **Nome do pacote** e o tipo, mas é possível editar o **Nome comercial** do pacote de regras. Clique em **Salvar**.
4. Se necessário, marque a caixa de seleção **Desabilitar serialização do pacote** para desligar a serialização para este pacote de regras, se estiver ativado. (A serialização de pacote pode ser habilitada/desabilitada globalmente no GRE. O cliente iWD deve sempre ter a serialização desligada.)

Importante

Se desejar atualizar o pacote de regras selecionando uma versão mais recente de um modelo que já está associado ao pacote, será preciso desmarcar a versão atual do modelo antes de salvar as suas alterações. É possível ter apenas uma versão de um determinado modelo de regra associado a um pacote a qualquer momento.

Aviso

Tenha cuidado ao alterar modelos ou versões de modelos pois isso pode afetar as regras existentes. Por exemplo, uma regra existente pode usar uma condição que não existe em uma versão diferente do modelo. Consulte o desenvolvedor de modelo de regras para garantir que você está escolhendo modelos/versões corretos para o seu aplicativo. Diversas versões dos modelos estão disponíveis para seleção.

5. Para editar as regras globais que estão configuradas para o pacote de regras, selecione o nome do pacote na navegação à esquerda e, em seguida, clique na guia **Regras**. Para editar as regras criadas para um contexto comercial específico, selecione o nó na navegação à esquerda.
6. Você também pode editar os **Calendários comerciais** que estão configurados para o seu pacote de regras.

Excluir pacotes de regras

Importante

Quando um pacote é excluído, todas as referências a ele também são excluídas do **Histórico de pacotes**.

Siga estas etapas para excluir um pacote de regras existente:

1. Selecione o nó ao qual o pacote de regras pertence a partir da lista suspensa.
2. Na Árvore do Explorer, selecione o nome do pacote de regras.
3. No **Painel de detalhes**, clique em **Excluir**. Uma caixa de diálogo de confirmação será exibida.
4. Clique em **OK** para confirmar a exclusão do pacote de regras e suas regras e calendários comerciais associados.

Aviso

Ao tentar excluir um pacote de regras que ainda está implantado, haverá um aviso antes de confirmar a ação de excluir. Essa é uma oportunidade para cancelar a implantação do pacote de regras. Uma vez que o pacote de regras seja excluído do GRAT, não será mais possível usar o recurso Cancelar implantação. Neste cenário, você precisaria remover manualmente os arquivos de todos os GREs.

Implantar/cancelar implantação de pacotes de regras

Resumo

Para que as regras sejam invocadas pelos aplicativos Genesys, é necessário implantar o pacote de regras para um ou mais Mecanismos de regras da Genesys (ou para o Genesys Web Engagement, para o servidor de backend GWEB). O processo de implantação (seja escolhendo implantar imediatamente ou agendar a implantação para depois) tenta compilar o pacote de regras e o informa sobre o resultado através da mensagem pop-up **Implantação pendente**. É possível verificar o status da sua implantação observando a guia **Histórico de implantação**, que mostra o status **Pendente**. Quando uma implantação está no status pendente, não é possível cancelá-la ou desfazê-la.

Este processo permite corrigir qualquer erro antes da implantação. Além disso, ao tentar realizar uma implantação que pudesse duplicar;

- Uma implantação já agendada ou;
- Um atributo de uma implantação já agendada, como;
 - O mesmo pacote de regras
 - Para o mesmo instantâneo
 - Para o mesmo servidor/cluster de destino

uma mensagem apropriada é exibida. Você pode então alterar os atributos da sua implantação ou acessar o **Histórico de implantação** e alterar/excluir a implantação programada.

Importante

Se a sua instância GRAT fizer parte de um cluster GRAT, você também poderá visualizar, editar, excluir ou reagendar implantações que foram programadas por outros membros do mesmo cluster GRAT (a guia **Histórico de implantação** agora possui um campo **Implantado de** mostrando o último GRAT a agendar a implantação). Assim que qualquer instância GRAT que não agendou originalmente uma implantação fizer alterações em uma implantação agendada, ele assumirá a responsabilidade pela implantação.

Para usar a tela de implantação, você precisa ter permissões de implantação configuradas no Genesys Administrator.

Para implantar um pacote de regras:

1. Selecione o Locatário ao qual o pacote de regras pertence a partir da lista suspensa.
2. Na navegação à esquerda, selecione o nome do pacote de regras.
3. No pacote de regras, selecione **Implantar regras**. (O número de regras que ainda não foram incluídas em um instantâneo aparece entre parênteses.) O **Painel de detalhes** contém duas guias:
 - A guia **Implantações pendentes** permite selecionar em uma lista de instantâneos do pacote, incluindo a versão MAIS RECENTE do pacote (se configurada por um administrador), criar um novo instantâneo, exportar um instantâneo (como um arquivo XML de download para o sistema de arquivos local do usuário), excluir um instantâneo, implantar o pacote de regras, agendar uma implantação para um momento futuro e mostrar a origem do pacote. (**Exibir origem do pacote** exibe o conteúdo real do instantâneo do pacote que você está implantando. O modelo de fatos, as definições de calendário e as definições de regras serão codificados na linguagem da regra e exibidos.)

Importante

Ao criar um instantâneo, é possível escolher marcar a opção **Executar como tarefa em segundo plano**. Para pacotes de regras muito grandes, pode levar muito tempo para criar um instantâneo. Quando essa opção estiver marcada, a operação será concluída em segundo plano. Isso permite fazer outras coisas ou se desconectar. Quando o instantâneo estiver concluído, ele será exibido em **Instantâneos do pacote**.

Mesmo se **Executar como tarefa em segundo plano** estiver marcado, o pacote será primeiro criado e validado para garantir que não haja erros. Assim que a validação for bem-sucedida, o instantâneo será colocado na fila para uma tarefa em segundo plano.

Não é possível excluir o instantâneo MAIS RECENTE, tampouco um instantâneo para o qual haja uma implantação agendada.

- A guia **Histórico de implantação** exibe os detalhes sobre quando o instantâneo do pacote foi implantado anteriormente e por quem. As falhas na implantação também são exibidas na lista. Além disso, o **Histórico de implantação** exibe implantações agendadas e permite cancelar ou alterar o agendamento de implantações futuras.

Para implantar o pacote imediatamente:

1. Selecione o instantâneo do pacote ou a versão MAIS RECENTE (se disponível).

Importante

A versão MAIS RECENTE está disponível apenas se estiver configurada no Genesys Administrator. Sua organização pode optar por não a disponibilizar pois o seu conteúdo pode variar ao longo do tempo, por exemplo, entre as implantações

agendadas.

2. Clique em **Implantar agora** na guia **Implantações pendentes**.
3. Selecione o **Local** onde o instantâneo do pacote será implantado. Locais podem incluir clusters de aplicativos configurados no Genesys Administrator, clusters inteligentes especiais baseados no modelo de aplicativo Genesys_Rules_Engine_Application_Cluster ou o servidor de backend GWEB para o Genesys Web Engagement.
4. Insira alguns comentários sobre a implantação (eles serão exibidos no Histórico de implantação).
5. Clique em **Implantar**.

Uma mensagem será exibida indicando se o resultado da implantação foi sucesso, falha ou parcial. Uma implantação parcial significa que nem todos os nós no cluster receberam com êxito o pacote de regras implantado.

Para implantar o pacote mais tarde:

1. Clique em **Agendar implantação** na guia **Implantações pendentes**.
2. Selecione o **Local** (o nome do aplicativo Mecanismo de regras ou o cluster de aplicativos, ou o servidor de backend GWEB do Genesys Web Engagement) em que o instantâneo do pacote será implantado.
3. Insira a data e a hora em que você gostaria que o instantâneo do pacote fosse implantado.
4. Insira alguns comentários sobre a implantação (eles serão exibidos no **Histórico de implantação**).
5. Clique em **Agenda**.

Uma mensagem será exibida indicando se a implantação foi agendada com êxito.

Se quiser reagendar uma implantação agendada anteriormente ou quiser cancelar uma implantação agendada, isso pode ser realizado a partir da guia **Histórico de implantação**.

Para atualizar a exibição de um histórico de implantação, clique no botão **Atualizar**, ou clique no nó pertinente na Árvore do Explorer.

Para exibir detalhes de uma implantação para um cluster:

Se estiver implantando um cluster, agora é possível exibir um relatório detalhado da implantação, independentemente do seu resultado ser sucesso, falha ou parcial. Isso fornece informações úteis sobre o progresso de uma implantação: é possível clicar no resultado **Status da implantação** para observar, por exemplo, se uma conexão de servidor está temporariamente indisponível em um momento crítico ou se é necessário modificar uma configuração de tempo limite de servidor. Onde uma implantação aparece como parcial, é possível clicar no link **Parcial** no painel **Status da implantação** para exibir detalhes de GREs individuais, se e quando eles forem sincronizados de forma automática posteriormente.

Se a implantação parcial NÃO estiver configurada

Ao implantar para um cluster, o GRAT usa um protocolo de confirmação em duas fases para garantir que todos os nós GRE em execução no cluster estejam funcionando na mesma versão do pacote de regras implantado. Se alguns desses nós no cluster falhar durante a Fase 1, a Fase 2 não é confirmada.

- Fase 1 – (Implantar) Todos os GREs no cluster são notificados sobre o novo pacote de regras. Cada GRE realiza o download do novo pacote de regras e a sua compilação.
- Fase 2 – (Confirmar) Assim que todos os GREs concluírem com sucesso a Fase 1, o GRAT notifica cada GRE para ativar e confirmar o novo pacote de regras.

O Status da implantação exibe os detalhes de cada nó no cluster e se ocorreram ou não erros.

Se a implantação parcial ESTIVER configurada

O GRAT tenta implantar o pacote de regras em todos os nós GRE em execução no cluster. Se algum nó estiver inoperante ou desconectado ou a implantação falhar por qualquer motivo, o pacote de regras ainda será implantado nos nós restantes no cluster. Os GREs no cluster podem ser configurados para sincronizarem-se automaticamente quando nós desconectados forem reconectados ou quando novos nós forem adicionados ao cluster.

O GRAT ainda usa um protocolo de confirmação em duas fases. A única diferença é que, em um caso de implantação parcial, continuamos na Fase 2 para GREs que concluíram com êxito a Fase 1. O status geral é definido como `Partial` quando 1 ou mais (mas não todos) GREs no cluster falham na implantação.

- Fase 1 – (Implantar) Todos os GREs no cluster são notificados sobre o novo pacote de regras. Se algum GRE não responder, o status geral da implantação será definido como `Partial`.
- Fase 2 – (Confirmar) Para GREs que concluíram com sucesso a Fase 1, o GRAT notifica cada GRE para ativar e confirmar o novo pacote de regras.

Para exibir o relatório da implantação:

1. Clique no link **Falha/Sucesso/Parcial** na coluna **Status**.
2. Os detalhes de cada ação de implantação para cada servidor no cluster são exibidos, incluindo:
 - O nome do servidor GRE
 - O status do servidor
 - A mensagem de sucesso ou erro gerada pelo servidor
 - Tempo de implantação da Fase 1 e Fase 2 em segundos
 - Se e quando o GRE foi sincronizado automaticamente e de qual outro membro do cluster os dados do pacote de regras foram recebidos (se o recurso de sincronização automática estiver configurado).

Importante

O fuso horário para implantações agendadas é sempre o fuso horário do servidor em que o Genesys Rules Authoring Tool está instalado.

Cancelar implantação de um pacote de regras

Para usuários com os privilégios corretos, um botão **Cancelar implantação** agora é exibido na guia **Implantações pendentes**. Esse botão o permite cancelar a implantação de um pacote de regras de um único GRE ou cluster (mas não de um mecanismo ou cluster de regras de backend do GWE).

Para cancelar a implantação de um pacote de regras:

1. Clique no botão **Cancelar implantação**. Isso exibe a caixa de diálogo **Cancelar implantação**.
2. Selecione o único GRE ou cluster a partir do qual deve-se cancelar a implantação do pacote de regras e clique em **Cancelar implantação**.
3. Se o cancelamento de implantação parcial estiver habilitado, os detalhes na guia **Histórico da implantação** poderão mostrar onde ocorreu um cancelamento de implantação parcial. Clique no link **Falha/Sucesso/Parcial** na coluna **Status** para exibir o relatório de Cancelamento de implantação. O status **parcial** indica que um ou mais nós GRE estavam offline quando o pacote de regras teve sua implantação cancelada. Quando esses nós voltarem a ficar online e se a sincronização automática estiver habilitada, eles serão sincronizados automaticamente com os outros nós GRE e irão cancelar a implantação do pacote.

Importante

Se você tentar cancelar a implantação de um pacote com uma implantação pendente, uma mensagem de aviso será exibida. Cancele a implantação ou aguarde até que a implantação seja concluída antes de tentar outro cancelamento de implantação.

Se o cancelamento da implantação parcial ESTIVER habilitado:

O GRAT tenta cancelar a implantação do pacote de regras de todos os nós GRE em execução no cluster. Se algum nó estiver inoperante ou desconectado ou o cancelamento da implantação falhar por qualquer motivo, o cancelamento da implantação do pacote de regras ainda ocorrerá nos nós restantes no cluster. Os GREs no cluster podem ser configurados para sincronizarem-se automaticamente quando nós desconectados forem reconectados ou quando novos nós forem adicionados ao cluster.

O status geral é definido como `Partial` quando 1 ou mais (mas não todos) os GREs no cluster falham o cancelamento da implantação.

Se o cancelamento da implantação parcial NÃO estiver habilitado

Ao cancelar a implantação a partir de um cluster, o GRAT cancela a implantação do pacote de regras apenas se todos os membros do cluster estiverem ativos. Se algum nó estiver inativo, o cancelamento da implantação falha e o pacote de regras permanece implantado em todos os nós no cluster.

Trilha de auditoria

A guia **Trilha de auditoria** permite exibir o histórico de regras individuais, como quando elas foram atualizadas ou implantadas e por quem.

A guia **Trilha de auditoria** lista as regras que existem para o pacote de regras ou para o contexto comercial selecionado (nó), dependendo de onde você acessa a Trilha de Auditoria. A guia **Trilha de auditoria** mostra o histórico da regra selecionada no momento.

É possível selecionar a lista suspensa **ID/nome da regra** para selecionar outra regra. Para cada regra, é possível exibir o histórico da regra, incluindo diferentes versões que foram salvas e as ações, condições, parâmetros e comentários configurados.

Se uma revisão específica de uma regra foi salva como parte do instantâneo, o nome do instantâneo é exibido na coluna **Nome do último instantâneo**. Isso permite determinar o conteúdo da regra quando o instantâneo foi obtido. É possível filtrar a lista de versões de regras por **Nome do último instantâneo**, **Ação (Criada, Modificada** e assim por diante) e por nome de usuário da pessoa que realizou as alterações (**Obtido por**). É possível classificar a lista clicando em um nome de coluna, p que exibe os resultados em ordem crescente ou decrescente pela coluna selecionada.

É possível exportar o histórico da regra em um arquivo (formato de planilha). Selecione a regra a partir da lista e clique em **Exportar histórico de regras**. É possível optar por abrir o arquivo que é criado ou salvá-lo.

É possível reverter para uma versão anterior de uma regra específica: selecione a versão para a qual gostaria de reverter e clique em **Reverter**. A operação de reversão criará uma nova versão da regra com o mesmo conteúdo que a versão anterior que você selecionou. As versões originais e o histórico de auditoria serão preservados. A opção Reverter também pode ser usada para restaurar uma regra excluída anteriormente. Para isso, selecione o **ID/nome da regra** na lista suspensa e reverta a versão excluída.

Exibir o histórico de pacotes

O GRAT controla todas as alterações realizadas em um pacote e as exibe em uma guia chamada **Histórico de pacotes**. Essa guia exibe o histórico de pacotes, seja ao nível do pacote individual ou através de um Locatário. Ao nível do Locatário, a guia exibe um histórico de todas as alterações para todos os pacotes desse Locatário.

É possível filtrar por nome do pacote, nome do instantâneo, descrição da alteração ou pelo usuário que fez a alteração. Também é possível classificar por qualquer coluna, tanto os resultados completos ou em um subconjunto filtrado.

Cada alteração salva feita em um pacote gera uma nova versão do pacote. Essa versão do pacote é exibida na guia **Histórico de pacotes**, bem como na visualização **Histórico de implantação**.

Observações

- A coluna **Hierarquia comercial** exibe o nó da hierarquia de negócios ao qual o pacote de regras ou o item de pacote de regras está relacionado — por exemplo, **Casas > Vendas > Fechamento**.
- O histórico de pacotes mostra apenas alterações nos nós da estrutura de negócios aos quais o usuário tem acesso.
- A coluna **Alterar por** é visível apenas para usuários com privilégios de função relevantes.
- A coluna **Nome do instantâneo** mostra o nome da captura instantânea na qual uma alteração foi feita.

Importar modelos de regras

Importante

Um modelo exportado com uma versão do GRAT anterior à 8.1.2 não pode ser importado com a versão 9.0.0. Será necessário publicar esse modelo novamente do GRDT 8.1.3 quando o GRAT 9.0.0 estiver em execução.

É possível importar modelos de regras de um arquivo .xml. Os modelos de regras são armazenados no repositório como ativos separados de forma que eles possam ser usados por vários pacotes de regras. Os modelos de regras não são parte dos pacotes de regras em si; o pacote de regras se refere aos modelos de regras dos quais ele necessita.

Se for necessário importar modelos de regras, você deverá importá-los antes de importar os pacotes de regras uma vez que esses pacotes fazem referência aos modelos que eles usam.

Não é necessário importar os modelos de regras se você está importando do mesmo sistema (por exemplo, realizando o backup de um pacote de regras ou restaurando-o) ou de um sistema equivalente (por exemplo, um laboratório em comparação a um ambiente de produção). No entanto, se você estiver movendo um pacote de regras para um novo sistema ou enviando-o para a Genesys para manutenção, deverá importar tanto os modelos de regras quanto os pacotes de regras para que, quando eles forem importados, todos os modelos mencionados estejam disponíveis no sistema de destino.

Consulte **Importar pacote de regras** para obter detalhes sobre como importar pacotes de regras.

Importante

Para importar um modelo de pacote de regras, é necessário ter a permissão **Criar modelo**.

Para importar um modelo de pacote de regras:

1. Na navegação à esquerda, selecione a guia **Locatário**.
2. Clique em **Importar modelos de regras**. Uma caixa de diálogo é aberta na qual você seleciona o arquivo .xml a ser importado.

Importante

Clicar em **Substituir modelos já existentes no repositório** limpa o repositório de qualquer versão anterior de cada modelo antes da importação das novas para o sistema de destino. Se essa opção não estiver habilitada e um modelo existente com o mesmo nome for encontrado no repositório, uma mensagem de erro será exibida e a importação será encerrada.

Aviso

Tenha cuidado ao alterar modelos ou versões de modelos pois isso pode afetar as regras existentes. Por exemplo, uma regra existente pode usar uma condição que não existe em uma versão diferente do modelo. Consulte o desenvolvedor de modelo de regras para garantir que você está escolhendo as versões e os modelos corretos para o seu aplicativo.

Importar pacote de regras

É possível importar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e cenários de teste para esse pacote de regras, de um arquivo .xml.

Se for necessário importar modelos de regras, você deverá importá-los antes de importar os pacotes de regras uma vez que esses pacotes fazem referência aos modelos que eles usam.

Não é necessário importar os modelos de regras se você estiver importando ou exportando do mesmo sistema (por exemplo, realizando o backup ou restaurando um pacote de regras) ou a partir um sistema equivalente (por exemplo, um laboratório em comparação a um ambiente de produção). No entanto, caso esteja importando o pacote de regras para um novo sistema ou enviando-o para a Genesys para manutenção, deverá exportar tanto os modelos de regras quanto os pacotes de regras para que, quando eles forem importados, todos os modelos mencionados estejam disponíveis no sistema de destino.

Importante

A data **last-modified-by** e todo o histórico de auditoria de regras não fazem parte da exportação do pacote de regras (ou da regra). Assim, ao reimportar um pacote de regras exportado, o usuário realizando a importação torna-se o novo proprietário de cada regra criada. O histórico de pacotes e o histórico de regras não são mantidos. As regras importadas são consideradas um novo pacote de regras com um novo histórico iniciando no ponto de importação.

Consulte [Importar modelos de regras](#) para obter detalhes sobre como importar modelos de regras.

Importar pacotes de regras permite:

- Copiar uma configuração inteira de regras a partir de um ambiente de teste para um ambiente de produção.
- Realizar um backup da configuração inteira de regras antes de atualizar o Genesys Rules System.

Importante

Para importar um pacote de regras, você deve as permissões Criar pacote e Criar calendário comercial.

Para importar um pacote de regras:

1. Selecione o Locatário ao qual o pacote de regras pertence a partir da lista suspensa.

2. Na navegação à esquerda, selecione **Novo pacote de regras** na Solução adequada.
3. Clique em **Importar pacote de regras**. Uma caixa de diálogo é aberta na qual são inseridos o **Nome do pacote** e o **Nome comercial** e o arquivo .xml a ser importado é selecionado.
4. Marque **Salvar automaticamente cada regra** para salvar cada uma das regras automaticamente na importação. Essa opção só deve ser usada se o pacote de regras for conhecido como válido no sistema de destino, como ao copiar entre dois sistemas idênticos (um ambiente de laboratório em comparação a um ambiente de produção). O salvamento automático confirma cada regra no pacote sem validar que ela corresponde aos modelos subjacentes. Caso não use essa opção, cada regra será importada no estado de rascunho e deverá ser salva manualmente. Esse método mostra quaisquer erros de validação e dá ao autor de regras a oportunidade corrigi-los antes da implantação.
5. **Se a sua hierarquia comercial for desaninhada**, marque **Criação automática de hierarquia comercial durante a importação** para informar o GRAT que deve criar automaticamente qualquer nó perdido em sua hierarquia comercial para as regras que estão contidas no arquivo .xml. Por exemplo, se essa opção for selecionada e se houver uma regra que está associada ao departamento de 'Venda de Widgets' durante a importação, mas nenhum departamento do tipo estiver definido na hierarquia comercial, o GRAT tentará criá-lo durante a operação de importação. O usuário do GRAT que está executando a importação do pacote de regras deverá ter permissão para criar essa pasta. Se a caixa não estiver marcada e houver regras associadas com nós perdidos, haverá uma falha na importação.

Se sua hierarquia de negócios estiver aninhada, e você selecionar **Criação automática de hierarquia comercial durante a importação** durante o processo de importação, o GRAT garantirá que ambas as estruturas de negócios sejam compatíveis e impedirá uma importação se não forem, exibindo uma mensagem de erro informando que a hierarquia de negócios não é compatível com o pacote de regras importado.

Importante

Mesmo se o botão **Criação automática de hierarquia comercial durante a importação** estiver selecionado, o GRAT impede que o mesmo nome de nó seja criado em qualquer lugar da hierarquia — a exclusividade dos nomes dos nós comerciais em toda a hierarquia ainda é aplicada.

6. Clique em **Importar**.

Exportar modelos de regras

É possível exportar modelos de regras para um arquivo .xml. Os modelos de regras são armazenados no repositório como ativos separados de forma que eles possam ser usados por vários pacotes de regras. Os modelos de regras não são parte dos pacotes de regras em si; o pacote de regras se refere aos modelos de regras dos quais ele necessita.

Se não for necessário importar nem exportar os modelos de regras que você está importando ou exportando para o mesmo sistema (por exemplo, realizando o backup ou restaurando um pacote de regras) ou para um sistema equivalente (por exemplo, um laboratório em comparação a um ambiente de produção). No entanto, se você estiver movendo o pacote de regras para um novo sistema ou enviando-o para a Genesys para manutenção, deverá exportar tanto os modelos de regras quanto os pacotes de regras para que, quando eles forem importados, todos os modelos mencionados estejam disponíveis no sistema de destino.

Consulte **Exportar pacote de regras** para obter detalhes sobre como exportar pacote de regras.

Importante

Para exportar um modelo de regra, é preciso ter permissão para **Visualizar modelo** e ter acesso de **Leitura** para os objetos de Script que representam os modelos que estão sendo exportados.

Para exportar um modelo de regras

1. Na navegação à esquerda, selecione a guia **Locatário**.
2. Clique em **Exportar modelos de regras**.
3. Selecione o modelo ou modelos individuais que você deseja exportar. Todas as versões dos modelos selecionados serão exportadas para manter a sequência de versões no sistema de destino.
4. O arquivo .xml é gerado.

Exportar pacote de regras

Você pode exportar um pacote de regras inteiro contendo as definições de regras, calendários comerciais e referências a modelos e versões dos quais o pacote de regras depende, para um arquivo .xml.

Não é necessário exportar os modelos de regras caso esteja exportando para o mesmo sistema (por exemplo, realizando o backup ou restaurando um pacote de regras) ou para um sistema equivalente (por exemplo, um laboratório em comparação a um ambiente de produção). No entanto, se você estiver movendo o pacote de regras para um novo sistema ou enviando-o para a Genesys para manutenção, deverá exportar tanto os modelos de regras quanto os pacotes de regras para que, quando eles forem importados, todos os modelos mencionados estejam disponíveis no sistema de destino.

Importante

A data **last-modified-by** e todo o histórico de auditoria de regras não fazem parte da exportação do pacote de regras (ou da regra). O histórico de pacotes e o histórico de regras não são mantidos.

Consulte **Exportar modelos de regras** para obter detalhes sobre como exportar modelos de regras.

Exportar pacote de regras permite:

- Copiar uma configuração inteira de regras a partir de um ambiente de teste para um ambiente de produção.
- Realizar um backup da configuração inteira de regras antes de atualizar o Genesys Rules System.

Importante

Para exportar um pacote de regras, é necessário ter as permissões **Visualizar regras** e **Visualizar calendários comerciais**.

Para exportar um pacote de regras:

1. Selecione o **Localatório** ao qual o pacote de regras pertence a partir da lista suspensa.
2. Na navegação à esquerda, selecione o nó correto (ao nível no qual o pacote de regras foi definido).
3. Clique em **Exportar pacote de regras**. O pacote de regras selecionado é exportado para um único arquivo .xml.

Importante

Este arquivo .xml contém informações gerais do pacote (nome, tipo, descrição e uma lista de modelos e versões), uma lista de regras (tabelas de decisões e regras lineares) e uma lista de calendários comerciais e de regras de calendário associadas ao pacote de regras. Este arquivo .xml não contém o conteúdo dos modelos, mas contém uma referência aos nomes e versões de modelos usados.

Visão geral de calendários comerciais

Pacotes de regras podem conter um ou mais Calendários comerciais. Os calendários comerciais definem os dias e as horas úteis da organização. Eles também podem estar associados a qualquer regra no pacote.

Os calendários são classes prontas para uso disponíveis no Modelo de fatos que podem ser usados pelas Regras. Um calendário contém:

- Nome
- Fuso horário (a lista de fusos horários disponíveis é definida no tempo de execução Java)

Importante

Os calendários comerciais podem ser configurados para permitir que o fuso horário seja fornecido no momento da avaliação da regra.

Quando o usuário GRAT configura um calendário comercial, um fuso horário é escolhido junto com os outros atributos do calendário (semana normal de trabalho, exceções, feriados). Os fusos horários que respeitam o horário de verão são exibidos com um sufixo '*'.

Também é possível usar métodos padrão que podem ser acessados de dentro do modelo de regra para permitir que o ID do fuso horário seja transmitido no momento da avaliação da regra pelo aplicativo que está solicitando a avaliação da regra. Se o ID do fuso horário não for passado dessa maneira, o fuso horário 'salvo' será usado. Se o ID do fuso horário for passado, ele substituirá o fuso horário salvo e os cálculos serão feitos usando o fuso horário fornecido. Consulte **Melhorias no calendário comercial** (Práticas recomendadas/Guia do usuário).

- Dia e hora de início da semana
- Dia e hora de término da semana
- Feriados (um ou mais)
- Alteração de hora (uma ou mais)

Um **feriado** é fixo, relativo ou anual.

- Um feriado fixo contém a data do feriado, incluindo o dia, o mês e o ano, como 01/01/2015.
- Um feriado relativo contém o mês e o dia da semana do feriado e se ele está no primeiro, segundo, terceiro, quarto ou último dia daquele mês, como na terceira quinta-feira de novembro.
- Um feriado anual contém o mês e o dia do feriado, como 7 de setembro.

Uma **alteração de hora** indica como as horas úteis podem ser ajustadas em dias específicos; por exemplo, definindo meio dia de trabalho em um dia específico da semana de trabalho. Como um

feriado, uma alteração de hora pode ser fixa, relativa ou anual e contém a mesma definição de data que a definição de feriado correspondente. Além disso, a alteração de hora contém a data de início e término da data definida.

Calendários comerciais são necessários para poder definir regras baseadas em horas úteis. Por exemplo:

QUANDO a tarefa ficar ociosa por mais de *3 dias úteis* ENTÃO aumente a prioridade em 20

QUANDO *hoje é um feriado* E a tarefa é urgente ENTÃO o Grupo de agentes é 'Atendimento urgente'

As partes em *itálico* dos exemplos acima usam informações do calendário comercial.

Os tópicos a seguir explicam como trabalhar com calendários comerciais no GRAT:

- **Criar calendários comerciais**
- **Copiar calendários comerciais**
- **Excluir calendários comerciais**

Criar calendários comerciais

Siga estas etapas para criar um novo calendário comercial:

1. Vá até a navegação à esquerda, para o pacote de regras ao qual o calendário comercial pertencerá (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**).
2. Selecione **Calendários comerciais** no pacote de regras.
3. Clique em **Novo calendário**.
4. O campo **ID** é preenchido automaticamente. Esse campo não é editável.
5. Insira um **Nome** para o calendário comercial. Use algo descritivo que tornará mais fácil identificar a regra, como Semana de trabalho normal.
6. Selecione em qual dia da semana a semana começa (como segunda-feira).
7. Selecione o dia no qual a semana termina (como sexta-feira).
8. Insira a hora de início (como 9:00).
9. Insira a hora de término (como 17:00). Se a hora de término for anterior à hora de início, presume-se que o dia de trabalho abrange a meia-noite.
10. Selecione fuso horário padrão que será aplicado a este calendário comercial.

Importante

1. Os fusos horários que respeitam o horário de verão são exibidos com um sufixo *******.
2. O fuso horário selecionado pode ser substituído se um ID de fuso horário diferente for passado no momento da avaliação da regra pelo aplicativo que solicita uma avaliação pelo mecanismo de regras.

11. Se necessário, é possível configurar as regras do calendário comercial para o seu novo calendário comercial.
 - a. No painel inferior, clique no botão **Adicionar**.
 - b. Uma nova linha será exibida no painel **Regras do calendário comercial**. Insira um **Nome** para a regra, como **Ano Novo**.
 - c. Selecione o **Tipo de entrada** para a regra, como **Feriado**.
 - d. Selecione o **Posicionamento do calendário**, como **Anual** para Ano Novo, ou **Relativo** para o Memorial Day (dia de homenagem aos combatentes norte-americanos mortos em ação). Talvez seja necessário também configurar um feriado Fixo, por exemplo, se o feriado for observado em um dia diferente em um ano devido ao feriado em si cair no fim de semana.
 - e. Insira os parâmetros para a regra, como a data específica (1º de janeiro para Ano Novo), ou o dia X de um mês específico (como a terceira segunda-feira de maio, para o Memorial Day).
 - f. Configure quaisquer exceções de alteração de horário para este calendário comercial. Uma alteração de hora indica como as horas úteis podem ser ajustadas em dias específicos; por exemplo, definindo meio dia de trabalho em um dia específico da semana de trabalho. Como um feriado, uma alteração de hora pode ser fixa, relativa ou anual e contém a mesma definição de

data que a definição de feriado correspondente. Além disso, a alteração de hora contém a data de início e término da data definida.

Caso tenha configurado mais de uma exceção de feriado ou alteração de horário, desde que o calendário não seja bloqueado por outro usuário, é possível usar a coluna à direita para:

- Ajustar a ordem na qual elas são processadas. Isso permite evitar configurar exceções conflitantes. Se houver um conflito, a entrada mais alta terá prioridade.
- Faça cópias das exceções existentes e ajuste-as.
- Criar novas exceções.

g. Clique em **Salvar**.

Copiar calendário comercial

É possível copiar um calendário comercial e colar essa cópia no mesmo pacote de regras ou em um pacote de regras diferente. Siga estas etapas para copiar um calendário comercial:

1. Navegue até o pacote de regras ao qual o calendário comercial pertence na Árvore do Explorer (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**). Selecione **Calendários comerciais** no pacote de regras na Árvore do Explorer.
2. Localize o calendário comercial na lista e clique em **Copiar calendário**.
3. Caso deseje que a cópia esteja no mesmo pacote de regras, clique em **Colar calendário**. Insira um nome para o novo calendário comercial.
4. Caso deseje que a cópia esteja em um pacote de regras diferente, localize o pacote de regras desejado e selecione **Calendários comerciais** sob aquele pacote de regras. Clique em **Colar calendário**. Insira um nome para o novo calendário comercial.
5. Atualize as informações, conforme necessário. Clique em **Salvar**. Consulte **Criar calendários comerciais** para obter informações sobre os diversos campos e sobre como configurar as regras do calendário comercial.

Excluir calendários comerciais

Siga estas etapas para excluir um calendário comercial:

1. Navegue até o pacote de regras ao qual o calendário comercial pertence (verifique se selecionou o Locatário correto na lista suspensa do **Locatário**).
2. Selecione **Calendários comerciais** no pacote de regras.
3. Localize o calendário comercial na lista e clique no botão Excluir.

Visão geral dos testes de regra

Importante

Certos recursos dos quais os cenários de teste dependem não são implementados na versão inicial 9.0 do GRS. Contudo, ainda é possível importar e usar modelos desenvolvidos no GRS 8.5 para essa finalidade, mas não é possível criá-los ou editá-los na versão inicial 9.0. Consulte a versão 8.5.3 da Ajuda para obter informações mais completas.

Usar Teste de divisão para comparar resultados de regras de negócios

Certos recursos dos quais o Teste de divisão depende não estão implementados na versão inicial 9.0 do GRS. Contudo, ainda é possível importar e usar modelos desenvolvidos no GRS 8.5 para essa finalidade, mas não é possível criá-los ou editá-los na versão inicial 9.0. Consulte a versão 8.5.3 deste tópico para obter informações mais completas.

Visão geral do desenvolvimento do modelo de regra

Modelos de regras permitem que o desenvolvedor de regras de negócios crie modelos de regras que definem as condições e ações que serão usadas pelo autor de regras de negócios. O desenvolvedor cria as instruções em linguagem simples que o autor de negócios observa e as mapeia para as instruções em linguagem da regra que o mecanismo de regras executará. Para cada condição e ação da regra, o desenvolvedor decide que tipo de dados o autor de regras fornecerá. Alguns exemplos incluem se a entrada deve ser um valor inteiro, um valor numérico não inteiro, uma cadeia de caracteres, uma seleção de uma lista predefinida ou uma seleção a partir de uma lista dinâmica. Os modelos de regras são usados pelos autores de regras para criar regras para classificação e priorização de tarefas aos níveis global, departamental e processual da estrutura de negócios da solução Genesys.

Links rápidos—Trabalhar com modelos

Importar e exportar modelos

Criar e editar modelos

Publicar modelos

Exemplos de desenvolvimento de modelos

Mapeamento da linguagem da regra

Usar o Drools 5

Links rápidos—Trabalhar com editores

Editor de ações

Editor de condições

Editor de enumerações

Editor de modelos de fatos

Editor de funções

Editor de parâmetros

Conteúdo do modelo

Modelos de regra são compostos de vários elementos:

- Ações
- Condições
- Enumerações
- Modelos de fatos
- Eventos
- Funções
- Parâmetros

Ações e condições

Ações e condições definem cenários QUANDO/ENTÃO, como QUANDO um cliente é um cliente Gold, ENTÃO o destino é o GoldAgentGroup. A instrução QUANDO é a condição e a instrução ENTÃO é a ação. Uma regra pode ter zero ou mais condições e uma ou mais ações. Esse exemplo também inclui parâmetros: o status do cliente (Gold) e o nome do Grupo de agentes (GoldAgentGroup).

Sempre que uma condição contiver um mapeamento da linguagem da regra que comece com `eval( ... )`, você deve colocar toda a expressão entre parênteses, da seguinte maneira:

```
( eval(....) )
```

Isso garantirá que ela seja compilada corretamente quando usada com o operador NÃO.

Consulte [Editor de ações](#) e [Editor de condições](#).

Enumerações

As enumerações são usadas para definir listas de opções possíveis que serão exibidas ao autor de regras de negócios, quando o autor estiver criando regras baseadas no modelo de regra. Em alguns casos, a lista de opções possíveis será selecionada dinamicamente dos objetos do Genesys Configuration Server ou de fontes de dados externas. Para atividades WFM e atividades multissite, a lista de opções possíveis é recuperada dinamicamente no Genesys WFM Server. Assim, as enumerações são usadas durante a definição de uma lista discreta de opções que não serão alteradas dinamicamente.

Consulte o [Editor de enumerações](#)

Modelos de fatos

Todos os modelos de regras incluem um modelo de fatos com um ou mais fatos. Um modelo de fato estrutura o conhecimento básico sobre operações de negócios a partir de uma perspectiva de negócios. Um modelo de fatos se concentra em conexões lógicas (chamadas de fatos) entre os principais conceitos do negócio. Indica o que você precisa saber sobre as operações de negócios para poder oferecer suporte (ou realmente realizar) a essas operações.

Um bom modelo de fatos o informa como estruturar seu pensamento (ou conhecimento) básico sobre o processo de negócios com base em um vocabulário padrão. Ao usar o vocabulário padrão, focado nos negócios, garantimos que as próprias regras de negócios possam ser bem compreendidas pelas principais partes interessadas, como analistas de negócios. Por exemplo, em sua empresa, você pode ter um Fato que representa um Cliente e outro Fato que representa um Pedido.

O Cliente pode ter campos como nome, idade, local, classificação de crédito e idioma preferido. O Pedido pode ter campos como valor e data do pedido. Uma regra pode ser construída usando esses valores, como:

Quando o Cliente tiver pelo menos 21 anos e o pedido for > 100,00, convide o cliente a participar de uma pesquisa.

Consulte [Editor de modelos de fatos](#)

Eventos

Para oferecer suporte ao Processamento de eventos complexos, os desenvolvedores de modelos precisam poder designar certos fatos como eventos e os autores de regras precisam alterar a maneira como o DRL é gerado quando um fato é designado como um evento.

Portanto, o modelo de fatos foi melhorado para incluir eventos e a caixa de diálogo do modelo de fato agora inclui um botão Criar evento. Um evento possui os seguintes campos:

- Nome
- Descrição
- Uma lista opcional de Propriedades.
- Metadados de expiração definidos pelo usuário para o evento

No GRAT, a tag de metadados `@role` determina se estamos lidando com um fato ou evento. A tag metadados `@role` pode aceitar dois valores possíveis:

- `fato` — Atribuir a função de fato declara que o tipo deve ser tratado como um fato regular. Fato é a função padrão.
- `evento`— Atribuir a função de evento declara que o tipo deve ser tratado como um evento.

Funções

Funções são usadas para definir outros elementos além de Condições e Ações, por exemplo, quando a análise de registros de data e hora é necessária. O Editor de funções permite gravar funções Java específicas para diferentes propósitos, para uso em modelos de regras. As funções especificadas podem ser usadas nos mapeamentos da linguagem da regra (consulte [Mapeamento da linguagem da regra](#)).

Quando os modelos de regra são criados, o desenvolvedor da regra os publica no Repositório de regras, disponibilizando-os no GRAT para os usuários comerciais criarem regras.

Ações e condições podem conter parâmetros. Há suporte para vários tipos de parâmetros.

Certos tipos de parâmetros dinâmicos que se referem a fontes de dados externas exigem que um Perfil seja selecionado. O Perfil é definido como um objeto de Script do tipo Coleta de dados e fornece informações de conexão que permitem ao GRAT recuperar esses dados dinâmicos da fonte de dados externa. As próximas seções descrevem como configurar Perfis para parâmetros de banco de dados, serviço Web e Workforce Management.

Consulte [Editor de funções](#).

Parâmetros

Os parâmetros banco de dados, serviço Web e Workforce Management são usados nas ações e condições.

Parâmetros de banco de dados

Propriedades de parâmetro do banco de dados

Propriedade	Obrigatório/opcional	Descrição
driver	Obrigatório	O nome do driver JDBC a ser usado. Por exemplo, <code>com.mysql.jdbc.Driver</code>
url	Obrigatório	A URL do banco de dados no formato correto para o driver JDBC a ser usado.
username	Obrigatório	Um nome de usuário válido para conectar-se ao banco de dados.
password	Obrigatório	A senha necessária para o usuário conectar-se ao banco de dados.
initSize	Opcional	O tamanho inicial do conjunto de conexões. O padrão é 5.
maxSize	Opcional	O tamanho máximo do conjunto de conexões. O padrão é 30.
waitTime	Opcional	O tempo máximo (em milissegundos) para aguardar a obtenção de uma conexão. O padrão é 5000.

Em geral, os valores opcionais não precisam ser definidos ou alterados.

É possível configurar apenas os parâmetros do banco de dados com uma instrução SQL SELECT. Qualquer outro tipo de instrução falhará quando configurada.

Parâmetros do serviço Web

No Configuration Server, os scripts de Serviço Web devem ter uma seção chamada webservice. A tabela abaixo lista as propriedades que podem ser especificadas para os parâmetros de serviço Web.

Propriedades de parâmetro do serviço Web

Propriedade	Obrigatório/opcional	Descrição
host	Obrigatório	O host para o serviço.
base-path	Obrigatório	O caminho base para acessar o serviço.
protocol	Opcional	O padrão é http.
port	Opcional	O padrão é 80.
headers	Opcional	Quaisquer cabeçalhos HTTP personalizados necessários para o serviço.
parameters	Opcional	Quaisquer configurações HTTP personalizadas necessárias para

Propriedade	Obrigatório/opcional	Descrição
		ajustar a conexão.

Em geral, os valores de parâmetros não precisam ser definidos ou alterados. Cabeçalhos e parâmetros são listas no seguinte formato:

key:value[,key:value]

Aviso:	Você não pode especificar cabeçalhos ou parâmetros que contêm ' , ' no valor. Aviso: Caso esteja enviando uma mensagem para o serviço, será esperado que Tipo de conteúdo seja especificado no cabeçalho, pois define a interação geral da mensagem com o servidor. Um conjunto de caracteres opcional pode ser incluído. Por exemplo, Content-Type:applicaton/json;charset=UTF-8.
---------------	--

É necessário definir completamente a mensagem a ser enviada e ela deve ser constante. Nenhuma substituição de variável é realizada. A Consulta de XPath é usada para extrair valores da resposta do servidor. A resposta deve estar em XML ou JSON, caso contrário, isso não funcionará. Uma consulta de XPath válida deve ser especificada para a resposta. Isso depende inteiramente do serviço com o qual você faz interface.

Observação:	A mensagem é enviada para o servidor somente uma vez por sessão. Isso é feito por razões de desempenho e porque espera-se que os valores na resposta sejam relativamente constantes.
--------------------	--

O caminho para o parâmetro é adicionado ao base_path no script.

Por exemplo:

Se o script conter:

```
host = api.wunderground.com
base_path = /auto/wui/geo/ForecastXML/
```

e o Desenvolvimento de modelos especificar:

```
query type = List
XPath Query = //high/fahrenheit/text()
HTTP Method = GET
path = index.xml?query=66062
message (not set)
```

então a mensagem enviada é:

```
GET /auto/wui/geo/ForecastXML/index.xml?query=66062 HTTP/1.1
```

Isso retornará as máximas da semana em Fahrenheit:

```
81
77
81
```

81
83
85

Parâmetros de Workforce Management

No Configuration Server, os scripts do Workforce Management devem ter uma seção chamada wfm. A tabela 4 lista as propriedades que você pode especificar para os parâmetros Workforce Management.

Propriedades de parâmetro Workforce Management

Propriedade	Obrigatório/opcional	Descrição
wfmCfgServerAppName	Obrigatório	Nome do aplicativo do Configuration Server para o servidor WFM.
wfmCfgServerUserName	Obrigatório	Nome de usuário do Configuration Server.
wfmCfgServerPassword	Obrigatório	Senha do Configuration Server.
wfmServerUrl	Obrigatório	URL do WFM Server.

Ao configurar um novo parâmetro do tipo 'Workforce Management', basta nomear o parâmetro e escolher o perfil WFM (objeto de script recém-criado) na lista suspensa. Quando o autor estiver usando esse parâmetro, o GRAT buscará a lista atual de Atividades WFM no WFM Server e as exibirá para o autor de regras.

Suporte para tipos de modelo definidos pelo usuário

O GRAT exibe automaticamente a lista de tipos de modelo publicados e os designers de modelos podem selecionar esses tipos de modelos definidos pelo usuário ou definir novos, de acordo com as suas próprias necessidades.

Versões de modelo

Cada vez que um modelo de regra é publicado, uma nova versão é criada no repositório. O autor de regras poderá selecionar qualquer versão do modelo na caixa de diálogo Seleção de modelo ao criar um pacote de regras. Essa caixa de diálogo mostra as últimas N versões de um modelo, em que N é um valor configurado usando a opção de configuração **display-n-template-versions** no Genesys Administrator.

Ao publicar versões mais recentes do modelo de regra, esteja ciente de que determinadas alterações podem afetar as regras que já foram criadas usando a versão anterior do modelo. Cuidado para não fazer alterações que possam anular regras existentes, a menos que essas alterações sejam comunicadas ao autor de regras.

Por exemplo, se o Modelo de regra versão 1 tiver uma condição removida posteriormente na versão 2

e se uma regra já tiver sido criada usando essa condição, ela não será mais compilada se o autor de regras atualizar para o Modelo de regra versão 2.

Por exemplo, se a configuração foi definida para mostrar as últimas 3 versões de um modelo, o modelo atualmente selecionado é GRS Template versão 2 e existem 5 versões no repositório, mostraria as versões GRS Template 5, 4 e 3, bem como GRS Template versão 2. Os usuários podem escolher entre as versões 3, 4 ou 5.

Comentários da versão

Para fornecer detalhes sobre as diferenças entre as versões do modelo, os desenvolvedores de modelo de regras podem publicar um comentário da versão que descreva as alterações específicas feitas nas versões individuais do modelo. Esse comentário de versão aparece no GRAT na tabela Seleção de modelo e pode ser editado pelo autor de regras no GRAT.

Importar e exportar modelos

Importar um projeto de modelo

Use este painel para importar modelos exportados anteriormente. Por exemplo, modelos de amostra ou modelos exportados de outro sistema.

1. Selecione o nó **Modelos** na árvore de navegação à esquerda.
2. Selecione o botão **Importar modelos**.
3. Selecione o botão **Escolher arquivo**, navegue até seu sistema de arquivos e selecione o projeto de modelo a ser importado.
4. Clicar em **Substituir modelos já existentes no repositório** limpa o repositório de qualquer versão anterior de cada modelo antes da importação das novas para o sistema de destino. Se essa opção não estiver habilitada e um modelo existente com o mesmo nome for encontrado no repositório, uma mensagem de erro será exibida e a importação será encerrada.
5. Clique em **Concluir**.

Aviso

Tenha cuidado ao alterar modelos ou versões de modelos pois isso pode afetar as regras existentes. Por exemplo, uma regra existente pode usar uma condição que não existe em uma versão diferente do modelo. Consulte o desenvolvedor de modelo de regras para garantir que você está escolhendo as versões e os modelos corretos para o seu aplicativo.

Exportar um projeto de modelo

1. Selecione o nó **Modelos** na navegação à esquerda.
2. Selecione os projetos de modelo que deseja exportar.
3. Nomeie o arquivo de exportação e o local para o qual o projeto do modelo será exportado.
4. Clique em **Concluir** para completar a exportação.

Importante

Cabe ao desenvolvedor do modelo de regra garantir que os modelos que eles desenvolvem não tenham problemas com a colisão de nomes. Por exemplo, nomes de

função, assinaturas de método Java e fatos devem ter nomes diferentes, mesmo que estejam em modelos diferentes, porque um autor de regra pode criar regras com base em vários modelos. Os nomes não devem ser duplicados e devem ser comunicados ao autor de regras quais modelos/versões usar e em que combinação.

Criar e editar modelos

Os modelos de regra são criados como projetos no nó **Projetos de modelo** na guia **Desenvolvimento de modelos** do GRAT.

Criar um novo projeto de modelo

1. Insira um nome para o novo projeto de modelo. Os nomes de modelo devem ser únicos no Locatário. Clique em **Avançar**.
2. Selecione o tipo de modelo que você está criando na lista suspensa ou digite o nome de um novo tipo de modelo a ser criado. Para criar um modelo iWD, selecione iWD:
3. Clique em **Concluir**. O novo projeto de modelo será exibido.

Editar e configurar modelos de regras

Uma vez criado, o modelo de regra é exibido no painel de navegação à esquerda. A expansão do modelo exibe uma lista de componentes que podem ser configurados. Clique duas vezes no tipo de componente para abrir o Editor apropriado e começar a configurar os componentes.

Renomear modelos de regras

Para renomear um modelo de regra, basta editar o seu nome. Uma cópia do modelo anteriormente nomeado permanece no repositório.

Importante

Nomes de modelo duplicados não são permitidos no mesmo locatário, mas em locatários diferentes são permitidos. A criação de um nome duplicado renomeará o projeto, mas o nome publicado no GRAT é definido em Propriedades **projeto/propriedades/** modelo.

Excluir modelos de regras

Modelos de regras podem ser excluídos, desde que:

- O usuário tenha permissão para excluir modelos de regra e;

-
- A regra não seja usada em nenhum pacote de regras.

Publicar modelos

É necessário publicar o modelo para que ele esteja disponível para os usuários comerciais criarem regras. A publicação também é o mecanismo preferido para compartilhar os modelos, para que outros desenvolvedores possam editar ou modificar os modelos, se necessário. A visibilidade do modelo é determinada pelas permissões de acesso. Essas permissões são definidas por um administrador no Configuration Manager ou no Genesys Administrator. Cada modelo possui um objeto de Script correspondente no Genesys Configuration Server para o qual o controle de acesso pode ser configurado.

Os autores de regra podem selecionar versões anteriores dos modelos de regra publicados. Opcionalmente, é possível publicar um comentário de versão para um modelo específico a fim de informar autores de regras sobre as diferenças específicas entre versões individuais de um modelo.

Publicar o modelo no repositório

1. Selecione o modelo na navegação à esquerda e clique em **Publicar**.
2. Adicione um comentário de versão.
3. Clique em **Publicar** para publicar o modelo.

Exemplos de desenvolvimento de modelos

Esta seção fornece alguns exemplos do que um desenvolvedor de regras pode configurar na guia Desenvolvimento de modelo. Para obter informações específicas sobre como os modelos de regras devem ser usados com a solução Genesys intelligent Workload Distribution (iWD), consulte o guia *iWD* e *Genesys Rules System*.

Exemplo 1: Condição e Ação

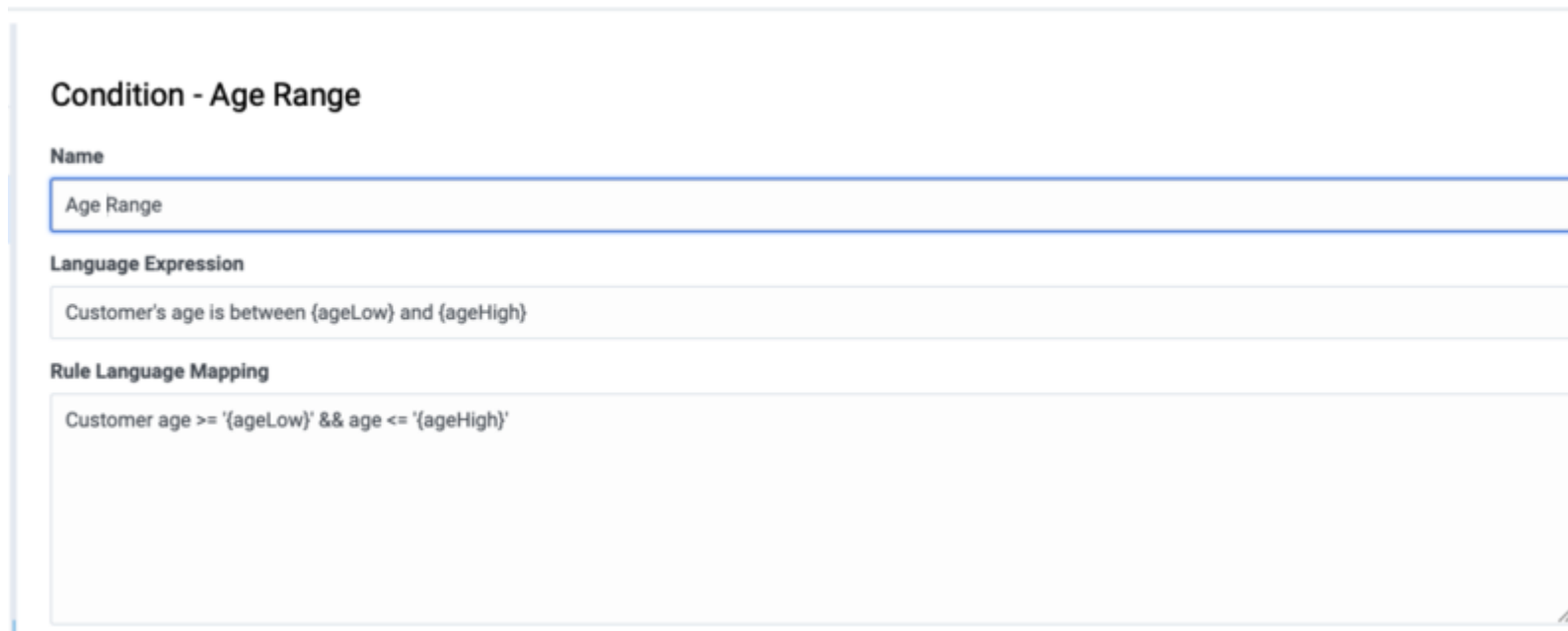
Condição de faixa etária

Se a idade de um cliente estiver dentro de um intervalo específico, um grupo de agentes específico será direcionado. Nesse cenário, a condição é se a idade do cliente está dentro do intervalo. No Genesys Rules Development Tool, as condições seriam configuradas da seguinte forma.

```
Name: Age Range  
Language Expression: Customer's age is between {ageLow} and {ageHigh}  
Rule Language Mapping: Customer(age >= '{ageLow}' && age <= '{ageHigh}')
```

Não use a palavra 'end' em expressões de linguagem da regra. Isso causa erros de análise de regra.

A figura abaixo mostra como essa condição apareceria no desenvolvimento de modelos do GRAT.



Condition - Age Range

Name

Age Range

Language Expression

Customer's age is between {ageLow} and {ageHigh}

Rule Language Mapping

Customer age >= '{ageLow}' && age <= '{ageHigh}'

Condição do autor da chamada

Além de testar a existência do autor da chamada, a próxima condição também cria a variável `$Caller`, usada por ações para modificar o fato do autor da chamada. O autor da chamada modificado retornará nos resultados da solicitação de avaliação.

Não é possível criar uma variável mais de uma vez dentro de uma regra e não é possível usar variáveis em ações se as variáveis não tiverem sido definidas na condição.

Name: Caller
Language Expression: Caller exists
Rule Language Mapping: `$Caller:Caller`

A figura abaixo mostra como essa condição apareceria no desenvolvimento de regras do GRAT.

Condition - Caller

Name
Caller

Language Expression
Caller exists

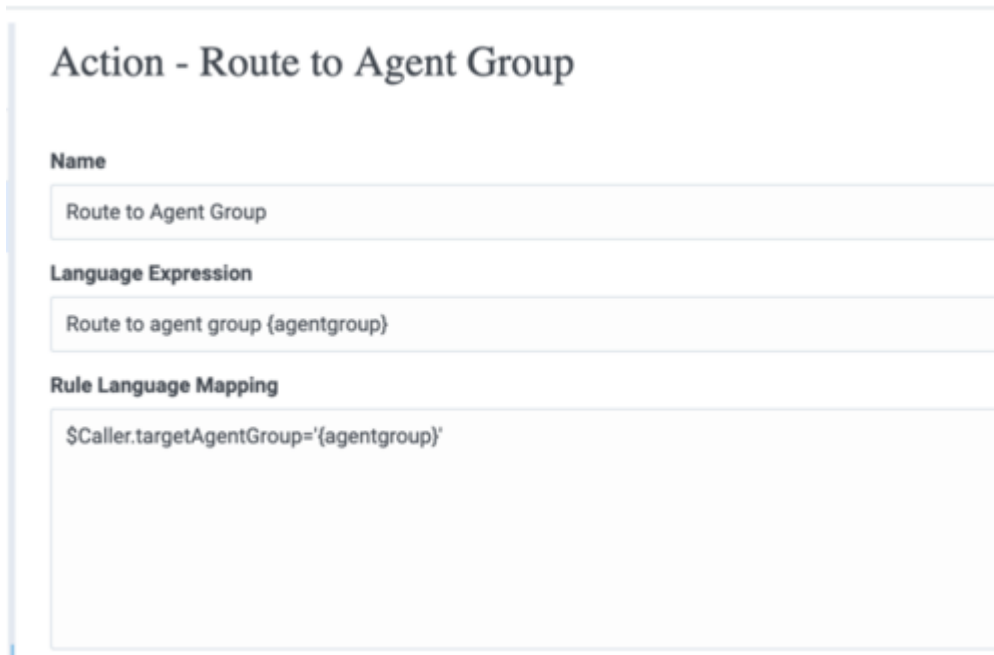
Rule Language Mapping
\$Caller:Caller()

Ação do grupo de agentes de destino

A ação seria configurada da seguinte forma:

Name: Route to Agent Group
Language Expression: Route to agent group {agentGroup}
Rule Language Mapping: \$Caller.targetAgentGroup='{agentgroup}'

A figura abaixo mostra como essa ação seria exibida no desenvolvimento de regras do GRAT.



Action - Route to Agent Group

Name

Route to Agent Group

Language Expression

Route to agent group {agentgroup}

Rule Language Mapping

\$Caller.targetAgentGroup='{agentgroup}'

Parâmetros

A condição neste exemplo possui dois parâmetros:

- {ageLow}
- {ageHigh}

A ação possui o parâmetro {agentGroup}. A captura de tela do Editor de parâmetros mostra um exemplo de parâmetro {ageHigh}.

Parameter - ageHigh

Name

Description

Type

Minimum

Maximum

Custom tooltip

Como funciona

O funcionamento do exemplo anterior é o seguinte:

1. O desenvolvedor de regras cria um modelo de fatos (ou o modelo de fatos pode ser incluído como parte de um modelo de regras pronto para uso com uma solução específica da Genesys). O modelo de fato descreve as propriedades do fato Customer e do fato Caller. Nesse caso, podemos ver que o fato Customer possui uma propriedade chamada age (provavelmente um número inteiro) e o fato Caller possui uma propriedade chamada targetAgentGroup (provavelmente uma cadeia de caracteres).

2. O desenvolvedor da regra cria os parâmetros `ageLow` e `ageHigh`, que se tornarão campos editáveis que o usuário comercial preencherá ao criar uma regra comercial que use esse modelo de regra. Esses parâmetros seriam do tipo `Input Value`, onde o `Value Type` provavelmente seria um número inteiro. O desenvolvedor de regras pode ter a opção de restringir os possíveis valores que o usuário comercial poderá inserir digitando um mínimo e/ou um máximo.
3. O desenvolvedor da regra também cria o parâmetro `agentGroup`, que provavelmente será uma lista selecionável pela qual o usuário comercial será apresentado a uma lista suspensa de valores extraídos do Genesys Configuration Server ou de uma fonte de dados externa. O comportamento desse parâmetro depende do tipo de parâmetro selecionado pelo desenvolvedor de regras.
4. O desenvolvedor da regra cria uma ação e uma condição para a regra, conforme descrito anteriormente. A ação e a condição incluem mapeamentos de linguagem da regra que instruem o Mecanismo de regras sobre quais fatos usar ou atualizar com base nas informações passadas para o Mecanismo de regras como parte (da solicitação de avaliação de regra vinda de um aplicativo cliente, como um aplicativo SCXML).
5. O desenvolvedor da regra publica o modelo de regra no Repositório de regras.
6. O autor de regras usa esse modelo de regra para criar uma ou mais regras de negócios que utilizam as condições e ações nas combinações necessárias para descrever a lógica comercial que o autor das regras deseja aplicar. Nesse caso, as condições e ações descritas anteriormente provavelmente seriam usadas juntas em uma única regra, mas as condições e ações também poderiam ser combinadas com outras condições e ações disponíveis para criar políticas de negócios diferentes.
7. O autor de regras implanta o pacote de regras no servidor de aplicativo do Mecanismo de regras.
8. Um aplicativo cliente, como um aplicativo VXML ou SCXML, chama o Mecanismo de regras e especifica o pacote de regras a ser avaliado. A solicitação para o Mecanismo de regras incluirá os parâmetros de entrada e saída para o modelo de fatos. Neste exemplo, teria que incluir a propriedade `idade` do fato do Cliente. Essa idade pode ter sido coletada por meio do GVP ou extraída de um banco de dados do cliente antes do Mecanismo de regras ser chamado. Com base no valor da propriedade de fato `Customer.age` que é passada para o Mecanismo de regras como parte da solicitação de avaliação de regras, o Mecanismo de Regras avaliará um conjunto específico de regras que foram implantadas. Neste exemplo, será avaliado se `Customer.age` fica entre os limites inferior e superior que o autor de regras especificou na regra.
9. Se a regra for avaliada como verdadeira pelo Mecanismo de regras, a propriedade `targetAgentGroup` do fato `Autor` da chamada será atualizada com o nome do Grupo de agentes selecionado pelo autor de regras de negócios quando a regra foi escrita. O valor da propriedade `Caller.targetAgentGroup` será transmitido de volta ao aplicativo cliente para processamento adicional. Neste exemplo, talvez o valor de `Caller.targetAgentGroup` seja mapeado para uma variável de aplicativo `Composer` que será então passada para o bloco `Meta` para solicitar ao Genesys Universal Routing Server que direcione esse grupo de agentes.

Exemplo 2: Função

As funções são usadas para elementos mais complexos e são escritas em Java. Neste exemplo, a função é usada para comparar datas. Ela seria

configurada da seguinte forma:

```
Name: compareDates
Description: This function is required to compare dates.
Implementation:
import java.util.Date;
import java.text.SimpleDateFormat;

function int _GRS_compareDate(String a, String b) {
    // Compare two dates and returns:
    // -99 : invalid/bogus input
    // -1 : if a < b
    // 0 : if a = b
    // 1 : if a > b

    SimpleDateFormat dtFormat = new SimpleDateFormat("dd-MMM-yyyy");
    try {
        Date dt1= dtFormat.parse(a);
        Date dt2= dtFormat.parse(b);
        return dt1.compareTo(dt2);
    } catch (Exception e) {
        return -99;
    }
}
```

Para classes fornecidas pelo usuário, o arquivo .jar deve estar no CLASSPATH para GRAT e GRE.

A figura abaixo mostra como essa função apareceria no desenvolvimento de regras do GRAT.

Function - compareDates

Name

compareDates

Description

Required for date field comparisons

Implementation

```
import java.util.Date;
import java.text.SimpleDateFormat;

function int _GRS_compareDate(String a, String b) {
    // Compare two dates and returns:
    // -99 : invalid/bogus input
    // -1 : if a < b
    // 0 : if a = b
    // 1 : if a > b

    SimpleDateFormat dtFormat = new SimpleDateFormat("dd-MMM-yyyy");
    try {
        Date dt1 = dtFormat.parse(a);
        Date dt2 = dtFormat.parse(b);
        return dt1.compareTo(dt2);
    } catch (Exception e) {
        return -99;
    }
}
```

Exemplo 3: Usar um Objeto JSON

Os desenvolvedores de modelos podem criar modelos que permitem que aplicativos clientes transmitam Fatos para GRE como objetos JSON sem precisar mapear cada campo para o modelo de fatos explicitamente.

Importante

Regras baseadas em modelos que usam essa funcionalidade não oferecem suporte à criação de cenários de teste no momento.

Este exemplo mostra como criar um modelo que contém uma classe (chamada MyJson) para transmitir um objeto JSON.

Início

1. Crie a seguinte classe e importe-a para um modelo de regra:

```
package simple;
import org.json.JSONObject;
import org.apache.log4j.Logger;

public class MyJson {
    private static final Logger LOG = Logger.getLogger(MyJson.class);
    private JSONObject jsonObject = null;

    public String getString( String key) {
        try {
            if ( jsonObject != null)
                return jsonObject.getString( key);
        } catch (Exception e) {
            LOG.debug("Oops, jsonObject null ");
            return null;
        }
    }

    public void put( String key, String value) {
```

```

        try {
            if (jsonObject == null) {
                jsonObject = new JSONObject();
            }
            jsonObject.put( key, value);
        } catch (Exception e) {
        }
    }
}

```

2. Crie um fato de demonstração com o mesmo nome (MyJson) no modelo.

3. Adicione a MyJson.class ao caminho da classe do GRAT e do GRE.

4. Crie a seguinte condição e ação:

```

Is JSON string "{key}" equal "{value}"      eval($MyJson.getString("{key}").equals("{value}"))
Set JSON string "{key}" to "{value}"        $MyJson.put("{key}", "{value}");

```

5. Use esta condição e ação em uma regra no pacote json.test. Com isso, o seguinte será gerado:

```

rule "Rule-100 Rule 1"
salience 100000
agenda-group "level0"
dialect "mvel"
when
    $MyJson:MyJson()
    and (
        eval($MyJson.getString("category").equals("test"))
    )
then
    $MyJson.put("newKey", "newValue");
end

```

6. Implante o pacote json.test no GRE.

7. Execute a seguinte solicitação de execução do RESTClient:

```

{"knowledgebase-request":{
  "inOutFacts":{"anon-fact":{"fact":{"@class":"simple.MyJson", "jsonObject":
    {"map":{"entry":[{"string":["category","test"]}, {"string":["anotherKey","anotherValue"]}]}}}}}}}

```

8. A seguinte resposta é gerada:

```
{"knowledgebase-response":{"inOutFacts":{"anon-fact":[{"fact":{"@class":"simple.MyJson","jsonObject":{"map":{"entry":[{"string":["category","test"]},{"string":["newKey","newValue"]}, {"string":["anotherKey","anotherValue"]}]}}]},"executionResult":{"rulesApplied":{"string":["Rule-100 Rule 1"]}}}}}}
```

Fim

Exemplo 4: Desenvolver modelos para habilitar cenários de teste

Importante

Não há suporte para a criação e edição de eventos na versão inicial 9.0.0 do GRAT, portanto, modelos que suportam cenários de teste não podem ser desenvolvidos. No entanto, os modelos que suportam cenários de eventos/teste criados no Genesys Rules Development Tool 8.5 ainda podem ser desenvolvidos no GRDT, importados para o GRAT 9.0 e usados para criar pacotes de regras com suporte para eventos/cenários de teste.

Para obter mais informações sobre este tópico, consulte *Desenvolver modelos para habilitar cenários de teste* na Ajuda do GRDT 8.1.3.

Mapeamento de diversas instâncias de um parâmetro de regra para uma única definição de parâmetro

No momento da criação dos parâmetros, em vez de criar os parâmetros `ageLow` e `ageHigh`, o desenvolvedor do modelo de regra pode criar um único parâmetro `{age}` e usar a notação de caractere sublinhado mostrada no exemplo abaixo para criar índices do parâmetro para cenários nos quais várias instâncias de parâmetro com o mesmo tipo (idade) são necessárias (mais comumente usadas com intervalos). Por exemplo: `{age_1}`, `{age_2}` . . . `{age_n}`. Esses campos serão editáveis. Esse recurso é normalmente usado para definir intervalos com mais eficiência.

Fato/condição

Os fatos podem ser referenciados em condições e ações prefixando o nome do fato com um sinal `$`. Por exemplo, o fato `Caller` pode ser referenciado pelo nome `$Caller`. O GRS gerará implicitamente uma condição que associa a variável `$Caller` ao fato `Caller` (ou seja, `$Caller:Caller()`).

A condição `$Caller:Caller()` exige um objeto `Caller` como entrada na execução de regras para que essa condição seja avaliada como verdadeira.

Predefinição:BestPractices

Mapeamento da linguagem da regra

Quando os desenvolvedores de regras criam as condições ou ações em um modelo de regra, eles inserem o mapeamento da linguagem da regra. Com o Genesys Rules System 9.0, use as versões 5.5 da Linguagem da regra do Drools, encontradas aqui:

<http://downloads.jboss.com/drools/docs/5.5.FINAL/drools-expert/html/ch04.html>

Como os URLs mudam com frequência, pesquise no site Drools pelo Guia do usuário do Drools Expert e, em seguida, consulte o índice desse guia para obter informações sobre a Linguagem da regra do Drools.

O mapeamento da linguagem da regra não é visível para o usuário comercial quando ele está criando regras no Genesys Rules Authoring Tool. Em vez disso, os autores de regra verão a Expressão da linguagem inserida pelo desenvolvedor do modelo de regra. A expressão da linguagem é uma descrição em linguagem simples que usa terminologia relevante para o usuário comercial, em vez de códigos de baixo nível. O mapeamento da linguagem da regra é fornecido nos exemplos na seção a seguir.

Expressões de linguagem

Ao criar um modelo de regra, a Expressão da linguagem não pode usar o caractere de parêntese aberto ou fechado. Por exemplo, a expressão:

```
More than {parCallLimit} calls within {parDayLimit} day(s)
```

resultará em um erro ao tentar salvar a regra no GRAT. Mas se você deseja que o usuário comercial veja os parênteses no GRAT, poderá usar caracteres de barra invertida na expressão da linguagem. Por exemplo:

```
More than {parCallLimit} calls within {parDayLimit} day\s\).
```

Construções HTML

Por motivos de segurança, o GRAT não permite que nenhum comando HTML seja inserido como parâmetros de uma regra. Por exemplo, se uma condição for:

```
Customer requests a callback on {day}
```

e {day} estiver definido como uma cadeia de caracteres, não permitiríamos que um autor de regra inserisse a cadeia de caracteres:

```
Customer requests a callback on <b>Tuesday</b>.
```

Todas as construções HTML serão removidas da cadeia de caracteres. Isso se aplica a parâmetros de cadeia de caracteres, bem como a parâmetros de lista dinâmica, como atributos

comerciais, banco de dados ou serviços Web.

Usar o Drools 5

O Drools 5 introduz o conceito de palavras-chave fixas e flexíveis.

Palavras-chave fixas

As palavras-chave fixas são reservadas — não é possível usar uma palavra-chave fixa ao nomear objetos de domínio, propriedades, métodos, funções e outros elementos usados no texto da regra. A seguinte lista de palavras-chave fixas deve ser evitada como identificadores ao escrever regras:

- true
- false
- null

Palavras-chave flexíveis

As palavras-chave flexíveis são reconhecidas apenas em seu contexto, permitindo-o usar essas palavras em qualquer outro lugar se desejar, embora a Genesys recomende evitá-las, se possível, para não gerar confusão. A lista de palavras-chave flexíveis é:

- | | | | | | | | | |
|------------------|--------------------|------------|--------------|------------|--------|--------------|-----------|--------|
| • lock-on-active | • activation-group | • package | • attributes | • template | • eval | • exists | • action | • init |
| | | • import | • rule | • query | • not | • forall | • reverse | |
| • date-effective | • agenda-group | • dialect | • extend | • declare | • in | • accumulate | • result | |
| | • ruleflow-group | • salience | • when | • function | • or | • collect | • end | |
| • date-expires | | • enabled | • then | • global | • and | • from | • over | |
| • no-loop | • entry-point | | | | | | | |
| • auto-focus | • duration | | | | | | | |

É possível usar essas palavras (fixas e flexíveis) como parte de um nome de método em maiúsculas e minúsculas, por exemplo `notSomething()` ou `accumulateSomething()` sem problemas.

Escapando palavras-chave fixas

Embora seja improvável que as três palavras-chave fixas acima sejam usadas nos seus modelos de domínio existentes, caso seja absolutamente necessário usá-las como identificadores em vez de palavras-chave, a linguagem do DRL fornecerá a capacidade de escapar de palavras-chave fixas no texto da regra. Para escapar de uma palavra, basta colocá-la entre acentos graves, assim:

```
Holiday( `true` == "yes" ) //
```

Observe que o Drools resolverá essa referência ao método:

```
Holiday.isTrue()
```

Editor de ações

O Editor de ações permite criar e editar ações de regras. Cada ação contém os mesmos campos:

- **Nome** — O nome da ação.
- **Expressão da linguagem** — A descrição da linguagem simples da ação que o autor de regras visualiza ao construir uma regra comercial na Rules Authoring Tool.
- **Mapeamento da linguagem da regra** — A ação expressa em código. Consulte Mapeamento da linguagem da regra para obter mais informações.

Ao configurar ações, os parâmetros podem ser usados no Mapeamento da linguagem da regra e na Expressão da linguagem.

Por exemplo, a ação **Agente de destino** pode ser configurada da seguinte maneira:

- **Nome** — Agente de destino
- **Expressão da linguagem** — Agente '{agent}' de destino específico
- **Mapeamento da linguagem da regra** — `$Caller.targetAgent='{agent}'`

Neste exemplo, {agent} é um parâmetro.

Importante

O exemplo acima também pressupõe que existe um fato chamado **Autor da chamada** com um campo chamado **targetAgent**.

Ações em regras lineares

Para uma regra linear, há um limite máximo 6 de colunas de dados de parâmetros (incluindo rótulos de texto estático). Então, por exemplo, se sua expressão for:

```
Set customer data to: {parm1} and {parm2} and  
{parm3} and {parm4}
```

Nesse caso, {parm4}, sendo o 7º parâmetro, não será exibido. Reformule as suas ações para que caibam dentro desses limites.

Aviso

Ao configurar ações para um 'parâmetro operacional', é importante que você *não*

coloque a variável entre aspas duplas ou simples na expressão da linguagem ou no mapeamento da linguagem da regra. Consulte o exemplo [aqui](#).

Editor de condições

O Editor de condições permite criar e editar condições de regra. Cada condição contém os mesmos campos:

- **Nome** — O nome da condição.
- **Expressão da linguagem** — A expressão da linguagem simples da condição que o autor de regras observa ao construir uma regra de negócios na guia Criação de regras.
- **Mapeamento da linguagem da regra** — A condição expressa em código. Consulte Mapeamento da linguagem da regra para obter mais informações.

Ao configurar condições, os parâmetros podem ser usados no Mapeamento da linguagem da regra e na Expressão da linguagem. Por exemplo, a Condição Faixa etária pode ser configurada da seguinte maneira:

- **Nome** — Faixa etária
- **Expressão da linguagem** — A idade do cliente está entre '{ageLow}' e '{ageHigh}'
- **Mapeamento da linguagem da regra** — Cliente (idade >= '{ageLow}' && idade <= '{ageHigh}')

Neste exemplo, {ageLow} e {ageHigh} são parâmetros.

Condições em regras lineares

Para uma regra linear, há um limite máximo 6 de colunas de dados de parâmetros (incluindo rótulos de texto estático). Então, por exemplo, se sua expressão for:

Defina os dados do cliente para: {parm1} e {parm2} e {parm3} e {parm4}

Nesse caso, {parm4}, sendo o 7º parâmetro, não será exibido. Reformule as suas condições para se ajustarem a estes limites.

Aviso

Ao configurar uma condição para um 'parâmetro operacional', é importante que você não coloque a variável entre aspas duplas ou simples, seja na expressão da linguagem ou no mapeamento da linguagem da regra. Consulte o exemplo [aqui](#).

Editor de enumerações

O editor de enumerações (Enums) permite criar e editar enumerações. Uma enumeração é um tipo de dados que consiste em um conjunto de valores nomeados que representam constantes. As enumerações são úteis para parâmetros que possuem um pequeno número de valores possíveis. Por exemplo, uma enumeração de **CustomerValue** pode ser ouro, prata ou bronze.

Cada enumeração contém os mesmos campos:

- **Nome** — Um nome para a enumeração.
- **Descrição** — Uma breve descrição da enumeração.
- **Valores** — Uma lista de valores possíveis. Clique em **Adicionar**, **Editar** e **Remover** para atualizar a lista.

O valor fornecido para a propriedade **Nome** corresponde ao valor da propriedade **Fato** que deve ser fornecida no Mecanismo de regras como parte de uma solicitação de avaliação de regras. O **Nome** diferencia entre maiúsculas e minúsculas.

O valor fornecido para a propriedade **Rótulo** corresponde ao que o autor de regras observará ao usar uma condição ou ação de regra que inclua um parâmetro do tipo **Valor de entrada**, com uma lista enumerada de valores.

Por exemplo, uma lista enumerada chamada **CustomerSegment** pode ser definida da seguinte maneira:

Nome	Rótulo
101	Bronze
102	Prata
103	Ouro

Neste exemplo, o **Nome** consiste apenas em dígitos, portanto, a diferenciação entre maiúsculas e minúsculas não é um problema. Por exemplo, se as propriedades **Nome** fossem bronze, prata e ouro, você deverá garantir que insere o valor exato Bronze como a propriedade **Fato** para que a regra seja avaliada conforme o esperado. Se você inserir bronze, a regra não retornará o resultado esperado.

Editor de modelos de fatos

Use o Editor de modelos de fatos para criar e editar Modelos de fatos para o modelo. O Editor de modelos de fatos é organizado como uma visualização padrão mestre/de detalhes. O componente mestre exibe fatos definidos e propriedades secundárias para cada fato.

Importante

No lançamento inicial do GRAT 9.0, não há suporte para Eventos no Modelo de fatos.

Fatos

Para criar um novo fato, clique no painel Fatos do Editor de modelos de fatos. Cada *fato* criado no editor contém os seguintes campos:

- **Nome** — Um nome para o fato.
- **Descrição** — Uma breve descrição do fato.
- **Sensível** — Determina se o fato é registrado quando o Mecanismo de regras avalia fatos. Se este campo estiver selecionado, o fato não será registrado. Se este campo estiver limpo, o fato será registrado.
- **Propriedades** — Uma lista de valores de propriedades. Clique em Adicionar, Editar e Remover para atualizar a lista. Clicar em Adicionar ou Editar abre uma caixa de diálogo. Essa caixa de diálogo contém os seguintes campos:
 - **Nome** — Um nome para a propriedade.
 - **Tipo** — O tipo de propriedade. Os tipos de propriedades são apresentados em uma lista suspensa. Cada tipo de propriedade possui um ícone para indicar o tipo. O ícone é exibido na lista de propriedades ao lado do nome da propriedade.
 - **Descrição** — Uma descrição da propriedade.
 - **Sensível** — Determina se a propriedade do fato é registrada quando o Mecanismo de regras avalia o fato. Se estiver selecionada, a propriedade fato não será registrada. Se estiver limpa, a propriedade fato será registrada.

Editor de funções

Use o Editor de funções para desenvolver métodos Java que podem ser chamados a partir de ações e condições da regra.

Cada função contém os mesmos campos:

Nome da função — Um nome para a função.

Descrição — Uma breve descrição da função.

Implementação — Java ou Groovy.

Exemplo

Esta função é usada para comparar datas. Ela seria configurada da seguinte forma:

```
Name: compareDates
Description: This function is required to compare dates.
Implementation:
import java.util.Date;
import java.text.SimpleDateFormat;

function int _GRS_compareDate(String a, String b) {
    // Compare two dates and returns:
    // -99 : invalid/bogus input
    // -1 : if a < b
    // 0 : if a = b
    // 1 : if a > b

    SimpleDateFormat dtFormat = new SimpleDateFormat("dd-MMM-yyyy");
    try {
        Date dt1= dtFormat.parse(a);
        Date dt2= dtFormat.parse(b);
        return dt1.compareTo(dt2);
    } catch (Exception e) {
        return -99;
    }
}
```

Para classes fornecidas pelo usuário, o arquivo .jar deve estar no CLASSPATH para GRAT e GRE.

Editor de parâmetros

O Editor de parâmetros permite criar parâmetros de regra que são usados opcionalmente em condições e ações da regra.

Importante

Na versão 9.0.0, não há suporte a mapeamento de parâmetros para modelos de fatos.

Cada parâmetro contém os mesmos campos na seção Detalhes:

Nome — Um nome para o parâmetro. **Descrição** — Uma breve descrição do parâmetro. **Tipo** — O tipo de parâmetro. Obtenha mais informações sobre os tipos nos tópicos a seguir:

Nomes dos parâmetros

O caractere sublinhado () nos nomes dos parâmetros tem um significado especial ao criar modelos de regra. É usado para especificar um índice do parâmetro nas circunstâncias em que a expressão da regra requer instâncias adicionais do parâmetro. O exemplo mais comum é uma definição de intervalo.

Por exemplo, suponha que você precise criar uma condição que precise verificar se a data de vencimento da tarefa está no intervalo de data1 até a data2 ou no intervalo de data3 até data4. É possível criar uma condição como:

```
Due is in "{dueDT1}" to "{dueDT2}" or in "{dueDT3}" to "{dueDT4}"
```

Mas isso exige a definição de 4 parâmetros com o tipo `InputDate` na seção `Parameters`. Essa abordagem pode se tornar ineficiente, especialmente se houver mais de uma ocorrência da condição/ação.

Uma solução melhor é usar a abordagem do caractere sublinhado e índice:

```
Due is in "{dueDT_1}" to "{dueDT_2}" or in "{dueDT_3}" to "{dueDT_4}"
```

Usando essa abordagem, é preciso especificar apenas um parâmetro, com nome `dueDT` e tipo `InputDate`.

Categorias de parâmetros

A seção **Configuração** contém informações que dependem do tipo de parâmetro. Quando um tipo é selecionado na lista suspensa, são exibidos diferentes campos relacionados a esse tipo.

Existem oito categorias principais de parâmetros:

- Cadeia de caracteres
- Inteiro
- Numérico
- Data
- Hora
- Booleano
- Calendário
- Configuration Server

Valores de entrada

Booleano, inteiro, numérico, cadeia de caracteres, data e hora são simplesmente parâmetros para os quais o autor de regras pode fornecer um valor com base no tipo de parâmetro definido. Esses parâmetros também podem ser restritos. Por exemplo, um valor inteiro pode ser limitado a estar dentro de um intervalo definido.

Padrão correspondente

Para os parâmetros Valor de entrada do tipo Cadeia de caracteres, é possível inserir um padrão correspondente que deve ser seguido. Digite uma expressão regular Javascript para definir o padrão correspondente. Por exemplo, um parâmetro de Código postal pode ter o padrão correspondente:

```
>^\d{5}$|^\d{5}-\d{4}$
```

que representa uma caixa postal de 5 dígitos. Um parâmetro de número de telefone pode ter o padrão correspondente:

```
^\((?\d{3})\)?[- ]?(?\d{3})[- ]?(?\d{4})$
```

que representa um número de telefone com 10 dígitos no formato (xxx)-xxx-xxxx.

Dicas de ferramentas personalizadas

Usar dica de ferramenta personalizada permite inserir textos úteis de dica de ferramenta ao definir todos os parâmetros do valor de entrada (exceto para o tipo de parâmetro booleano, que não precisa de uma dica de ferramenta). Se você marcar **Usar dica de ferramenta personalizada**, o texto digitado no campo dica de ferramenta será exibido no GRAT quando esse parâmetro for usado em uma condição ou ação de regra. Se você não marcar **Usar dica de ferramenta personalizada**, o GRAT exibirá uma dica de ferramenta gerada automaticamente; por exemplo, {age} é um número inteiro entre 1 e 99.

O padrão regex suportado deve estar em conformidade com o mecanismo Javascript do navegador e pode variar um pouco, dependendo da versão do navegador.

Calendários

Os parâmetros do calendário indicam ao GRAT que ele deve exibir uma lista suspensa de calendários comerciais associados ao pacote de regras que está sendo editado. O autor de regras pode escolher um dos calendários:

Exemplo: Os parâmetros do calendário podem ser usados em uma regra para atribuir um calendário dinamicamente da seguinte maneira:

```
Assign business calendar "{businessCalendar}"
```

Ao definir um parâmetro Calendário, o designer do modelo precisa fornecer apenas o nome do parâmetro e selecionar um tipo de Calendário. Nenhuma outra configuração é necessária.

Configuration Server

Os parâmetros do Configuration Server oferecem ao autor de regras a capacidade de escolher um único valor em uma lista suspensa de valores. Por exemplo, um parâmetro do Configuration Server pode ser configurado para obter uma lista de Grupos de agentes do banco de dados do Configuration Server. A lista é preenchida a partir do Configuration Server. Os parâmetros do Configuration Server exigem a seleção do Tipo de objeto:

- Agente
- Grupos de agentes
- Habilidades do agente
- Atributo comercial
- Contexto comercial
- Extensão
- Ponto de roteamento externo
- Fila de interações
- Tipo de mídia
- Local
- Grupos de lugares
- Ponto de rota
- Alternar
- Servidor T
- Pontos de roteamento virtual

Ao selecionar **Atributo comercial** você será solicitado a selecionar o nome do Atributo comercial em uma lista definida no Gerenciador de configurações.

Ao selecionar **Contexto comercial** você será solicitado a inserir o nível do Contexto comercial

relevante a este parâmetro. Aqui, Contexto comercial refere-se ao nível da hierarquia na pasta **Estrutura comercial** no Configuration Server.

Banco de dados

Os parâmetros do banco de dados oferecem ao autor de regras a capacidade de escolher um único valor em uma lista suspensa de valores. Por exemplo, um parâmetro Banco de dados pode ser configurado para obter uma lista de Tipos de pedido de um banco de dados. A lista é preenchida por uma consulta ao banco de dados. Os parâmetros do banco de dados exigem que o Nome de perfil (o nome do objeto Script do Configuration Server que contém as suas informações de conexão do banco de dados), o Tipo de consulta (valor único ou lista, dependendo do que você deseja que apareça no GRAT) e a Consulta SQL sejam executados.

Observação: A lista de valores é obtida no momento em que o autor de regras realiza o login no GRAT. Se algum valor for atualizado do sistema externo após o usuário realizar o login no GRAT, o usuário deverá clicar no botão Logoff e, em seguida, realizar login novamente para encontrar as alterações.

Exemplo

Para usar um parâmetro Banco de dados, um perfil de parâmetro deve ter sido configurado anteriormente para o Locatário no Configuration Server. Este é um objeto de Script que especifica o driver JDBC, bem como a URL do banco de dados, o nome de usuário e a senha necessários para executar a consulta. Consulte o Guia de implantação do Genesys Rules System para obter mais informações sobre a configuração desses perfis. O nome deste objeto de Script é usado como o nome do perfil para o parâmetro do banco de dados.

Para obter valores do banco de dados, uma instrução SQL Select válida deve ser especificada. Por exemplo, para obter todos os valores de uma coluna, use uma instrução Select do seguinte formato:

```
SELECT column_name FROM table_name
```

Para parâmetros dinâmicos de Banco de dados, é possível configurar o parâmetro de forma que um nome (um valor interno incluído em uma solicitação de avaliação de regra) e um rótulo (as informações exibidas para um autor de regras ao criar uma regra que usa esse parâmetro) possam ser recuperados a partir de duas colunas diferentes do banco de dados.

Propriedades de parâmetro do perfil do banco de dados

Propriedade	Obrigatório/opcional	Descrição
driver	Obrigatório	O nome do driver JDBC a ser usado. Por exemplo, com.mysql.jdbc.Driver
url	Obrigatório	A URL do banco de dados no formato correto para o driver JDBC a ser usado.
username	Obrigatório	Um nome de usuário válido para conectar-se ao banco de dados.
password	Obrigatório	A senha necessária para o usuário conectar-se ao banco de dados.

Propriedade	Obrigatório/opcional	Descrição
initSize	Opcional	O tamanho inicial do conjunto de conexões. O padrão é 5.
maxSize	Opcional	O tamanho máximo do conjunto de conexões. O padrão é 30.
waitTime	Opcional	O tempo máximo (em milissegundos) para aguardar a obtenção de uma conexão. O padrão é 5000.

Em geral, os valores opcionais não precisam ser definidos ou alterados.

No desenvolvimento de modelos, é possível configurar apenas os parâmetros do banco de dados com uma instrução SQL SELECT. Qualquer outro tipo de instrução falhará quando configurada.

Operacional

Os parâmetros operacionais são criados pelos usuários através da Genesys Administrator Extension e, quando implantados, são armazenados como opções de objetos de Transação do tipo List no banco de dados do Genesys Configuration Server. No momento da execução da regra, quando o Mecanismo de regras avalia uma regra que contém um parâmetro Operacional, ele obtém o valor atual da opção de objeto de Transação associada do Configuration Server. O desenvolvedor da regra determina a partir de qual objeto de Transação e qual opção desse objeto, o valor do parâmetro Operacional é buscado, e o autor de regras usa esse parâmetro como parte de uma condição ou ação.

Exemplo Um parâmetro operacional chamado **waitTimeThreshold** pode ser definido. Se o autor de uma chamada estiver aguardando mais que esse limite por um agente disponível, alguma outra ação poderá ser realizada.

Em vez de especificar um valor para o limite na regra como o seguinte:

When

Caller's wait time is greater than 30 seconds

Then

Offer a callback

o autor de regras pode especificar:

When

Caller's wait time is greater than {waitTimeThreshold}

Then

Offer a callback

O valor de {waitTimeThreshold} pode ser alterado a qualquer momento por um usuário usando a Genesys Administrator Extension e terá efeito imediato sem a necessidade de modificar e reimplantar um pacote de regras.

Por exemplo, use a seguinte condição ao definir o mapeamento:

```
Queue(waitTime > {waitTimeThreshold} )
```

Para configurar um parâmetro Operacional, são necessários dois IDs:

- O **ID da lista**, que corresponde ao nome do objeto de Transação no qual o parâmetro Operacional está armazenado
- O **ID do parâmetro**, que corresponde ao nome de uma opção desse objeto de Transação.

O valor da opção contém o valor real do parâmetro Operacional que é recuperado pelo Mecanismo de regras quando a regra é avaliada. Os parâmetros operacionais são sempre armazenados como objetos de Transação do tipo List, mas a configuração precisa das opções nesse objeto de Lista varia dependendo de como o parâmetro Operacional foi configurado.

Aviso

Ao configurar um 'parâmetro operacional', é importante que você *não* coloque a variável entre aspas duplas ou simples, seja na expressão da linguagem ou no mapeamento da linguagem da regra. Por exemplo, uma condição que use um parâmetro operacional {opParmEwtThreshold} deve ser configurada da seguinte maneira:

- **Expressão da linguagem** — 0 tempo estimado de espera é maior que {opParmEwtThreshold}
- **Mapeamento da linguagem da regra** — CallInfo (ewt > Integer.parseInt({opParmEwtThreshold}))

Observe que não há aspas simples ou duplas em torno de {opParmEwtThreshold}

Para obter exemplos adicionais, consulte o modelo de amostra 'Parâmetros operacionais' e o pacote de regras enviado com o GRAT (na pasta **exemplos**).

Consulte a Ajuda da Genesys Administrator Extension para obter informações gerais sobre parâmetros Operacionais.

Selecionar enumeração

Os parâmetros Selecionar enumeração estão vinculados a uma Enumeração. Isso fornece ao autor de regras uma lista específica da qual selecionar.

Serviço Web

Os parâmetros do Serviço Web oferecem ao autor de regras a capacidade de escolher um único valor em uma lista suspensa de valores. Por exemplo, um parâmetro Serviço Web pode ser configurado para obter uma lista de símbolos de ação de um serviço Web externo. A lista é preenchida por uma consulta de Serviço Web. Os parâmetros de Serviço Web exigem que o Nome do perfil (o nome do objeto Script do Configuration Server que contém suas informações de conexão do serviço Web), o

Tipo de consulta (valor único ou lista) e a Consulta de XPath sejam executados. Além disso, os parâmetros de serviço Web exigem que algumas configurações de protocolo sejam inseridas, especificamente o método HTTP, o caminho e o corpo da mensagem.

Observação: A lista de valores é obtida no momento em que o autor de regras realiza o login no GRAT. Se algum valor for atualizado do sistema externo após o usuário realizar o login no GRAT, o usuário deverá clicar no botão Logoff e, em seguida, realizar login novamente para encontrar as alterações.

Exemplo

Semelhante a um parâmetro Banco de dados, um perfil de parâmetro deve também ter sido criado anteriormente. Esse perfil conterá informações como o endereço do servidor (host e porta), o caminho para o serviço e quaisquer outras configurações HTTP necessárias. Consulte o Guia de implantação do Genesys Rules System para obter mais informações sobre a configuração desses perfis.

Para obter valores do serviço, uma mensagem válida deve ser especificada para o serviço. Essa mensagem deve ser constante. Em outras palavras, nenhuma substituição de variável ocorrerá.

Observação: Nenhuma mensagem pode ser enviada para solicitações HTTP GET. Todas as informações na solicitação são fornecidas através da cadeia de caracteres de consulta e/ou cabeçalhos.

Por exemplo, para obter a previsão do tempo para São Francisco, a seguinte solicitação pode ser feita ao serviço REST Weather Underground:

<http://api.wunderground.com/auto/wui/geo/ForecastXML/index.xml?query=94129>

Contudo, essa é a solicitação completa. O host (api.wunderground.com) e o caminho base (/auto/wui/geo/ForecastXML/) devem ser especificados no perfil.

Para definir um parâmetro para realizar essa solicitação, o nome do perfil deve fazer referência às informações corretas descritas acima. Além disso, as Configurações do protocolo devem especificar GET como o método, juntamente com `index.xml?query=94129` como o caminho. Nenhuma mensagem é necessária para esta solicitação.

Para obter valores do resultado, uma expressão Xpath válida deve ser especificada. O Serviço Web deve retornar resultados em XML ou JSON. Consulte as especificações Xpath para obter mais informações sobre expressões Xpath.

Por exemplo, para obter as altas da previsão da solicitação descrita anteriormente, a expressão XPath a seguir pode ser usada:

```
//high/fahrenheit/text()
```

No Configuration Server, os scripts de Serviço Web devem ter uma seção chamada webservice. A tabela abaixo lista as propriedades que podem ser especificadas para os parâmetros de serviço Web.

Propriedades de parâmetro do perfil de serviço Web

Propriedade	Obrigatório/opcional	Descrição
Host	Obrigatório	O host para o serviço.

Propriedade	Obrigatório/opcional	Descrição
base-path	Obrigatório	O caminho base para acessar o serviço.
protocol	Opcional	O padrão é http.
port	Opcional	O padrão é 80.
headers	Opcional	Quaisquer cabeçalhos HTTP personalizados necessários para o serviço.
parameters	Opcional	Quaisquer configurações HTTP personalizadas necessárias para ajustar a conexão.

Em geral, os valores de parâmetros não precisam ser definidos ou alterados. Cabeçalhos e parâmetros são listas no seguinte formato:

```
key:value[, key:value]
```

Você não pode especificar cabeçalhos ou parâmetros que contêmham ',' no valor. Caso esteja enviando uma mensagem para o serviço, será esperado que Tipo de conteúdo seja especificado no cabeçalho, pois define a interação geral da mensagem com o servidor. Um conjunto de caracteres opcional pode ser incluído. Por exemplo, Content-Type:applicaton/json;charset=UTF-8.

Importante

No Desenvolvimento de modelos, é necessário definir completamente a mensagem a ser enviada e ela deve ser constante. Nenhuma substituição de variável é realizada. A Consulta de XPath é usada para extrair valores da resposta do servidor. A resposta deve estar em XML ou JSON, caso contrário, isso não funcionará. Uma consulta de XPath válida deve ser especificada para a resposta. Isso depende inteiramente do serviço com o qual você faz interface.

Importante

A mensagem é enviada para o servidor somente uma vez por sessão. Isso é feito por razões de desempenho e porque espera-se que os valores na resposta sejam relativamente constantes.

No Desenvolvimento de modelos, o caminho para o parâmetro é adicionado ao base_path no script. Por exemplo, se o script conter:

```
host = api.wunderground.com  
base_path = /auto/wui/geo/ForecastXML/
```

e o GRDT especifica:

```
query type = List
```

```
XPath Query = //high/fahrenheit/text()  
HTTP Method = GET  
path = index.xml?query=66062  
message (not set)
```

então a mensagem enviada é:

```
GET /auto/wui/geo/ForecastXML/index.xml?query=66062 HTTP/1.1
```

Isso retornará as máximas da semana em Fahrenheit:

```
81  
77  
81  
81  
83  
85
```

Workforce Management

Os parâmetros Workforce Management (WFM) permitem que o autor de regras selecione um valor em uma lista suspensa de atividades (um objeto de banco de dados WFM que representa tarefas do centro de contato em que os agentes podem ser envolvidos) e atividades multissite (uma coleção de atividades executadas em diversos locais físicos) que é recuperada dinamicamente do Servidor Genesys Workforce Management. Os parâmetros Workforce Management exigem o perfil WFM (o objeto de Script do Configuration Server do tipo Coleta de dados).

Importante

A lista de valores é obtida no momento em que o autor de regras realiza o login no GRAT. Se algum valor for atualizado do sistema externo após o usuário realizar o login no GRAT, o usuário deverá clicar no botão Logoff e, em seguida, realizar login novamente para encontrar as alterações.

Exemplo

Uma atividade é o principal objeto de planejamento usado na criação de previsões e agendas. Uma atividade pode ser associada a um objeto de site WFM individual ou podem ser criadas atividades multissite ao nível da unidade de negócios WFM, que agrega informações de várias atividades 'secundárias' em vários Sites WFM. Portanto, ao fornecer para o autor de regras uma lista de atividades WFM que são buscadas dinamicamente no WFM Server, o nome da atividade WFM ou atividade multissite é prefixado com o nome do Site WFM ou da Unidade de negócios WFM, respectivamente.

Por exemplo, se a configuração do WFM tiver a seguinte estrutura:

Unidade de negócios com o nome 'ACME'

- Site com o nome 'São Francisco'
- Atividade com o nome 'Disputas'
- Atividade com o nome 'Consultas de cobrança'

- Site com o nome 'Chicago'.
- Atividade com o nome 'Disputas'
- Atividade com o nome 'Alterações de endereço'
- Atividade multissite com o nome 'Faturamento' (composta por 'Consultas de faturamento' do site de São Francisco e 'Alterações de endereço' do site de Chicago)
- Atividade multissite com o nome 'Disputas' (composta por 'Disputas' do site de São Francisco e 'Disputas' do site de Chicago)

O autor de regras perceberá os seguintes itens na lista suspensa ao usar a ação de regra Atribuir atividade WFM em uma regra:

U.N. ACME: Cobrança U.N. ACME: Disputas Site de Chicago: Alterações de endereço Site de Chicago: Disputas Site de São Francisco: Consultas de cobrança Site de São Francisco: Disputas

Importante

Os nomes das Unidades de negócios e sites são prefixados com 'U.N.' e 'Site', respectivamente, para ajudar o autor de regras a entender o contexto.

No Configuration Server, os scripts do Workforce Management devem ter uma seção chamada wfm. A tabela abaixo lista as propriedades que você pode especificar para os parâmetros Workforce Management.

Propriedades do parâmetro do perfil Workforce Management

Propriedade	Obrigatório/opcional	Descrição
wfmCfgServerAppName	Obrigatório	Nome do aplicativo do Configuration Server para o servidor WFM.
wfmCfgServerUserName	Obrigatório	Nome de usuário do Configuration Server.
wfmCfgServerPassword	Obrigatório	Senha do Configuration Server.
wfmServerUrl	Obrigatório	URL do WFM Server.

Ao configurar um novo parâmetro do tipo 'Workforce Management' em Desenvolvimento de modelos, basta nomear o parâmetro e escolher o perfil WFM (objeto de script recém-criado) na lista suspensa. Quando o autor estiver usando esse parâmetro, o GRAT buscará a lista atual de Atividades WFM no WFM Server e as exibirá para o autor de regras.

Pesquisar

O GRAT inclui um recurso de Pesquisa. Você pode pesquisar por regras existentes para localizá-las com mais facilidade para edição.

Alguns dos campos que podem ser pesquisados são:

- ID da regra
- Nome da regra (o nome completo, 'começa com' ou 'inclui')
- Última modificação por: o nome do usuário da última pessoa que atualizou a regra
- Intervalo da data de criação (qualquer regra criada entre x e y)
- Calendário Comercial (nome do calendário)
- Regras com instantâneo pendente
- Regras que contêm um parâmetro específico em uma condição de regra.
- Regras que contêm um parâmetro específico em uma ação de regra.

Importante

Depois de serem apresentados os resultados da busca, você poderá clicar nas regras para ver o conteúdo, modificar as regras ou excluí-las da tela de pesquisa. Também é possível clicar no ícone de navegação para localizar o contexto no qual a regra foi originalmente definida.