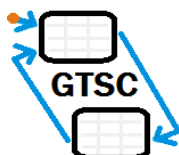




MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS



GTSC: Geração Automática de Casos de Teste Baseada em Statecharts

Manual do Usuário

INPE – Instituto Nacional de Pesquisas Espaciais
São José dos Campos
Maio/2012

1 Descrição do Software

Este documento apresenta o manual de instalação e utilização da versão 2.0 do ambiente para Geração Automática de Casos de Teste Baseada em Statecharts (GTSC).

O ambiente GTSC permite a geração automática de casos de teste baseando-se em modelagem comportamental em Statecharts e em MEF e tem sido extensivamente usado em estudos de caso que envolvem software embarcado em instrumentos que estão a bordo de satélites científicos e balões estratosféricos.

O Ambiente integra uma série de ferramentas, de modo que o projetista de teste possa ter um ambiente de modelagem comportamental de software em Statecharts e em MEF para gerar casos de teste. A Figura 1 apresenta a arquitetura do GTSC.

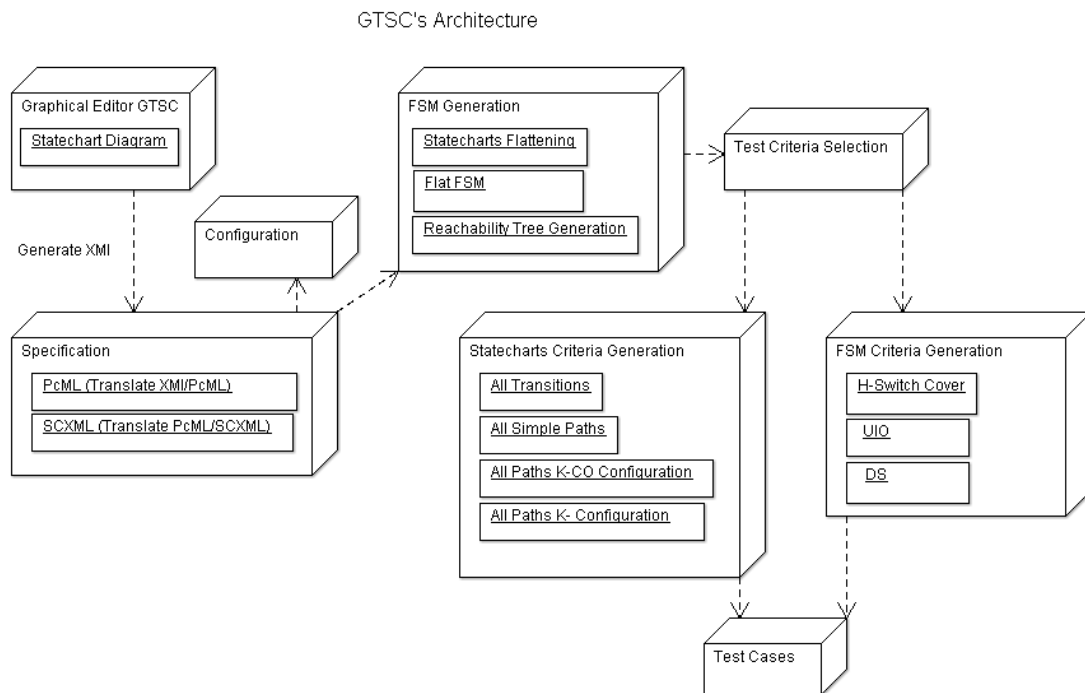


Figura 1 – Arquitetura do GTSC.

2 Instalação

2.1 Requisitos de Software

Os requisitos de software para a instalação e uso da versão 2.0 do GTSC estão descritos a seguir:

- ✓ Sistema Operacional Windows XP, Vista e 7;
- ✓ Versão mais recente do Active Perl instalado no diretório raiz da unidade C (C:\). Informações de instalação do Active Perl podem ser encontradas no Apêndice A;
- ✓ No mínimo Java Runtime Environment (JRE) 1.7 instalado;
- ✓ Arquivo de instalação do GTSC. Nesse caso, existem duas opções:
 - GTSCv2-x32.exe para plataforma de 32 bits;
 - GTSCv2-x64.exe para plataforma de 64 bits.

Obs.: O Active Perl e o JRE devem estar previamente instalados para que a instalação do GTSC possa ser iniciada na máquina.

2.2 Procedimentos

Este tópico descreve os procedimentos de instalação do GTSC com o objetivo de deixá-lo operacional.

1. Faça o *download* do instalador do GTSC, dependendo da plataforma (32 ou 64 bits).
2. Uma vez obtido o arquivo, execute-o para iniciar a instalação. Será apresentada a tela mostrada abaixo, clique em **Install** para prosseguir com a instalação.

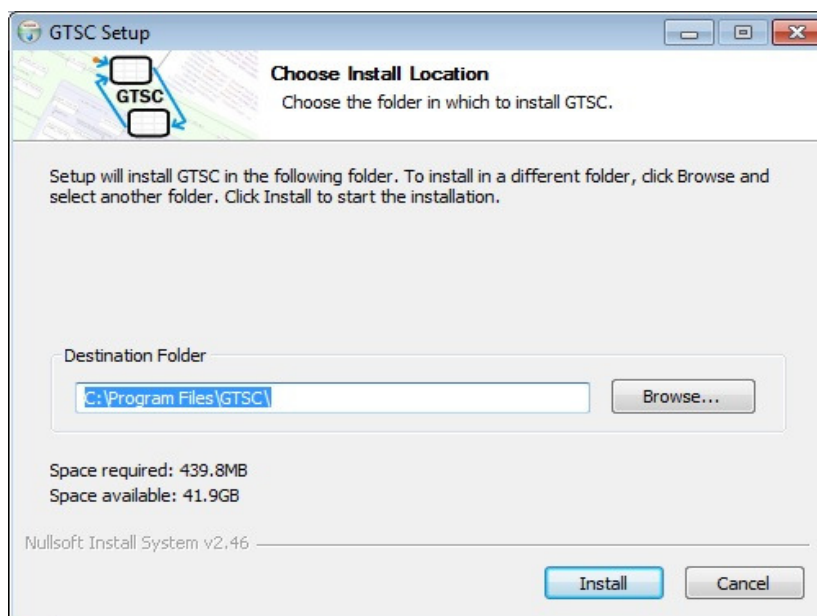


Figura 2 – Tela inicial de instalação do GTSC.

3. Uma vez finalizado o processo de instalação basta-se clicar em **Finish**.

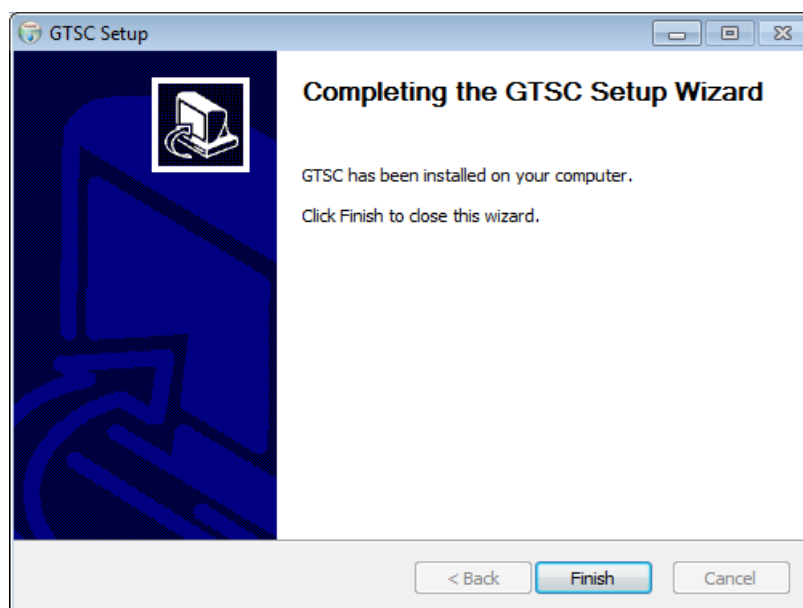


Figura 3 – Finalização de instalação do GTSC.

4. Será então solicitado que a máquina seja reiniciada para completar a instalação. Salve seu trabalho e clique em **Sim**.

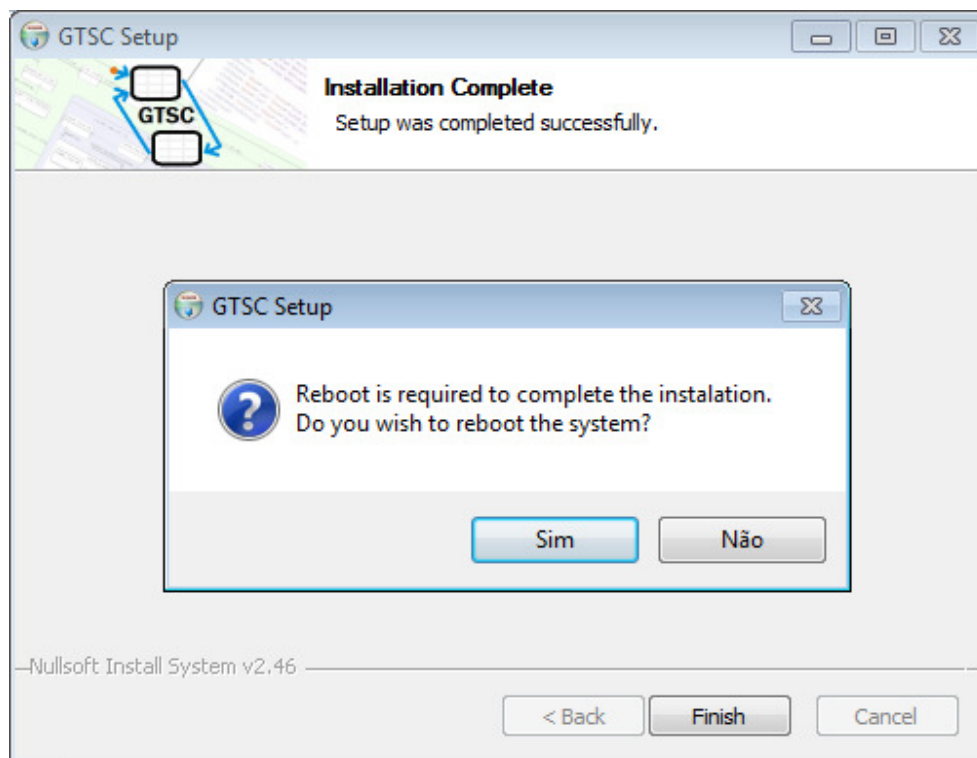


Figura 4 – Tela de confirmação de reinicialização.

5. Para que o sistema fique operacional é necessário instalar os seguintes pacotes no Perl: libxml-perl, XML-DOM, XML-DTDParse, XML-Schema e o XML-Validate. Para permitir esta instalação basta clicar em **OK** na seguinte tela que irá surgir após o computador ser reinicializado.

Obs.: É necessário conexão com a internet para que a instalação dos pacotes ocorra com sucesso, caso ocorra algum erro, basta executar o arquivo **PerlModules.vbs**, presente na pasta **prerequisites** dentro do diretório de instalação. O processo de instalação dos pacotes poderá levar alguns minutos, dependerá da velocidade de conexão.

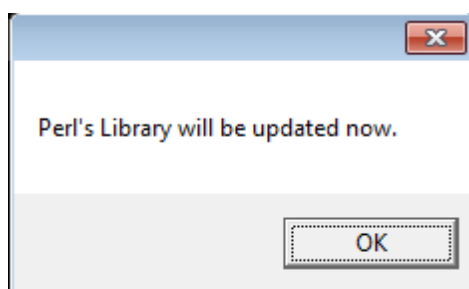


Figura 5 – Instalação dos pacotes do Active Perl.

6. Após clicar em **OK**, as bibliotecas necessárias do Perl serão instaladas. Telas de progresso serão mostradas ao usuário como as apresentadas nas figuras abaixo.

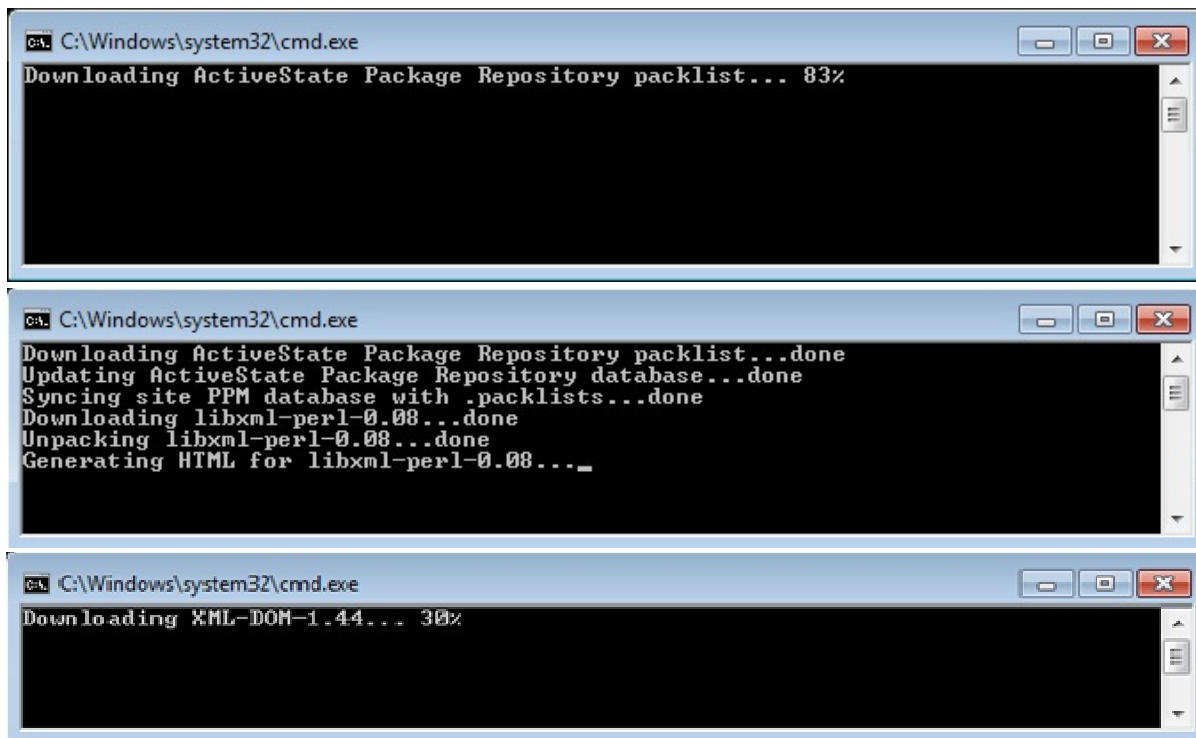


Figura 6 – Telas de progresso apresentadas na instalação dos pacotes do Active Perl.

3 Utilizando o ambiente

3.1 Interface do GTSC

A interface gráfica do GTSC foi desenvolvida com base no ARGOUML, ferramenta de modelagem UML. A janela principal, ilustrada na Figura 7, possui uma barra de menu (número 1), uma barra de ferramentas (número 2), e quatro painéis principais: *Explorer* (número 3), *Edição* (número 4), *Tarefa* (número 6) e *Detalhes* (número 7). O painel de *Edição* é composto pelas abas: *Diagram* e *GTSC*, na aba *Diagram* encontra-se área de desenho (número 5), onde é possível modelar sistemas em Statecharts e obter sua especificação em PcML apresentada na aba *GTSC*.

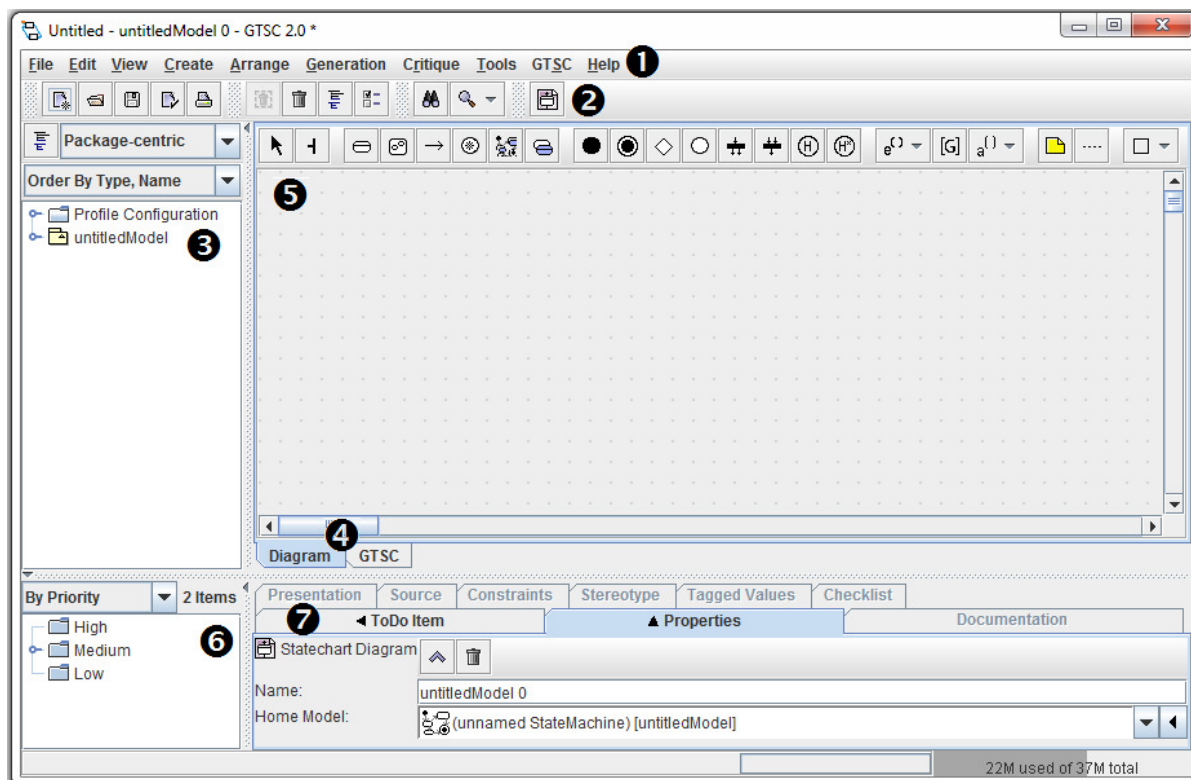


Figura 7 – Tela principal do ambiente GTSC.

Na Figura 8, é apresentado um exemplo de modelagem de um sistema usando o GTSC.

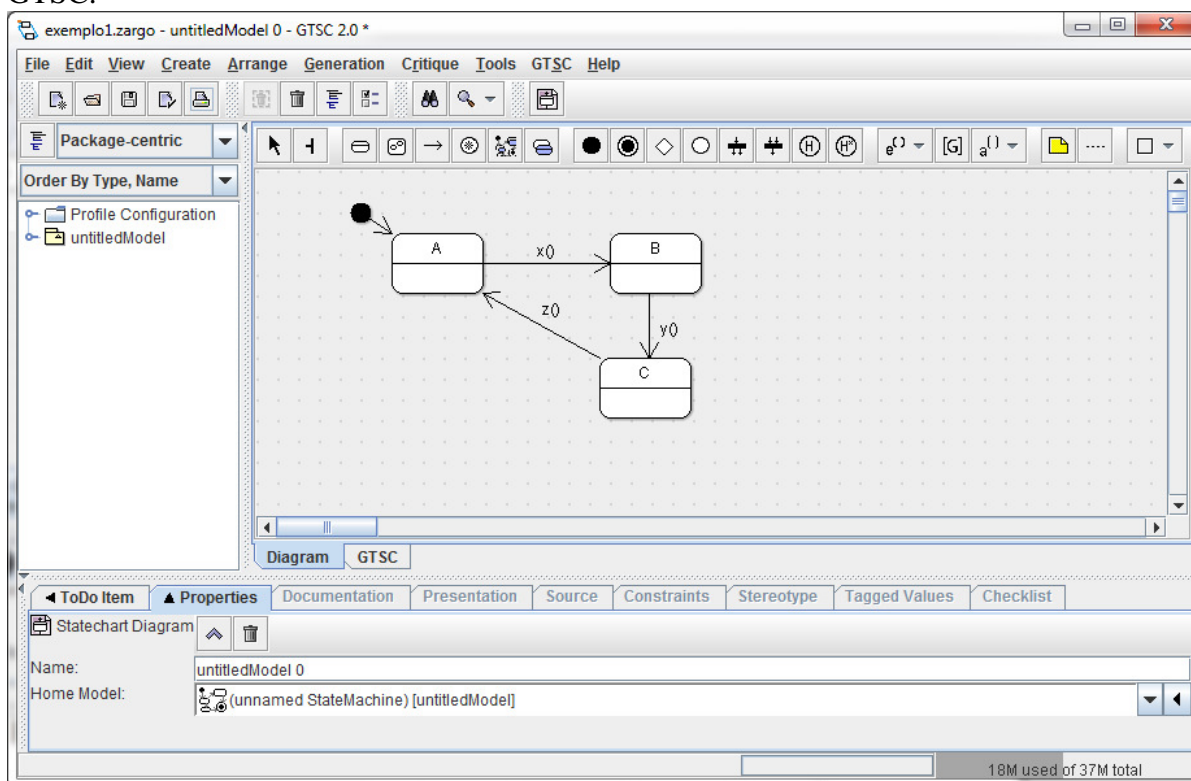


Figura 8 – Exemplo de modelagem.

Ao selecionar um elemento do modelo suas propriedades são mostradas na aba *Properties* do painel Detalhes. O conteúdo da aba varia de acordo com o tipo de elemento do projeto. A aba propriedade é definida para todos os elementos do esquema lógico e ela possibilita a criação/alteração/exclusão de atributos, operações, relacionamentos, entre outros. Na Figura 9 são apresentadas as propriedades da transição **x**, do exemplo anterior.

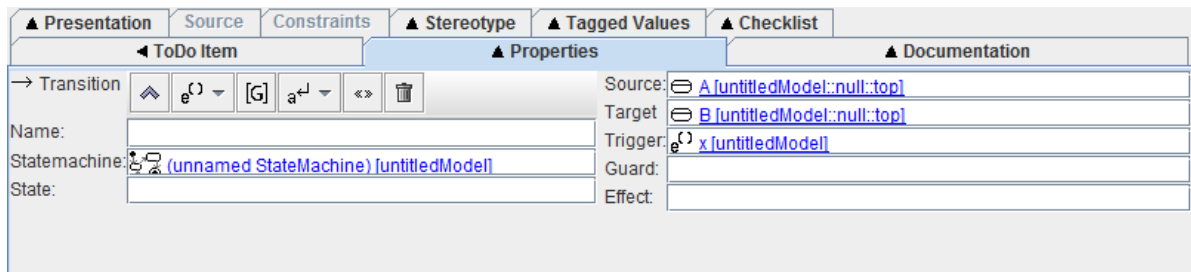


Figura 9 – Propriedades da transição **x**.

Clicando-se na aba “GTSC”, obtém-se a especificação PcML do modelo.

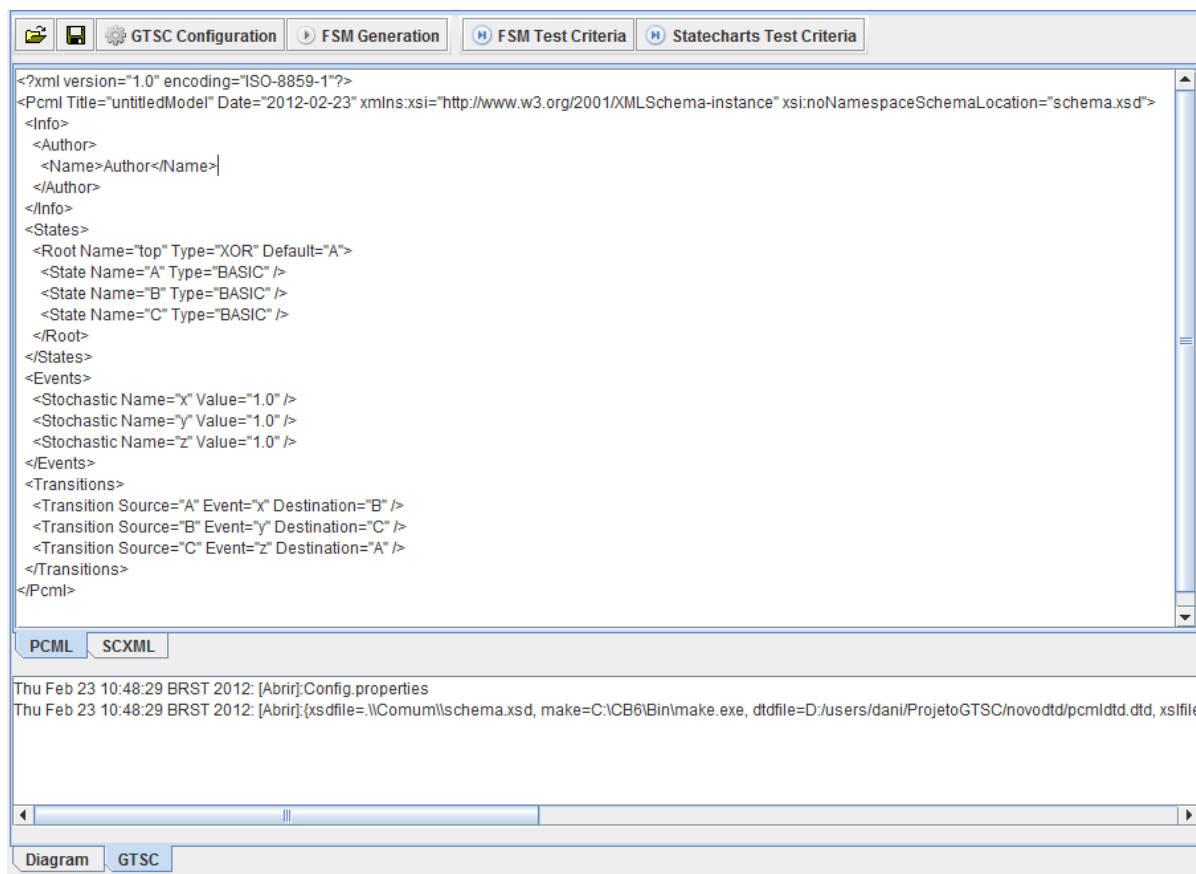


Figura 10 – Especificação PcML do modelo.

3.2 Configurações

As configurações padrões do GTSC podem ser alteradas a partir do botão “GTSC Configuration”, onde deve-se informar a localização dos arquivos:

Aba Model (Figura 11):

- **PcML:** localização do arquivo PcML. Se o usuário quiser usar um arquivo PcML já existente, sem a necessidade de usar a interface gráfica do GTSC, então ele pode colocar o nome e diretório do arquivo PcML existente nesse campo. Caso o usuário deseje usar a interface gráfica do GTSC para implementar o modelo em Statecharts que depois é transformado para uma especificação PcML, então esse nome deverá ser fixo em **pcml.xml**;
- **Perl(exe):** localização do executável Perl. Normalmente presente em C:\Perl\bin\perl.exe para instalações 32 bits e C:\Perl64\bin\perl.exe para instalações 64 bits. Esse campo não deverá ser alterado pelo usuário.

Obs.: Além do campo PcML existem os campos **XSD** e **PerlScript** que não devem ser alterados pelo usuário.

Aba Compiler (Figura 12):

Os campos presentes na aba **Compiler** são: Project name, Make, Performcharts. Estes campos não devem ser alterados pelo usuário.

Aba Generation (Figura 13):

- **FSM:** nome da MEF flat que será gerada (.xml). A MEF flat será gerada dentro do diretório da raiz da instalação ou em diretório a critério do usuário.

Após informar a localização de todos os arquivos, basta clicar no botão **OK** e as informações serão gravadas.

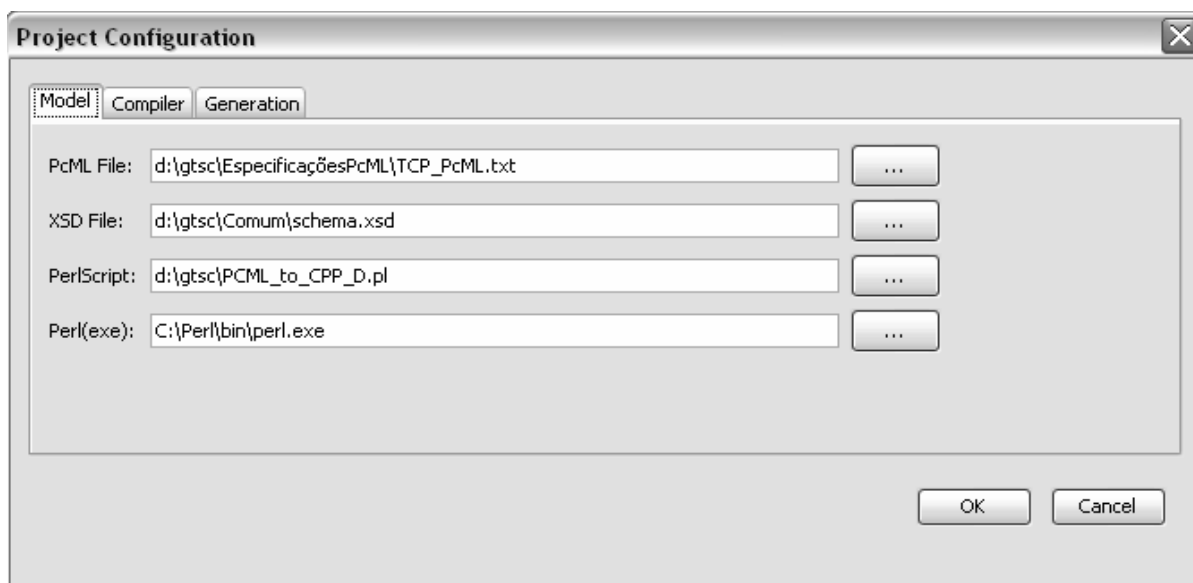


Figura 11 - Project Configuration – Model.

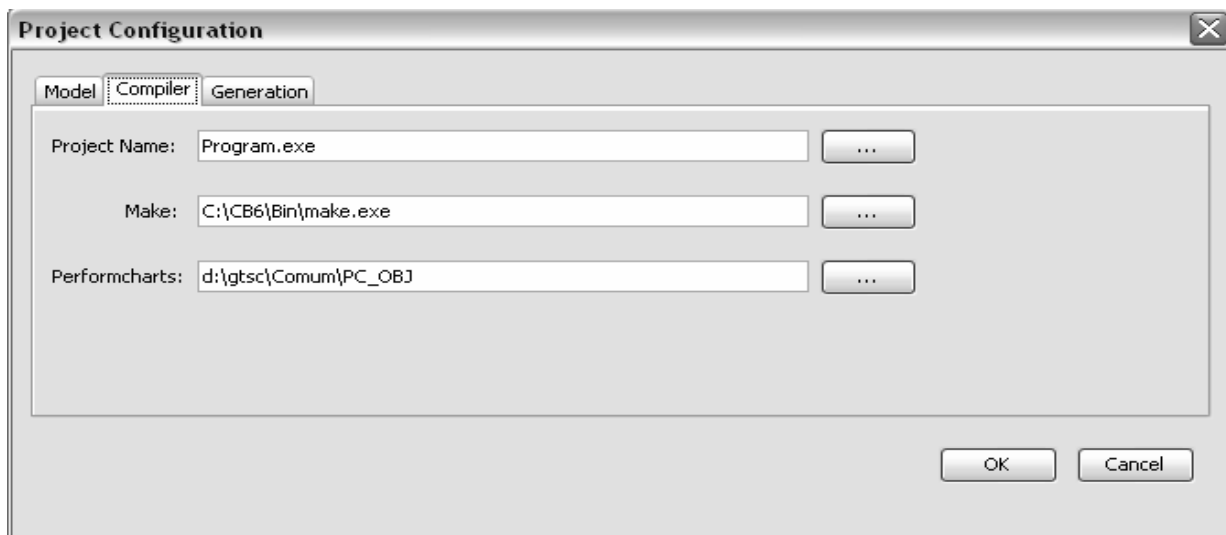


Figura 12 - Project Configuration – Compiler.

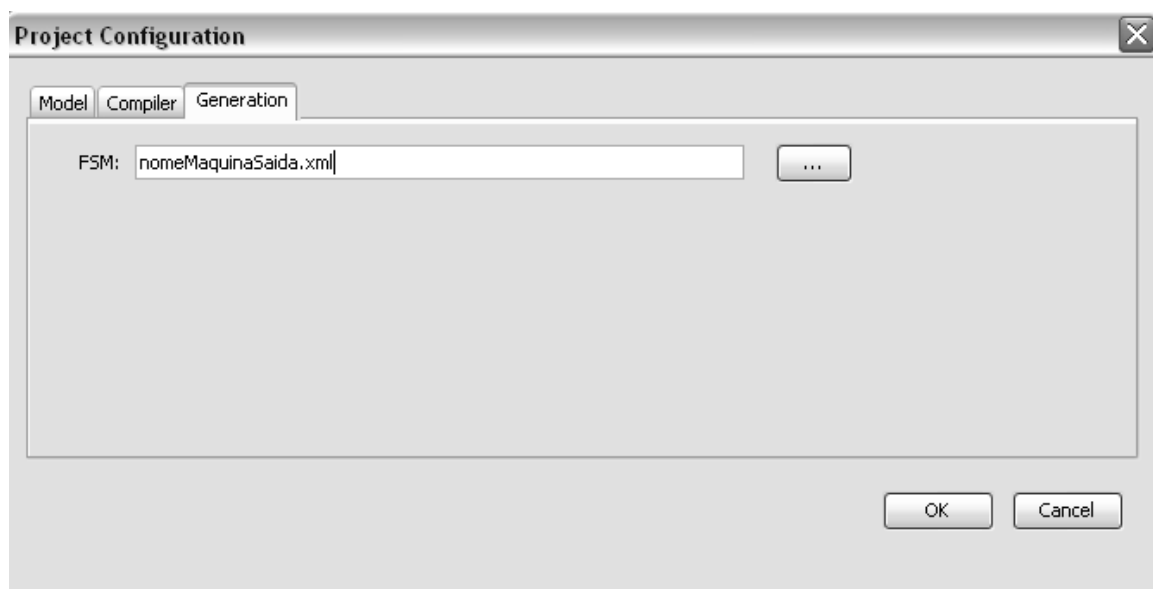


Figura 13 - Project Configuration.

Após finalizar as configurações, é possível modelar um sistema e gerar a MEF flat em XML.

3.3 Exemplos

3.3.1 Exemplo 1

Na Figura 14 é apresentado um exemplo de sistema modelado em *Statecharts* usando o GTSC.

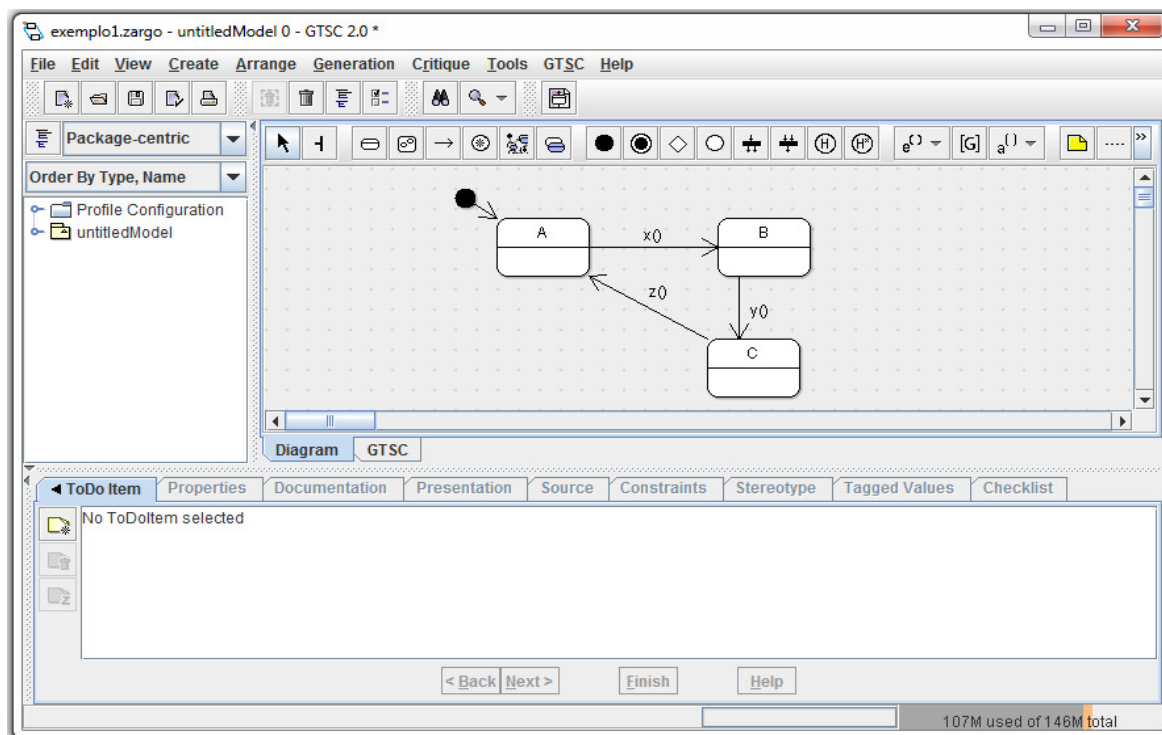


Figura 14 – Exemplo 1.

Clicando-se na aba “GTSC” obtém-se a especificação PcML do sistema. A geração da máquina é feita através do botão “*FSM Generation*” mostrado na Figura 15. A máquina será gerada dentro do diretório selecionado no campo “FSM” na aba “*Generation*” do “*Project Configuration*”.

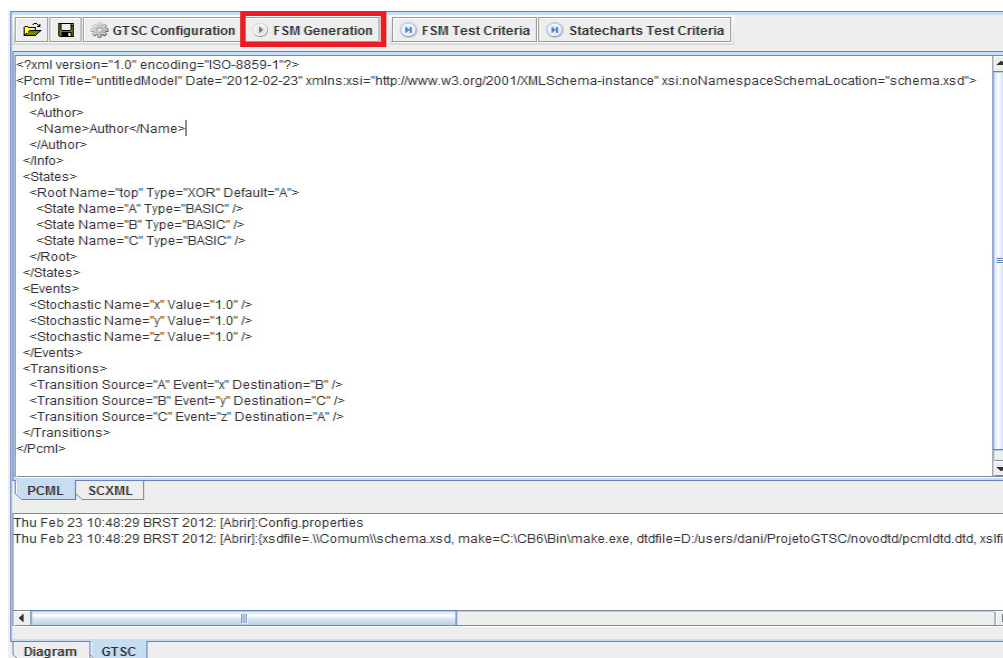


Figura 15 - FSM Generation.

Uma vez obtida a máquina, basta escolher se os casos de teste serão gerados com base em métodos para Máquinas de Estado Finito (*FSM Test Criteria*, Figura 14) ou critérios

baseados em *Statecharts* (*Statecharts Test Criteria*, Figura 15), escolher qual a máquina XML (por padrão a máquina já é escolhida com base no “*Project Configuration*” mas pode ser alterada), o método/critério e clicar no botão *Execute*. Ao clicar no botão *Execute*, são gerados os casos de teste.

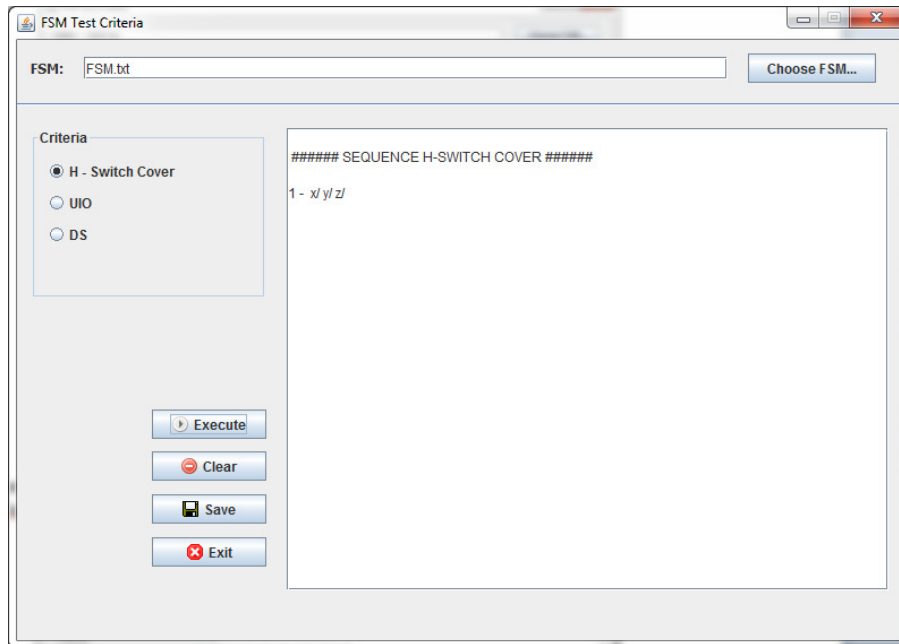


Figura 16 - Geração dos casos FSM método H - *Switch Cover*.

O resultado obtido são os casos de testes para o método H - *Switch Cover*. Neste exemplo apenas um caso de teste foi gerado.

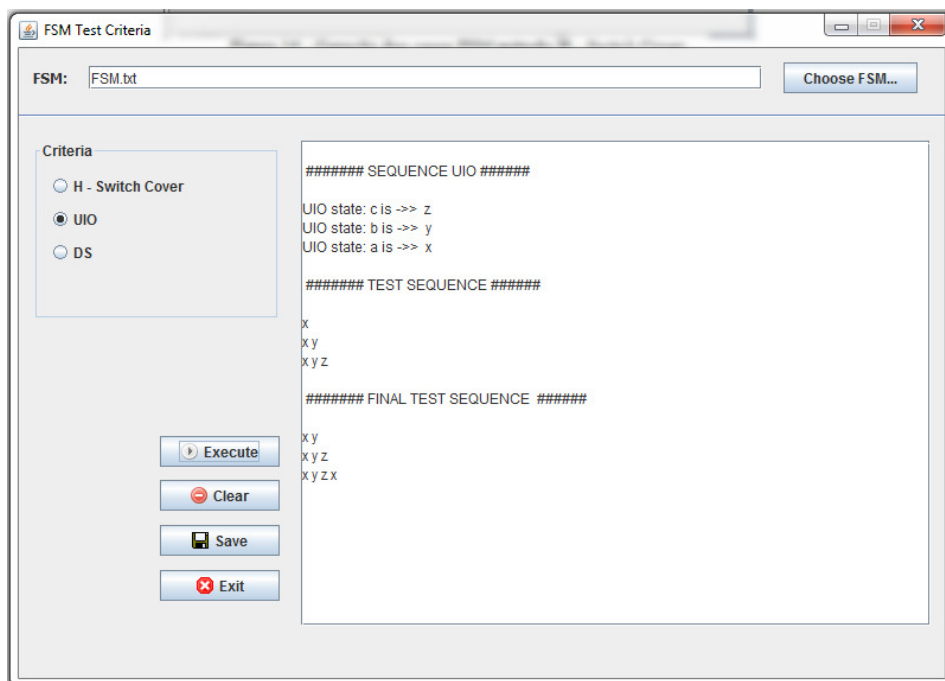


Figura 17 - Geração dos casos FSM método UIO.

Já para o método UIO são apresentadas sequências UIO, sequência de teste e a sequência final de teste, como pôde ser observado.

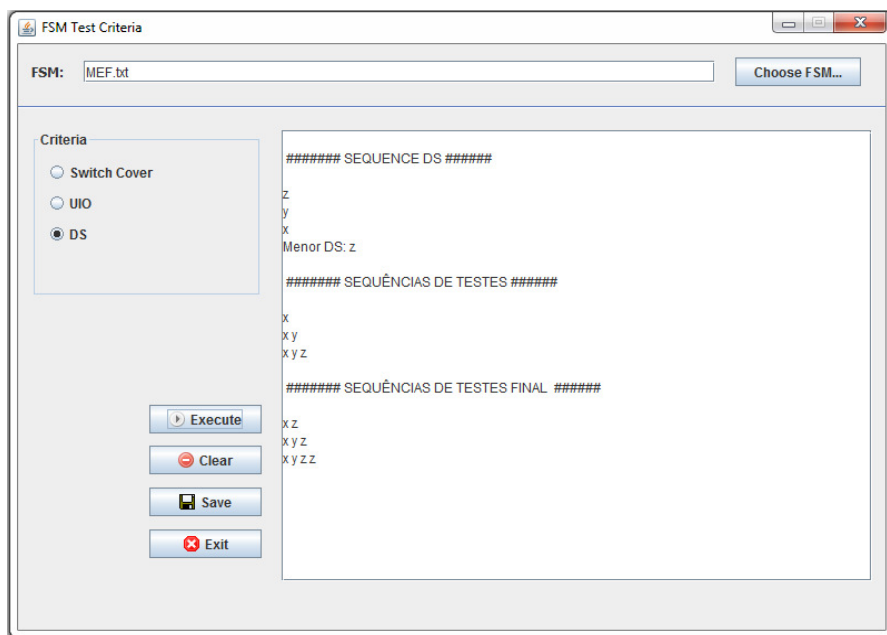


Figura 18 - Geração dos casos – FSM.

Semelhante ao método UIO são apresentadas três sequências a DS, a sequência de teste e a sequência final de teste.

Ao clicar no botão **Save**, os casos de teste serão salvos dentro do diretório desejado.

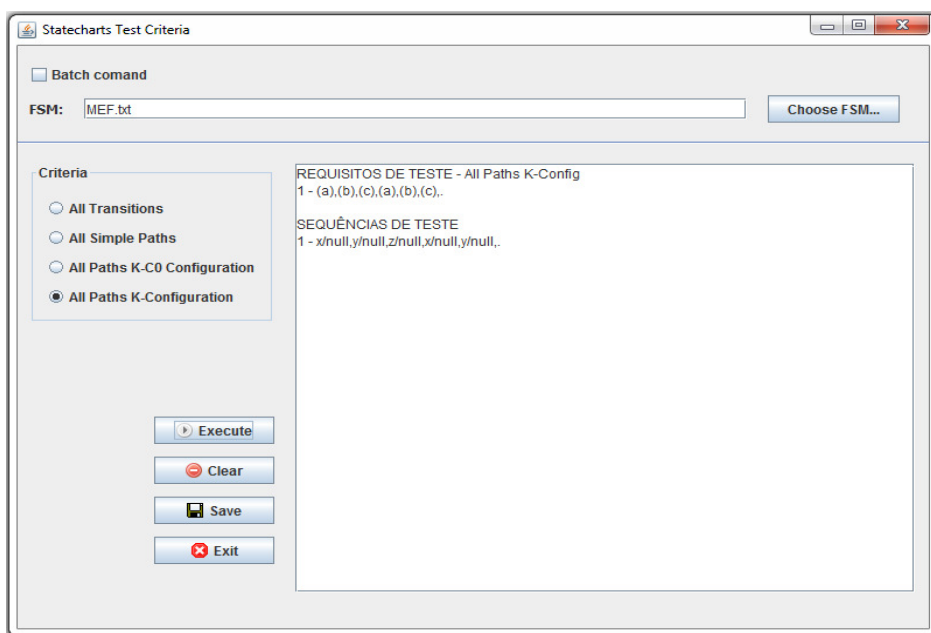


Figura 19 - Geração dos casos – Statecharts.

Ao clicar no botão **Save**, os casos de teste poderão ser salvos no local de sua preferência. Os casos de teste serão salvos no formato texto(.txt).

3.3.2 Exemplo 2

Sistemas que apresentam concorrência também podem ser modelados na ferramenta, para representar atividades paralelas entre componentes deve-se adicionar em um estado composto regiões concorrentes (basta-se clicar com o botão direito do mouse sobre o estado composto e depois na opção “*New Concurrent Region*”), estas regiões são separadas por linhas a traço interrompido, representando subestados concorrentes. Dependências entre regiões podem ser expressas através de condições de guarda (em um componente o estado do outro é testado).

Um exemplo de sistema com paralelismo é apresentado na figura 18. Este exemplo foi extraído de Santiago et al. (2010).

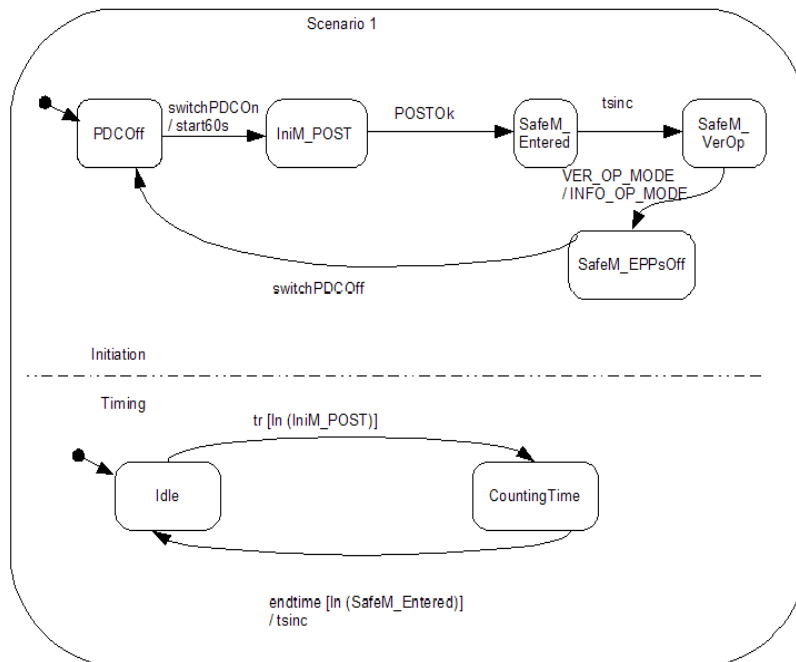


Figura 20 – Modelo Statecharts do cenário 01 (um) do software SWPDC.

Para modelar o cenário apresentado na figura 18 usando a ferramenta GTSC deve-se:

- 1) Adicionar um estado composto a partir clicando-se no botão “New Composite State”.

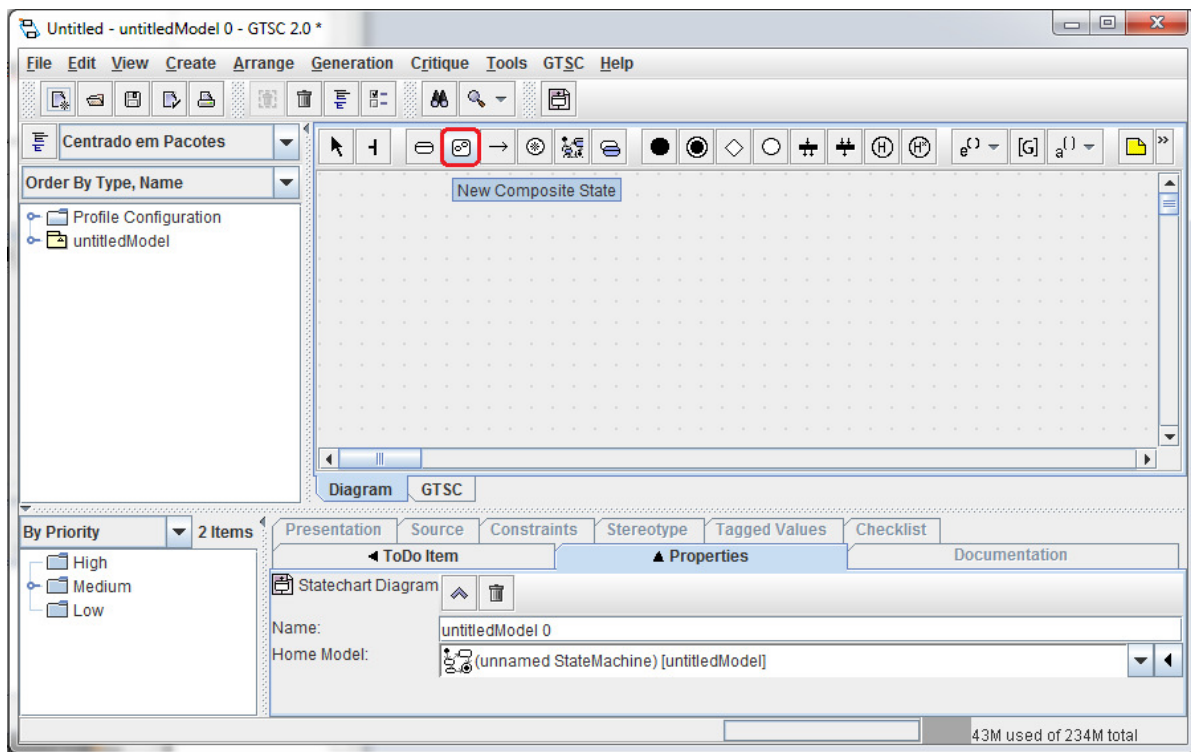


Figura 21 – Adicionando um estado composto.

- 2) Clicar com o botão direito do mouse sobre o estado composto e depois na opção “*New Concurrent Region*”.

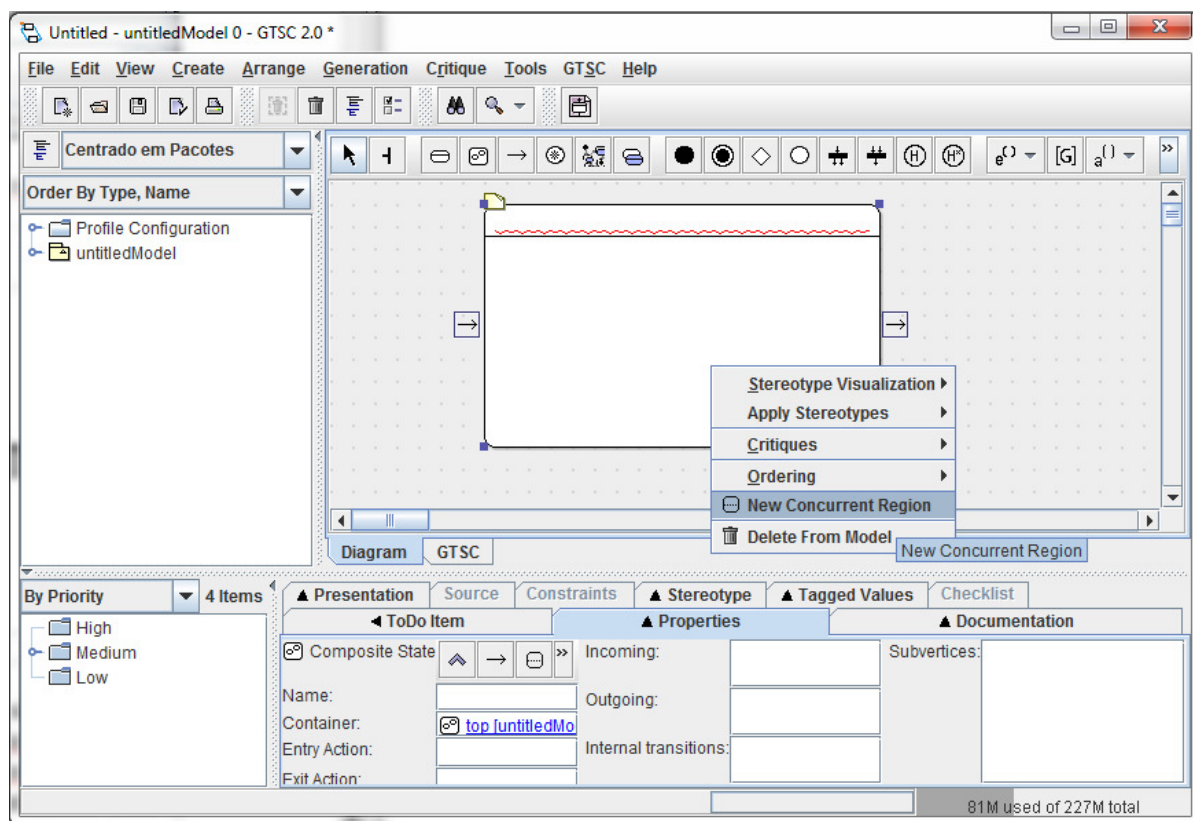


Figura 22 – Adicionando uma região concorrente.

- 3) Pode-se então adicionar os subestados.

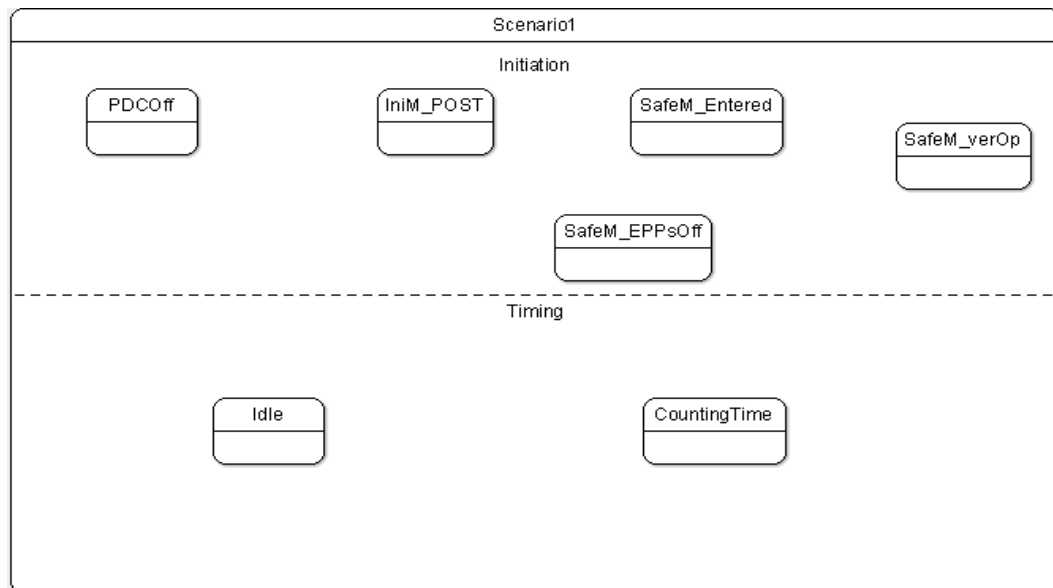


Figura 23 – Adicionando subestados.

- 4) As transições são criadas ao clicar-se nas setas que aparecem quando um estado é selecionado e arrastá-las até o estado desejado.

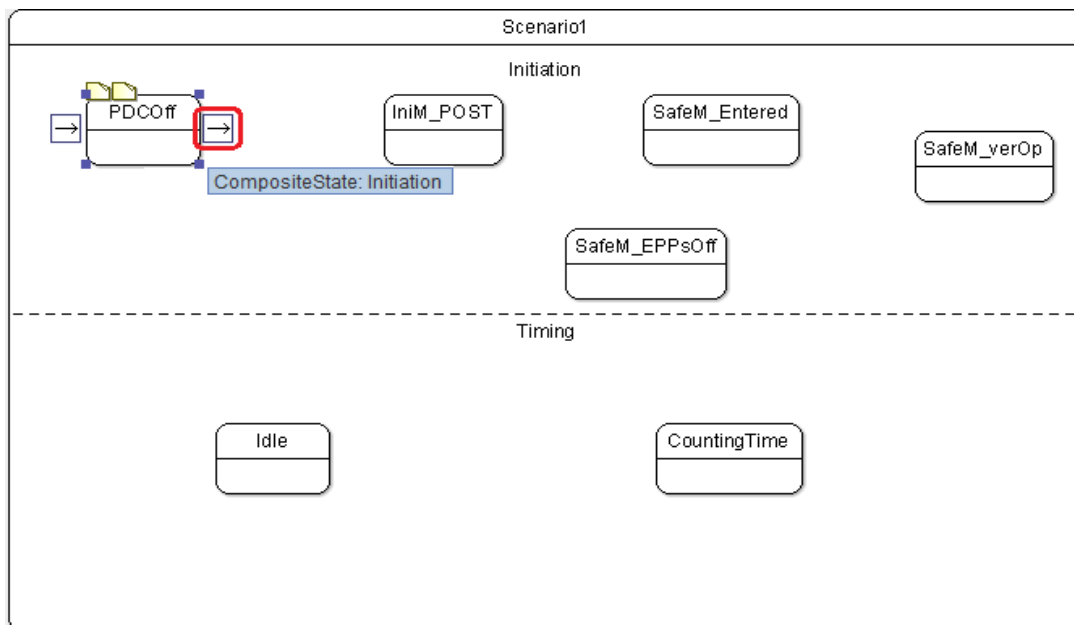


Figura 24 – Adicionando transições.

- 5) Para se adicionar um evento a uma transição desejada, é recomendado usar o tipo *Signal Event*, para isto basta-se clicar no botão de mesmo nome ou com um clique duplo na transição. Também se pode usar a transição do tipo “*Call Event*”.

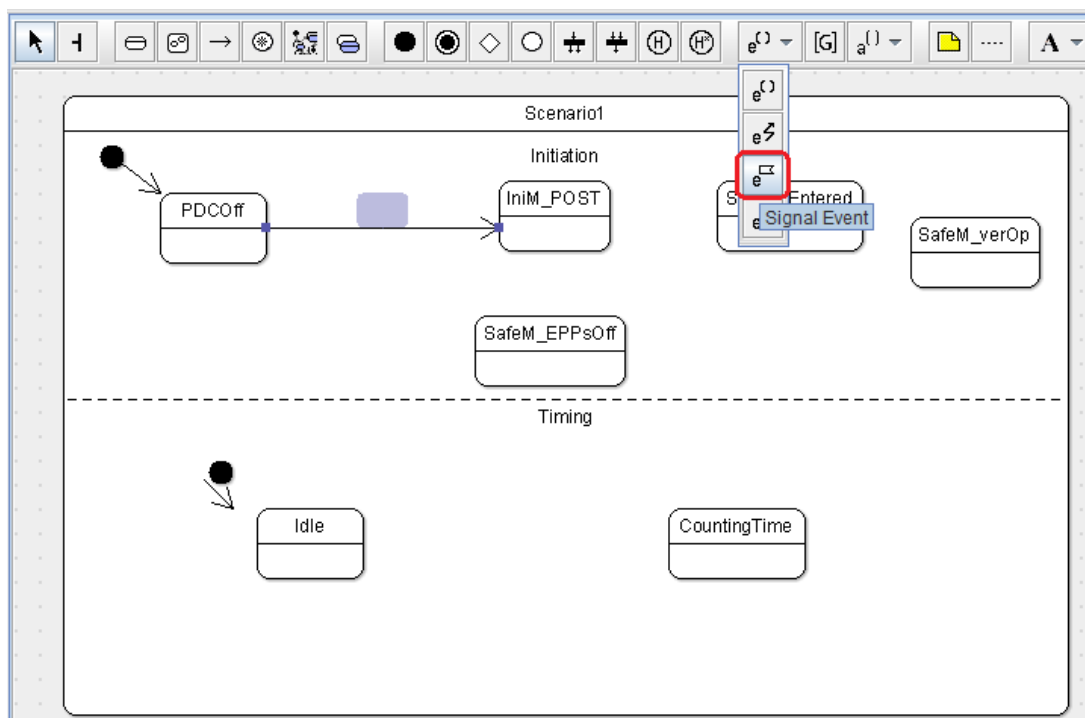


Figura 25– Adicionando eventos a transições.

- 6) Para se definir o estado padrão clica-se em “*New Initial*”

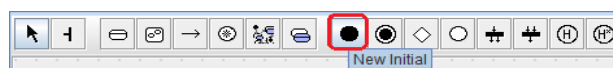


Figura 24– Adicionando marcador de estado padrão.

- 7) E liga-o ao estado padrão desejado

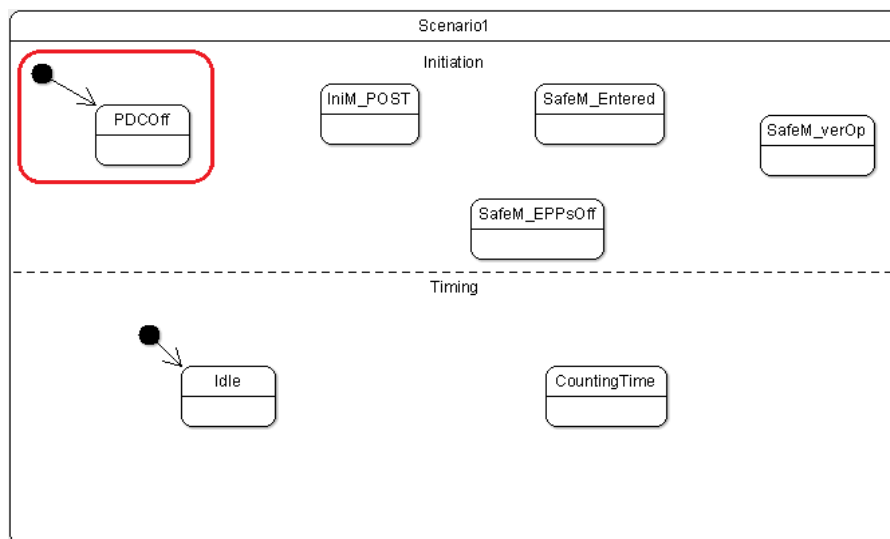


Figura 26– Definindo estado padrão.

- 8) Condições de guarda podem ser adicionadas a partir do botão “*New Guard*”

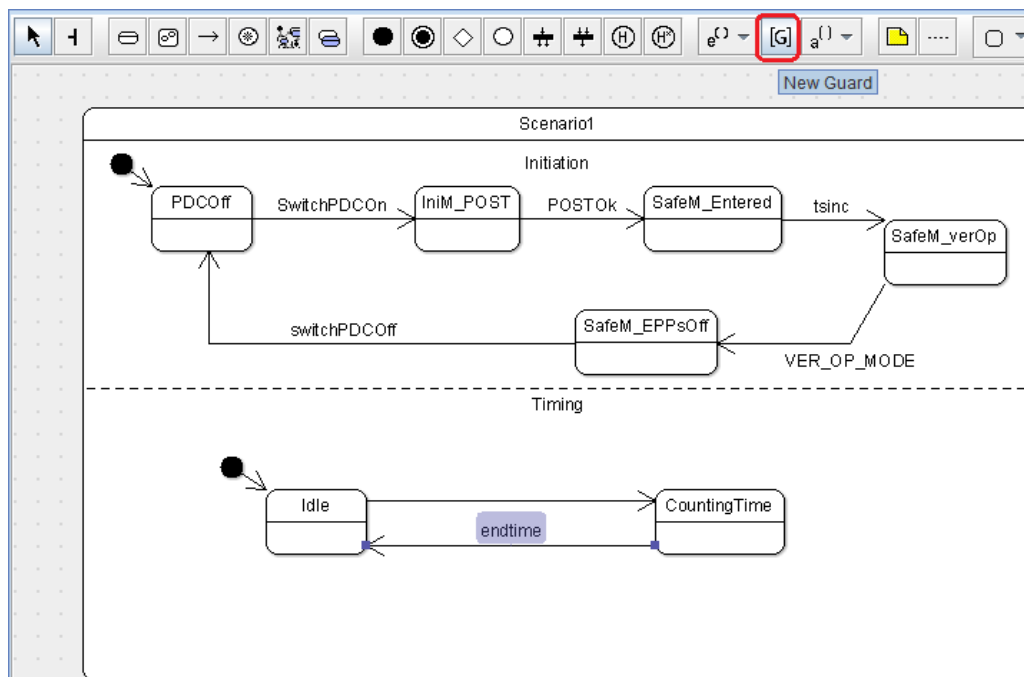


Figura 27– Adicionando guardas.

Aparecerá então, na aba “*Properties*” do painel de detalhes, campos para preenchimento de informações referentes à Guarda. Apenas a condição de guarda é obrigatória, sendo necessário o preenchimento do campo “*Expression*”.

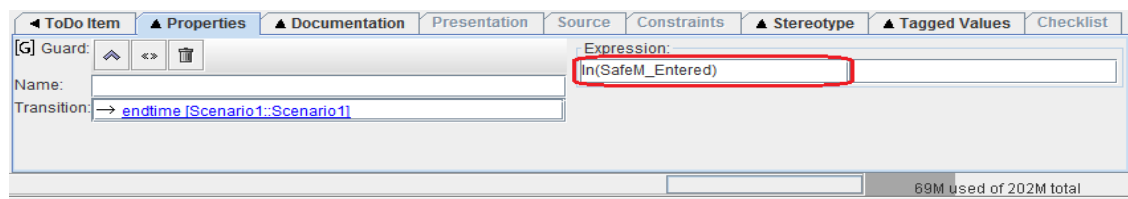


Figura 28– Definindo condições de guarda.

Os tipos de elementos permitidos nas condições de guarda são: “*InState*”, “*InParamState*” (não implementado), “*NotCondition*” e “*ComposedCondition*”. Da mesma forma, “*ANDCond*” e “*ORCond*” podem ser especificados dentro de “*ComposedCondition*”.

- *InState*: Basta-se colocar o estado desejado entre parênteses juntamente com o prefixo “In”. Exemplo: In(estado_desejado);
- *Not InState*: Basta-se colocar o estado desejado entre parênteses juntamente com o prefixo “NotIn”. Exemplo: NotIn(estado_desejado);
- *ANDCond*: Basta-se colocar entre as condições os caracteres “&&”. Exemplo: In(estado_desejado1)&&NotIn(estado_desejado2);
- *ORCond*: Basta-se colocar entre as condições os caracteres “||”. Exemplo: In(estado_desejado1)||NotIn(estado_desejado2);

Quando o evento for do tipo “*TrueCondition*”, deverá ser adicionado na modelagem um evento do tipo *Signal Event* nomeado de **tr** (obrigatoriamente) e ter sua condição de guarda adicionada da forma apresentada anteriormente.

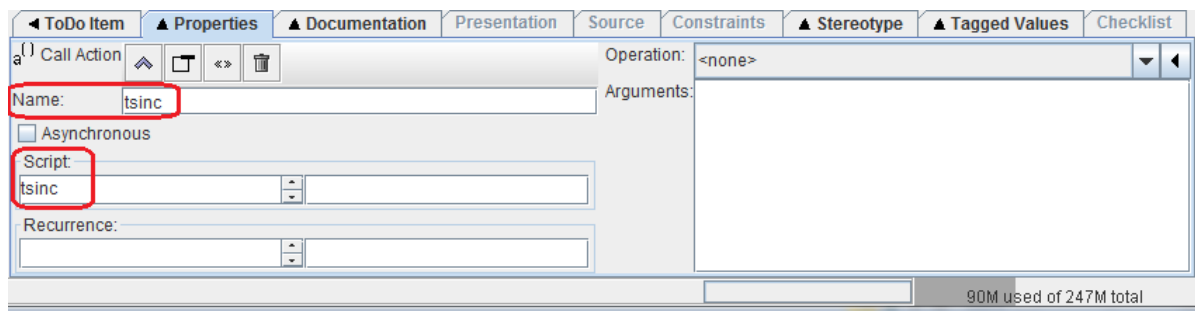


Figura 31– Campos “Script” e “Name”.

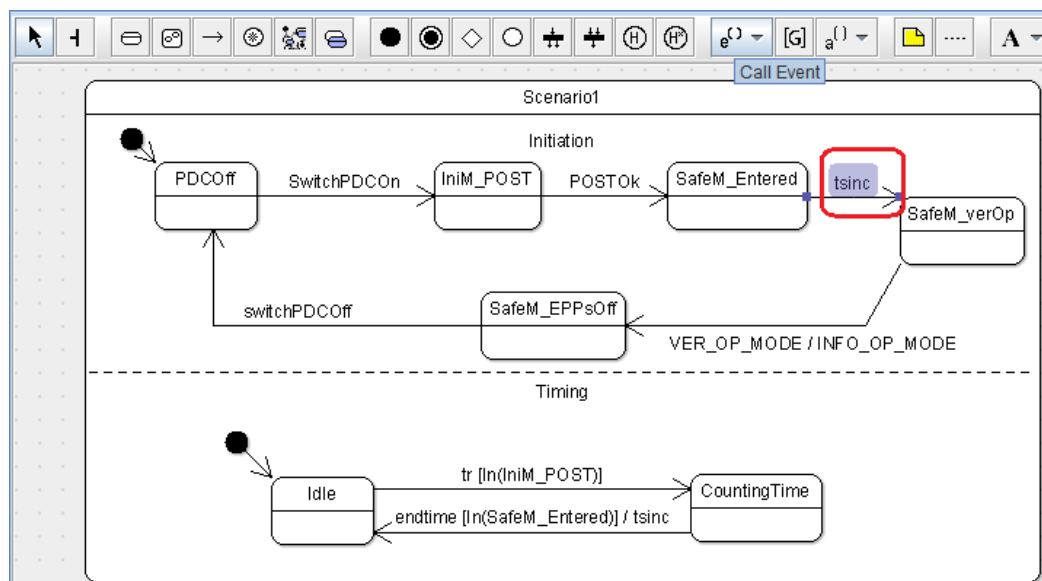


Figura 32– Adicionando um “Call Event” na transição que será disparada internamente.

- 10) Uma vez modelado o sistema pode-se obter a especificação PcML, clicando-se na aba GTSC.

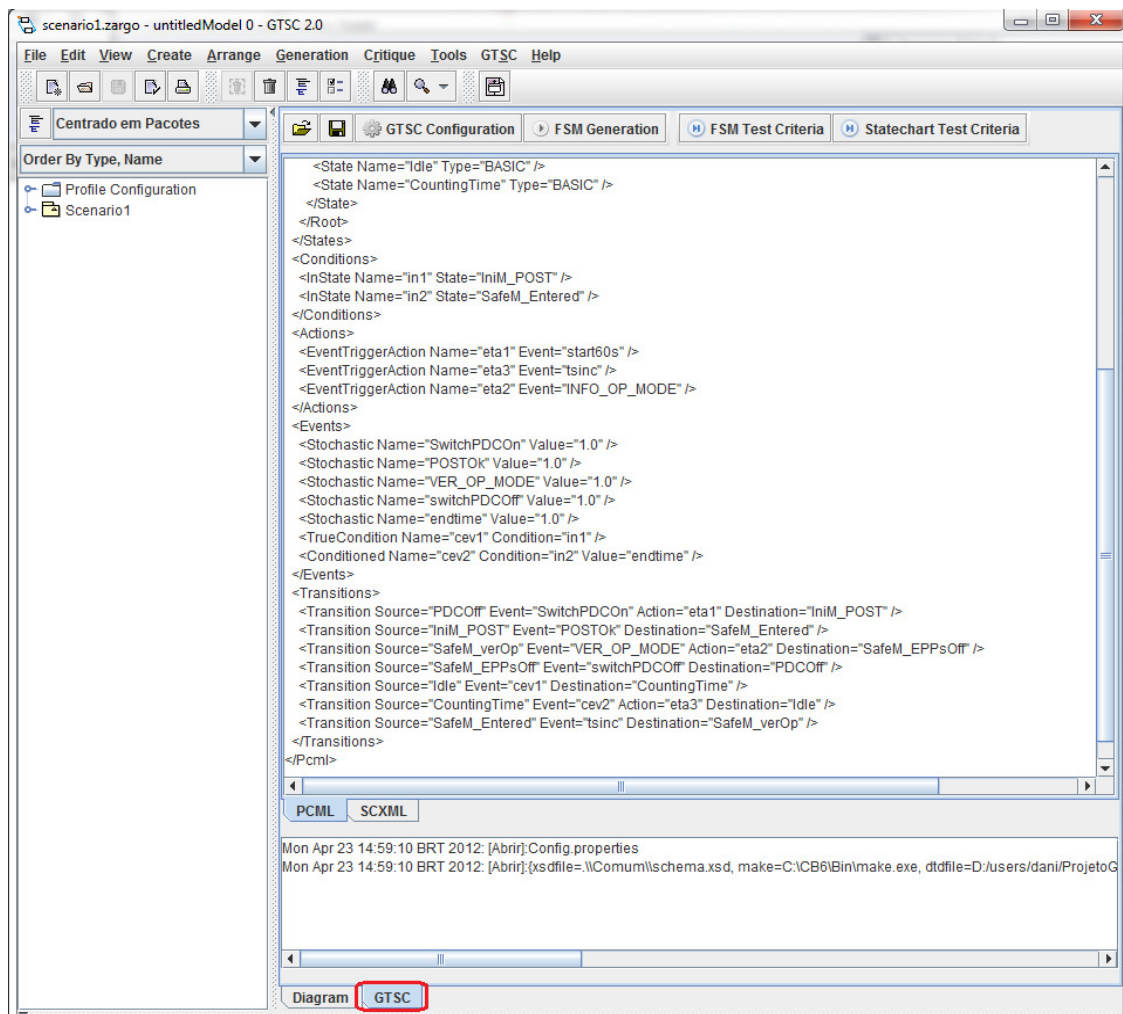


Figura 33– Modelo PcML gerado.

11) Para gerar a Máquina de Estados Finitos (*Finite State Machine* - FSM) e obter os casos de testes é só proceder como descrito no primeiro exemplo desta seção.

3.3.3 Exemplo 3

Para este exemplo será modelado o sistema apresentado na figura abaixo. Este modelo foi extraído de Santiago et al. (2010).

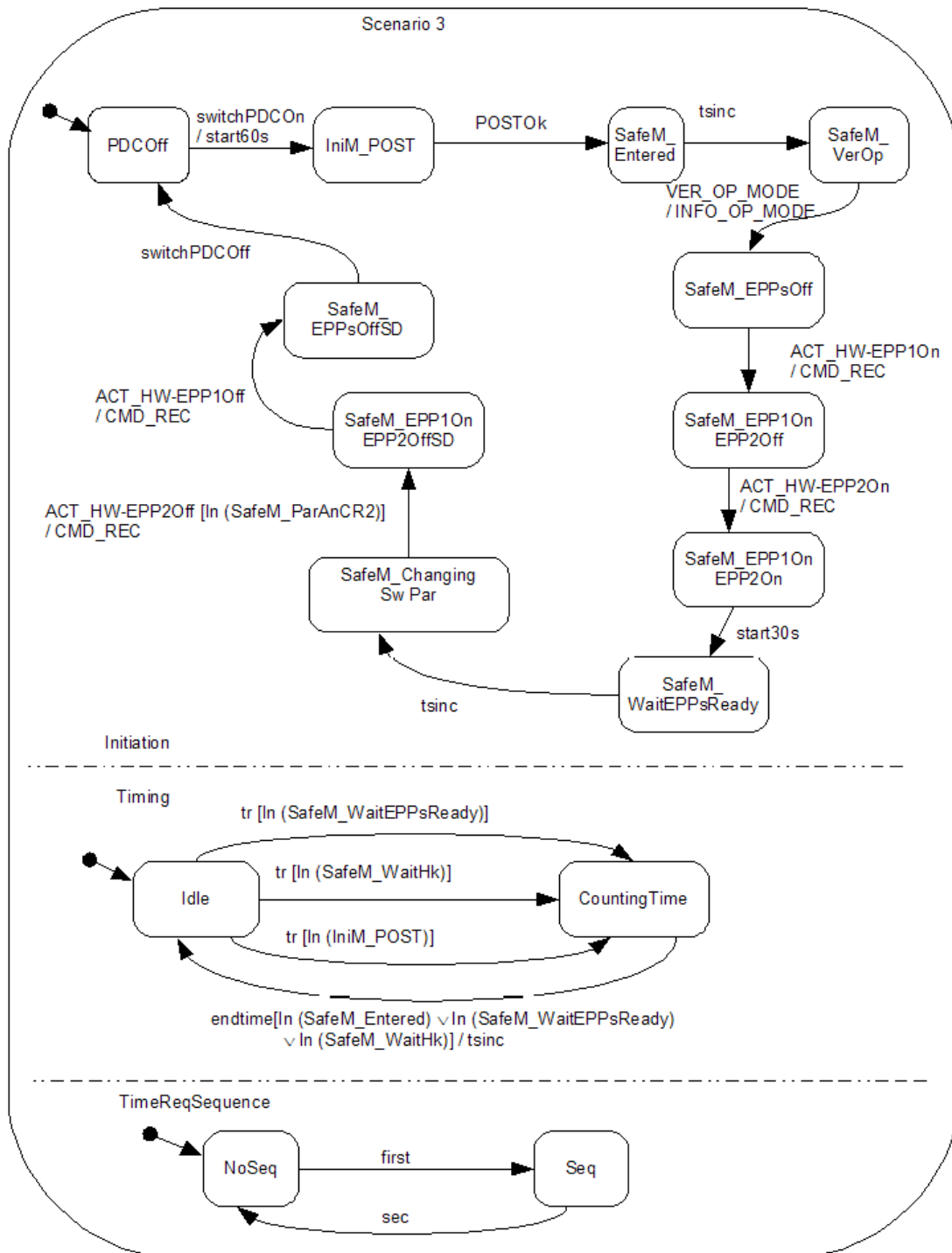


Figura 34 – Modelo Statechart: Scenario 3 (modelo principal).

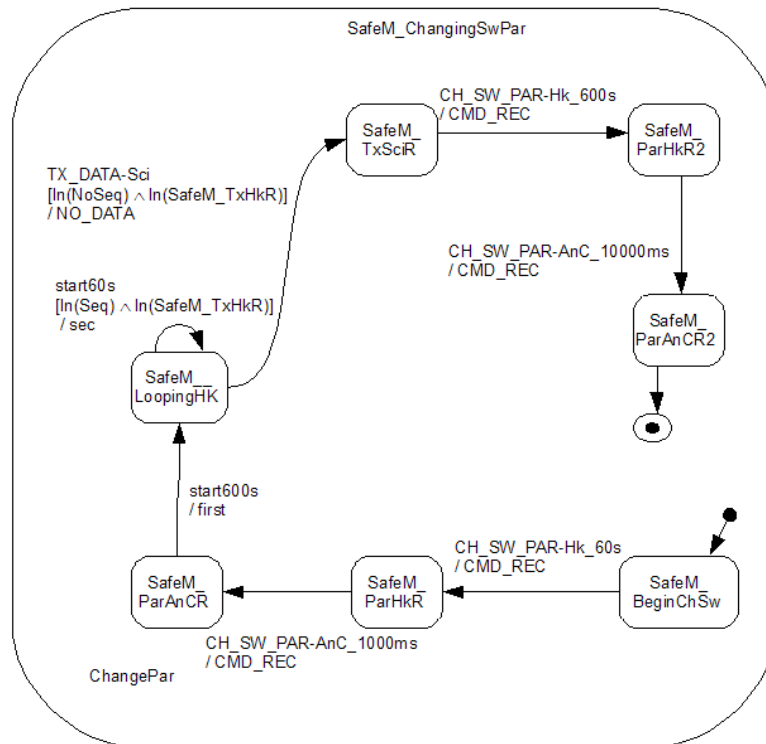


Figura 35 – Modelo Statechart: Scenario 3 (segundo nível de hierarquia).

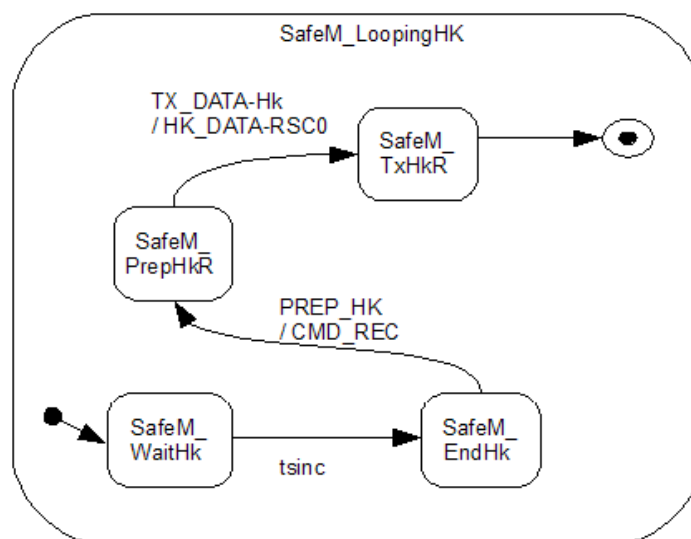


Figura 36 – Modelo Statechart: Scenario 3 (terceiro nível de hierarquia).

Para modelagem deste sistema no GTSC foram adicionados todos os estados presentes nos três níveis de hierarquia. Como pode ser observado na figura a seguir

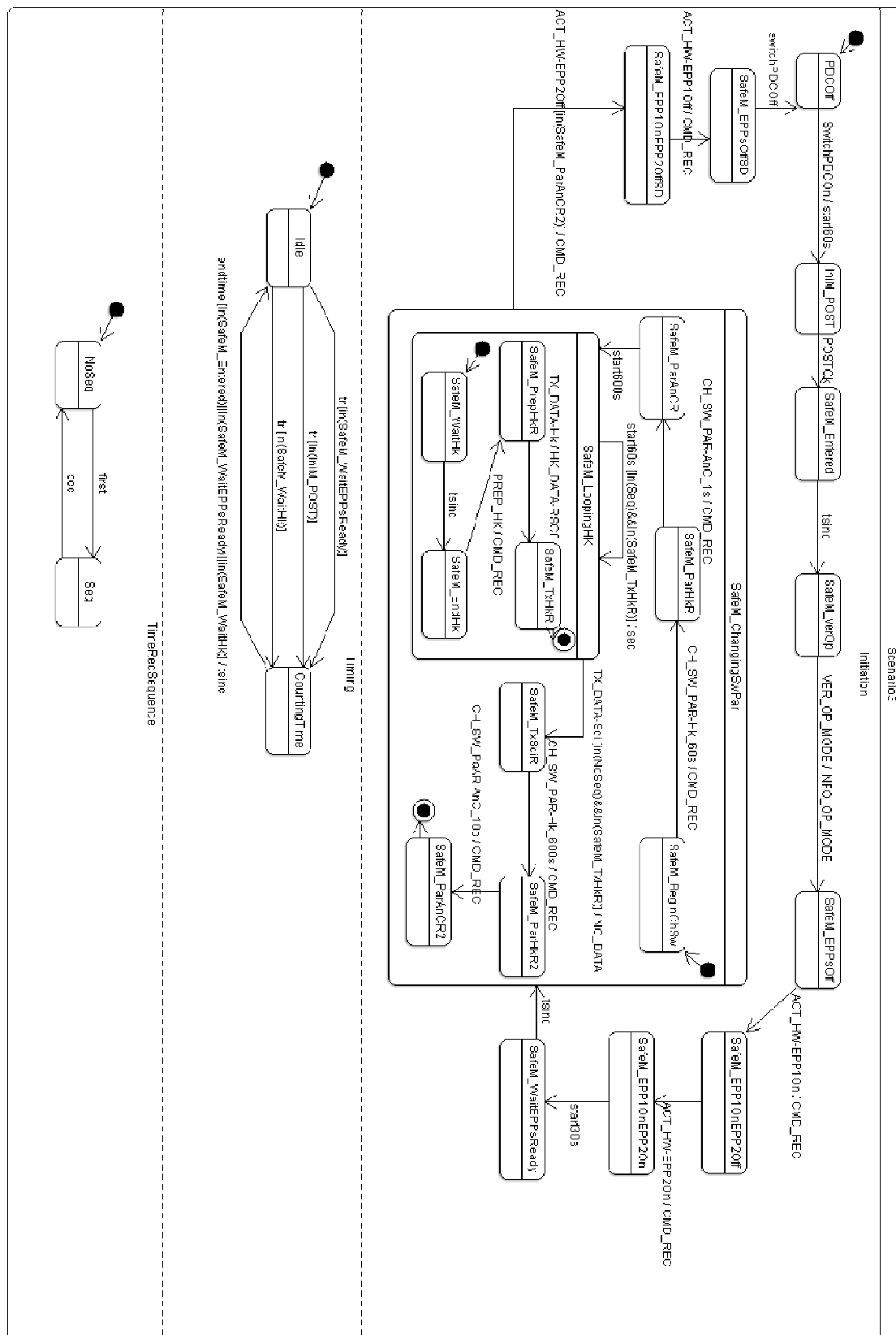


Figura 37 – Modelo Statechart completo do Scenario 3.

Inicialmente foi inserido um estado composto. Adicionaram-se três regiões de concorrências e os estados presentes em cada região. Os eventos (*EventTriggerAction* e

OutputTriggerAction), ações foram adicionados como nos exemplos anteriores. Neste exemplo existem transições com guardas com condições compostas do tipo AND e OR.

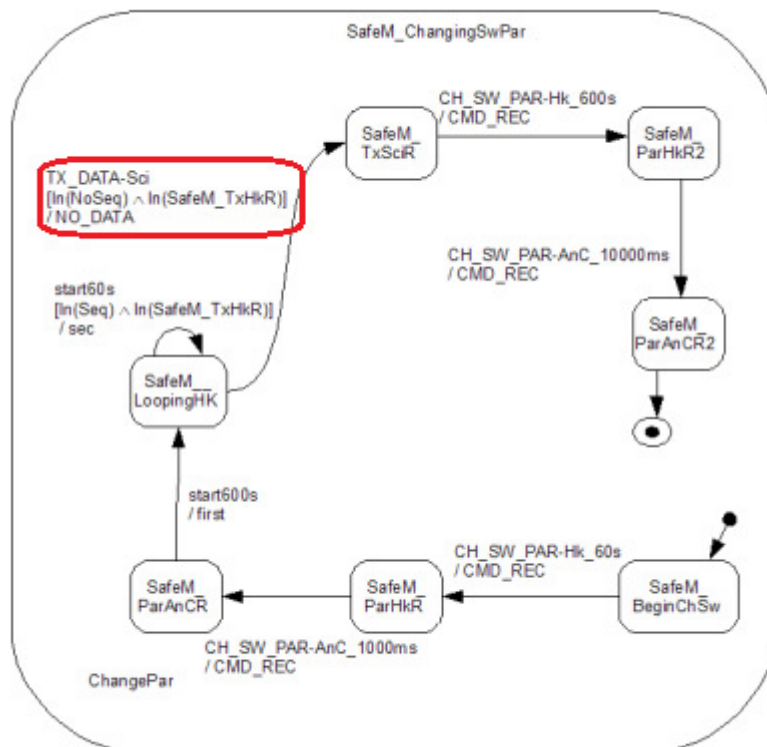


Figura 38 – Exemplo de guarda composta AND.

Como pode-se observar no fragmento acima, existe uma guarda composta AND. Como já foi apresentado anteriormente, para modelagem do AND no GTSC utiliza-se &&, semelhante ao que é feito em algumas linguagens de programação, para representar a guarda apresentada, selecionou-se a transição e clicou-se no botão *New Guard* e preencheu o campo *Expression* com valor: **In(NoSeq)&&In(SafeM_TxHkR)**

Onde o prefixo In significa a condição de guarda InState e && entre as condições significa condição1 AND condição2, como já foi explicado anteriormente.

Figura 39 – Campo *Expression* preenchido com guarda composta AND.

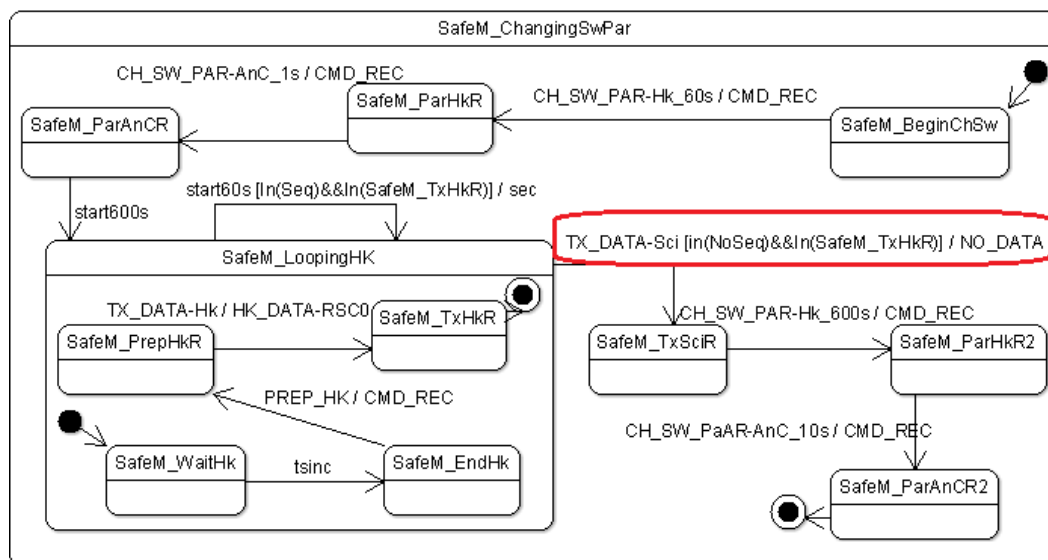


Figura 40 – Exemplo de guarda composta AND modelada no GTSC.

Para condição composta OR procede-se de forma análoga, utilizando os caracteres || para representa-la.

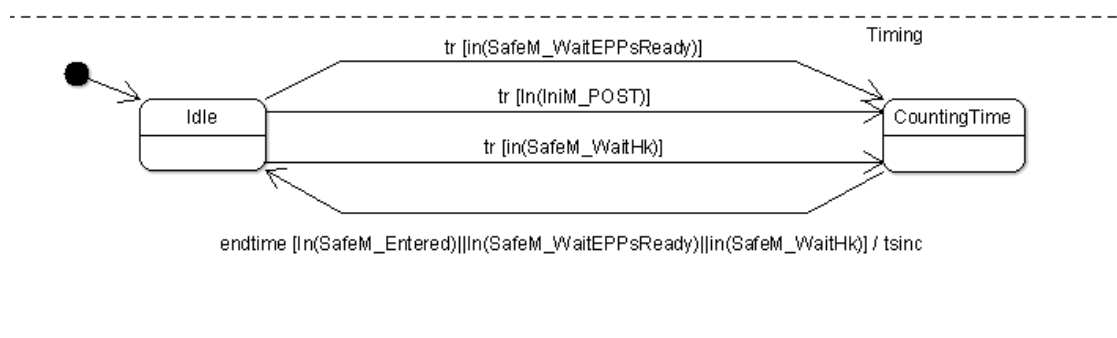
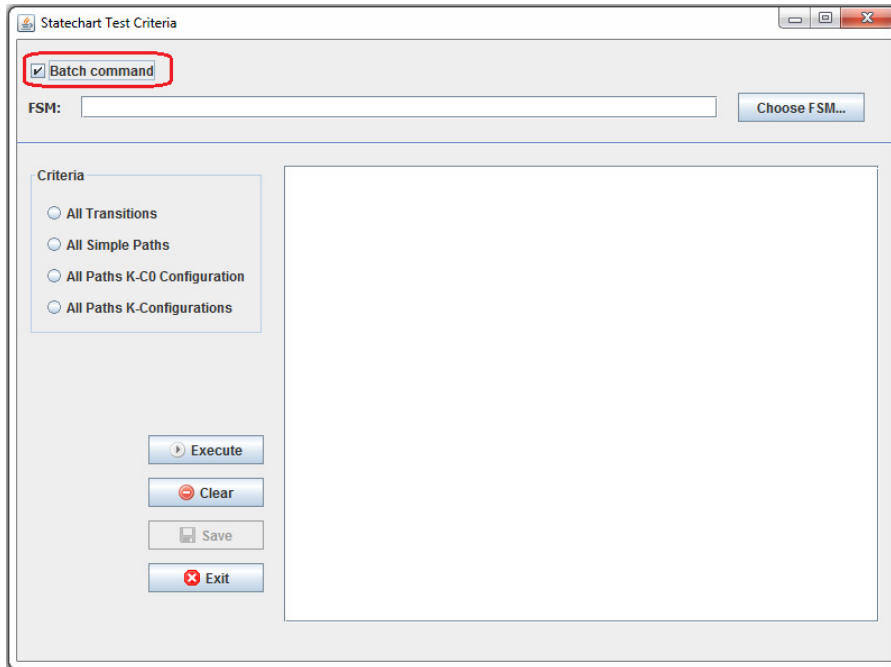


Figura 41 – Exemplo de guarda composta OR modelada no GTSC.

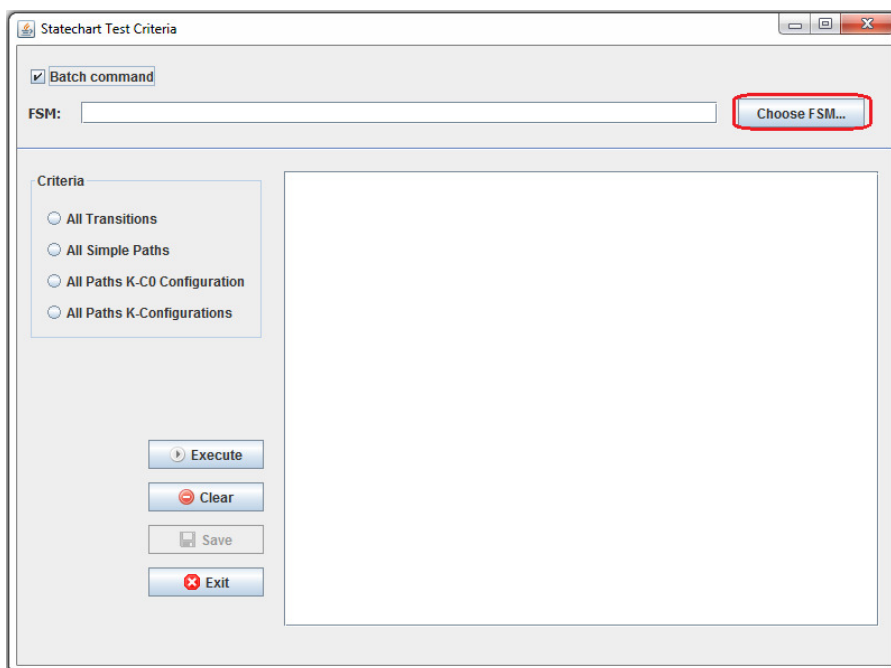
O resto do sistema foi modelado conforme explicado nos exemplos anteriores.

3.4 Batch command

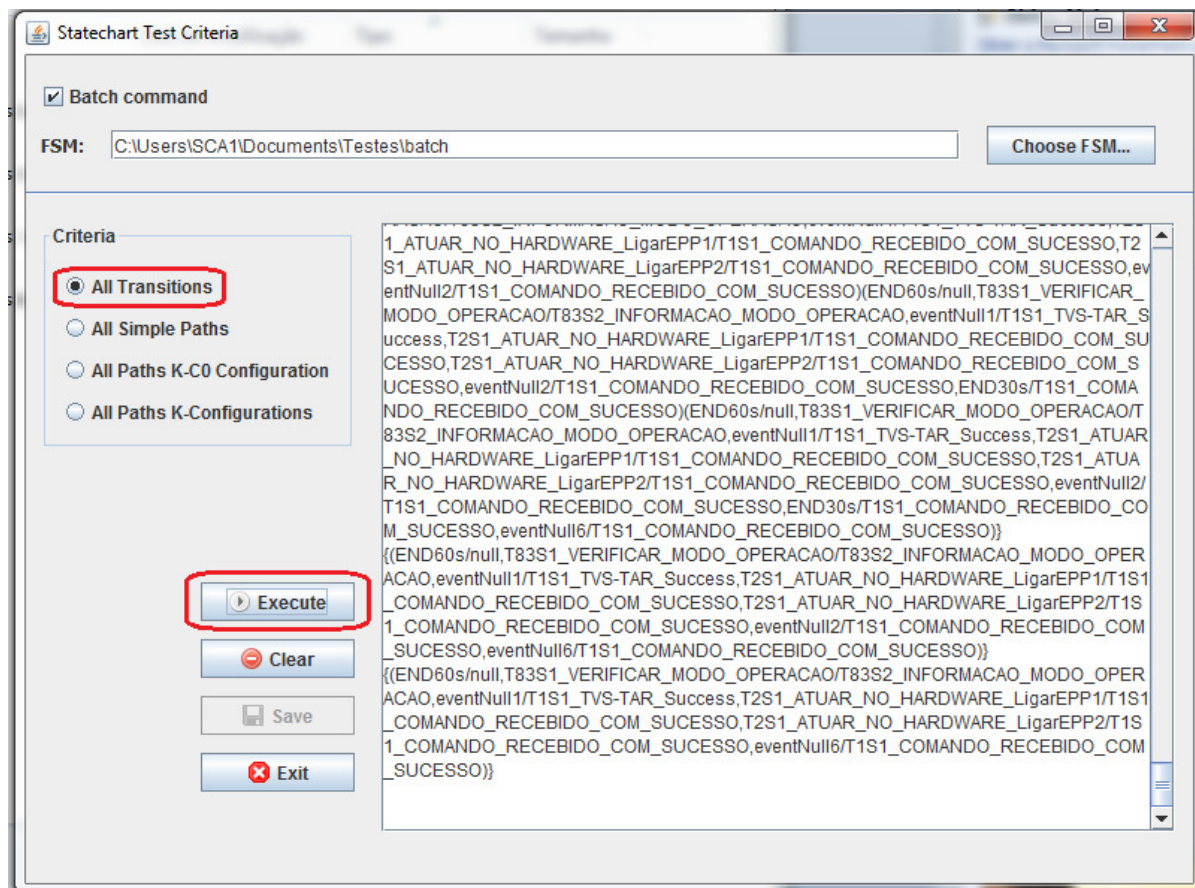
Para geração de casos de testes usando os critérios de *Statecharts* de várias MEF's geradas anteriormente é possível utilizar o comando *Batch Command*. Para utilizar esta funcionalidade basta marcar a opção *Batch Command*.



Selecionar o diretório com as MEF's geradas, clicando em *Choose FSM...*



Selecionar o critério de teste e executar.



No diretório selecionado com as MEF's serão salvos os arquivos gerados com critério selecionado.

Biblioteca Documentos

batch

Organizar por: Pasta

Nome	Data de modificação	Tipo	Tamanho
SWPDC Cenário 002 Unfolding 04 all-transitions.txt	04/05/2012 18:18	Documento de Te...	21 KB
SWPDC Cenário 002 Unfolding 04 FSM.txt	03/02/2012 14:32	Documento de Te...	17 KB
SWPDC Cenário 002 Unfolding 05 all-transitions.txt	04/05/2012 18:18	Documento de Te...	21 KB
SWPDC Cenário 002 Unfolding 05 FSM.txt	03/02/2012 14:36	Documento de Te...	17 KB
SWPDC Cenário 002 Unfolding 06 all-transitions.txt	04/05/2012 18:18	Documento de Te...	21 KB
SWPDC Cenário 002 Unfolding 06 FSM.txt	03/02/2012 14:37	Documento de Te...	17 KB
SWPDC Cenário 002 Unfolding 07 all-transitions.txt	04/05/2012 18:18	Documento de Te...	21 KB
SWPDC Cenário 002 Unfolding 07 FSM.xml	03/02/2012 14:39	Documento XML	17 KB

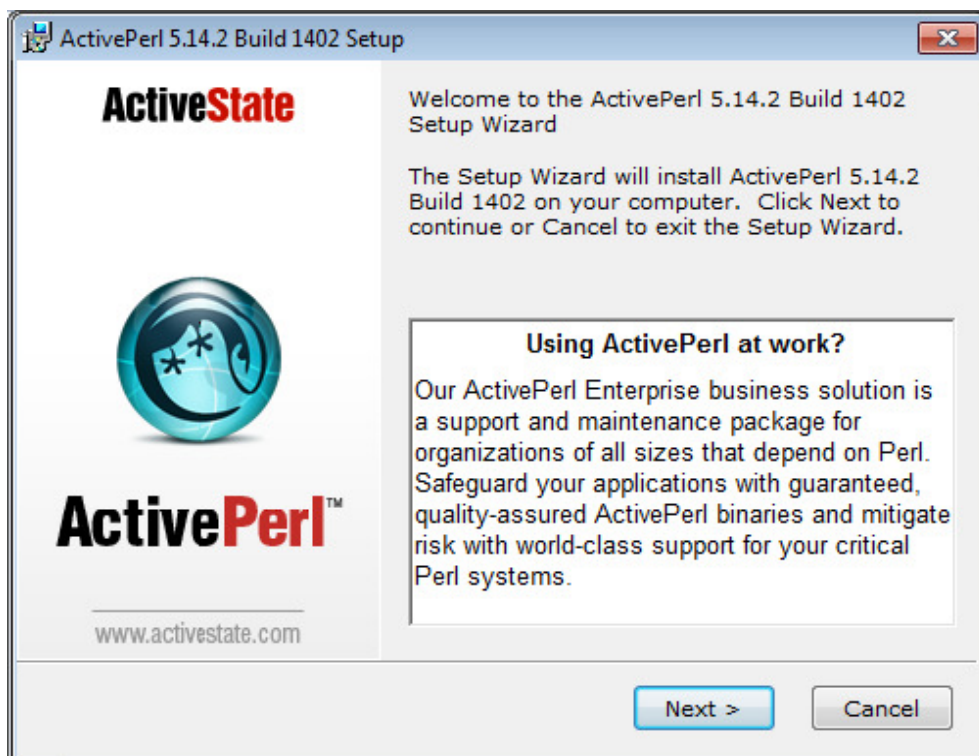
REFERÊNCIAS BIBLIOGRÁFICAS

SANTIAGO, V. A.; CRISTI A, M.; VIJAYKUMAR, N. L. Model-based test case generation using Statecharts and Z: A comparison and a combined approach. São José dos Campos, 2010. 72 p. (INPE-16677-RPQ/850). Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19@80/2010/02.26.14.05>>. Acesso em: 14 abr. 2010. 32, 68, 82, 83, 97, 109, 110, 111

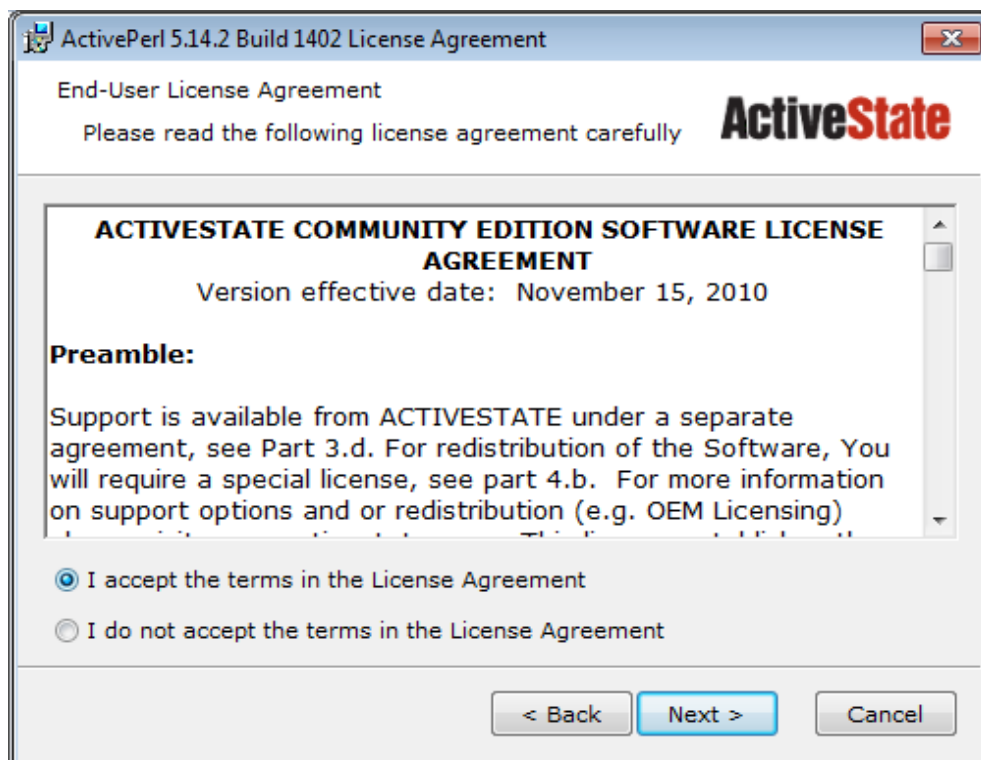
Vijaykumar, N.L.; Carvalho, S.; Abdurahiman, V. On proposing Statecharts to specify performance models. **International Transactions in Operational Research**, v. 9, n. 3, p. 321-336, 2002.

APÊNDICE A - Instalação PERL

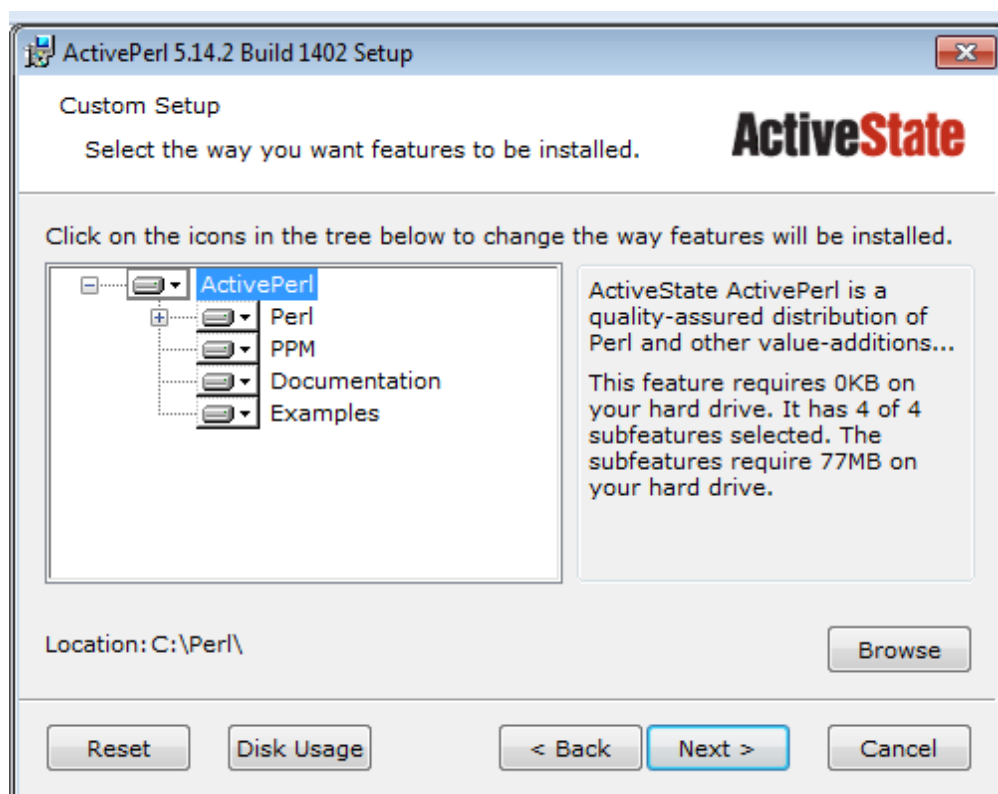
1. O download do Perl pode ser feito no seguinte endereço:
<http://www.activestate.com/activeperl/downloads> (Acesso em: 15/04/2012). A versão escolhida foi “**ActivePerl 5.14.2**” para Windows.
2. Terminado o download, clique duas vezes sobre o arquivo para iniciar a instalação. Será aberta a seguinte tela:



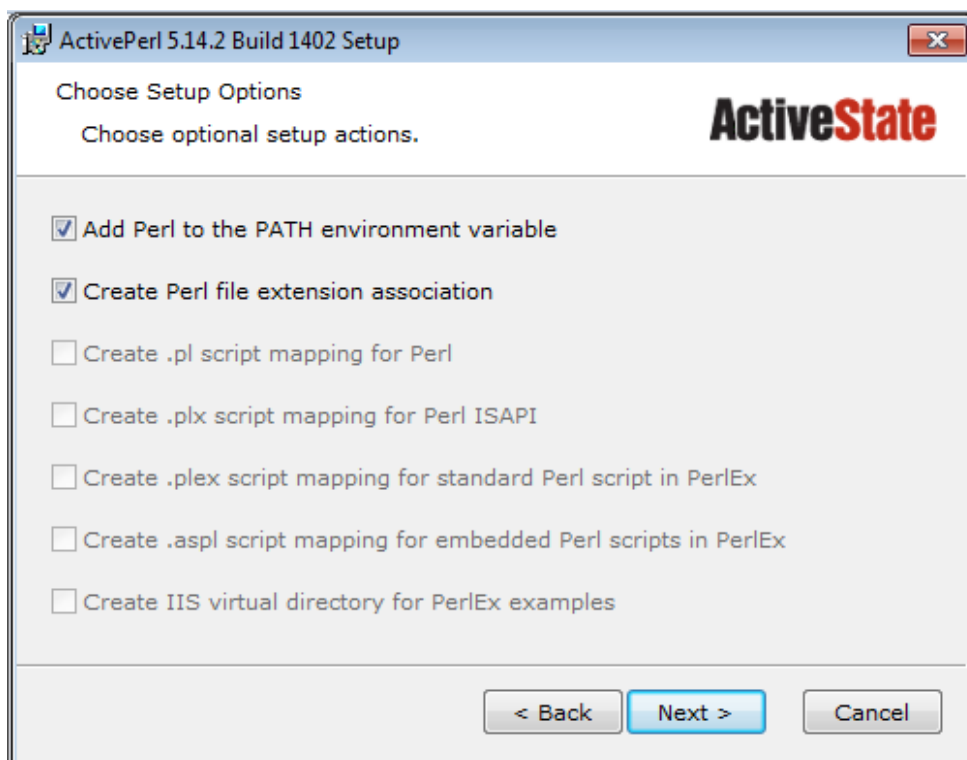
3. Nessa tela, clique em **Next** para continuar a instalação. O próximo passo é aceitar os termos de uso e novamente clicar em **Next**.



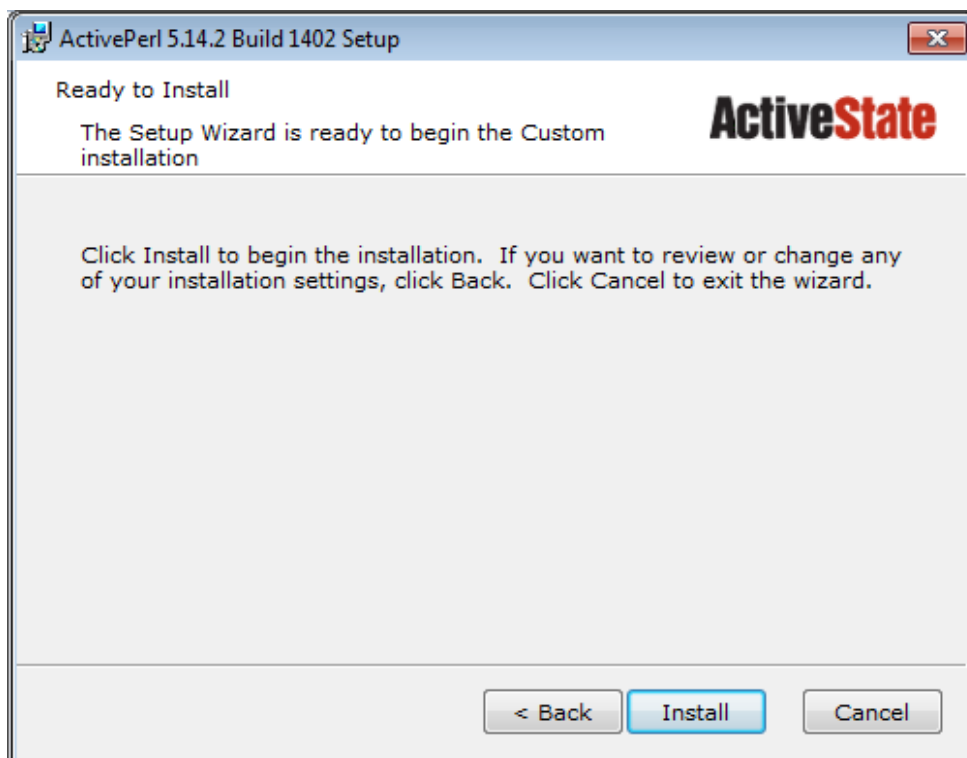
4. Escolha o local de instalação do Perl (C:\Perl) e clique em **Next**.



5. Em seguida, deixe as duas primeiras opções marcadas, como mostra a figura, e clique em **Next**.



6. Na tela seguinte clique em **Install** e aguarde até a instalação ser concluída.



7. Clique em **Finish** para concluir a instalação.

