

MANUAL DO DESENVOLVEDOR GCOMPRIS

Emmanuelle A. Cota Quites

e-mail: manuacq@yahoo.com.br

Copyright (c) 2007 Emmanuelle Quites.

É garantida a permissão para copiar, distribuir e/ou modificar este documento sob os termos da Licença de Documentação Livre GNU (GNU Free Documentation License), Versão 1.2 ou qualquer versão posterior publicada pela Free Software Foundation; sem Seções Invariantes, Textos de Capa Frontal, e sem Textos de Quarta Capa. Uma cópia da licença é incluída na seção intitulada "GNU Free Documentation License".

Sumário

1	Introdução.....	4
1.1	Por que GCompris?.....	4
1.2	Instalação e Obtenção.....	4
2	Manual do Usuário.....	6
2.1	Controle.....	6
2.2	Jogando o GCompris.....	6
2.2.1	Janela Principal.....	6
2.2.2	Inicializando um Jogo.....	7
2.2.3	Terminando um Jogo.....	7
3	Assuntos de Internacionalização.....	8
3.1	Minha linguagem é suportada?.....	8
3.2	Como traduzir o Gcompris para a minha linguagem?.....	8
4	Guia dos Desenvolvedores.....	8
4.1	Por que acrescentar módulos ao GCompris?.....	8
4.2	Como acrescentar módulos ao GCompris?.....	9
4.2.1	A Figura Completa.....	9
4.2.2	Gcompris API.....	10
4.2.3	Exemplo de como duplicar um módulo já existente.....	12
4.2.4	Solução Alternativa.....	14

1. INTRODUÇÃO

Este manual foi construído a partir de traduções de alguns arquivos contidos na documentação do pacote fonte do GCompris e apresenta um 'howto' de como acrescentar um novo módulo/jogo para o software livre e educacional GCompris. Também é apresentado uma visão geral dos controles do GCompris e de questões de internacionalização. Espera-se que este manual possa trazer alguns esclarecimentos sobre como acrescentar um novo módulo ao GCompris.

1.1 Por que GCompris?

Hoje em dia, o sistema operacional Linux não possui muitas ferramentas educacionais úteis para crianças. E quando estas existem, são difíceis de encontrar e instalar. E também, torna-se complicado para os professores/pais administrarem diversas ferramentas pequenas e independentes.

O GCompris foi projetado de maneira que, acrescentar novos jogos, seja uma tarefa mais fácil. A intenção do GCompris é promover em um único pacote diferentes tipos de conteúdo educacional pequeno, chamado de 'boards'. Este 'boards, também chamados de módulos, são na verdade pequenos jogos de cunho pedagógico educacional. Hoje já há muito software livre que provê pequenos jogos de conteúdo educacional. Mas, infelizmente, é difícil localizar, instalar e usá-los porque eles não evoluem à mesma velocidade e oferecem a sua própria interface. GCompris aponta para prover uma interface do usuário sem igual, que dá acesso a diferentes jogos em um só pacote. GCompris faz parte do projeto de GNU.

1.2 Instalação e Obtenção

O passo inicial obviamente é instalar o pacote GCompris. Uma vez que o objetivo é criar um novo módulo e, para isto, torna-se necessário estudar e trabalhar no código fonte, foi necessário fazer o download do arquivo gcompris-XX.tar.gz(onde XX é a versão do programa) pelo site [<http://www.gcompris.net/-Obtencao-e-instalacao->] e instalar o GCompris pelo terminal através dos comandos:

1. ``cd'` para entrar no diretório onde se encontra o pacote com o código fonte. Logo depois, digite `'./configure'` para configurar o pacote no seu sistema.
2. Digite ``make'` para compilar o pacote.
3. Digite `'make install'` para instalar os programas e quaisquer outros arquivos de dados e a documentação. Por default, ``make install'` irá instalar os arquivos do pacote em ``usr/local/bin'`, ``usr/local/man'`, etc.
4. Você pode remover os arquivos binários e os arquivos objetos do diretório do código fonte com o comando `'make clean'`. Para também remover os arquivos `'configure'` criados (então você pode compilar o pacote em diferentes tipos de computadores), digite ``make distclean'`.

Logo no primeiro comando, provavelmente aparecerão erros de dependência de pacotes. Os pacotes que estiverem faltando, aparecerão na linha de comando, basta então instalá-los. O comando para instalar estes pacotes depende da distribuição Linux utilizada. Por exemplo, vamos supor que falte o pacote `'gnuchess'`. No Ubuntu, basta você digitar no terminal o seguinte comando: `sudo apt-get install gnuchess` e o pacote em questão será instalado.

GCompris requer `gtk-libs`, `libxml2`, `gnomecanvas`, `sqlite`, e `gdk-pixbuf`. Também requer os pacotes `gnuchess` e `gnucap` para a atividade de xadrez e eletricidade, respectivamente.

GCompris inclui várias atividade escritas em `python`. Para habilitá-las você precisa ter o `python` instalado no seu sistema com os seguintes pacotes: `libpython`, `python`, `libpython2.4-devel`, `python-base`, `python-sqlite2` (faça o download e instale-o através do site [<http://initd.org/tracker/pysqlite>]).

Como o som é administrado interiormente, você também precisa dos seguintes pacotes: `libSDL`, `libSDL-mixer`.

O GCompris usa o GTK e a biblioteca GNOME (especialmente a GNOME canvas).

2. MANUAL DO USUÁRIO

2.1 Controle

Todos os controles são projetados para serem amigáveis para os usuários mais jovens. Por isto, o GCompris não adere ao padrão do Gnome para interfaces do usuário. GCompris é controlado com o mouse. Não pode ser usado sem um mouse ou um dispositivo de ponteiro.

2.2 Jogando o GCompris

2.2.1 Janela Principal

Uma vez começado, o GCompris apresenta uma visão gráfica que inclui de cima para baixo:

1. Lista de ícones com os Jogos: Cada ícone representa um jogo. Quando você move o mouse em cima deles, eles são realçados e são exibidas o nome e descrição do jogo na área de descrição do jogo.
2. Área de Descrição dos Jogos: pequena descrição do que o jogo é. Note que as descrições são internacionalizadas o que significa que as traduziram em um idioma de destino (Veja 3. Regras de Internacionalização para mais informações sobre Internacionalização).
3. Controle de barra: A barra de controle está sempre presente no GCompris. Estes ícones são contextuais, o que significa que quando um ícone não for significativo em um contexto específico, simplesmente não é exibido. A barra de controle contém os seguintes ícones da esquerda para a direita:
 - **Help**: Em alguns casos, um jogo pode ser muito complexo para ser descrito apenas na área de descrição dos jogos. Neste caso, este ícone proverá acesso às ordens de instruções adicionais.
 - **Nível**: Alguns jogos provêm diferentes níveis. O número de níveis é dependente do jogo.

- **OK:** Alguns jogos não são detectados automaticamente assim que a criança tenha terminado determinada tarefa. Clicando em 'OK' é semelhante a entrar em RET no teclado.
- **END:** Termina o jogo atual se qualquer, caso contrário sai do GCompris. Quando entrar ao nível de menu de um jogo, criará uma janela de diálogo para confirmar a saída do GCompris.
- **About:** Exibe o about para o GCompris com o número da versão, o autor, a licença e ligações para recursos on-lines.
- **Configure:** Exibe a caixa de configuração para o GCompris. A configuração pode ser feita ao nível do GCompris quando nenhum jogo está selecionado. Se um jogo é selecionado, isto pode provê opções de configuração. Esta configuração é persistente e salva no arquivo de gnome no diretório home do usuário abaixo *..gnome/gcompris*.

2.2.2 Inicializando um jogo

Ao nível de menu, apenas clique no ícone do jogo.

2.2.3 Terminando um jogo

Na barra de controle, selecione o ícone mais a direita.

3. ASSUNTOS DE INTERNACIONALIZAÇÃO

3.1 Minha linguagem é suportada?

GCompris é internacionalizado e hoje já apóia diferentes idiomas. Você pode

conferir o 'po', diretório encontrado dentro do pacote fonte do GCompris. Se seu idioma é traduzido, conterá um arquivo da forma 'pt_BR.po' onde pt_BR significa português do Brasil. Se seu idioma não é apoiado, você pode ajudar fazendo isto para o GCompris.

3.2 Como traduzir o GCompris para a minha linguagem?

Primeiro, acrescente seu nome de local em ALL_LINGUAS dentro do *configure.in*. Restaure o arquivo de configuração com *autoconf*.

Segundo, gere um arquivo recente ou novo '*gcompris.pot*' usando o comando `cd gcompris/po make update-po`.

Agora, pegue o arquivo '*gcompris.pot*', e o copie para um arquivo chamado 'XX.po' onde XX representa para qual linguagem irá ser traduzido.

Leia o arquivo, ele contém todo o texto em inglês achado no GCompris. Você apenas precisa escrever sua tradução na área vazia depois de cada frase em Inglês.

Para conferir sua tradução, você tem que instalar o GCompris.

Então, envie a tradução ao mantenedor do GCompris. Ele/ela acrescentará isto na próxima liberação oficial.

4. GUIA DOS DESENVOLVEDORES

4.1 Por que acrescentar módulos no Gcompris?

Se você tem uma idéia de jogo que ache satisfatório para crianças, você pode criar um projeto novo. Vamos chamá-lo de gldea. Se você começar do nada, você terá que implementar um menu, uma barra de estado, mecanismo de avaliação de usuário e assim por diante.

Se a gldea é um programa complexo e enorme, pode ser mais fácil e mais rápido para você implementar seu gldea como um módulo/jogo do GCompris. Deste modo, você se beneficia de toda a infra-estrutura atual e pode focalizar apenas na sua idéia.

Para o usuário final, é também mais fácil ter uma única interface para um grupo de diferentes módulos em vez de ter que instalar gldea2 de gldea1 e assim por diante.

Não é a intenção do GCompris se tornar uma linguagem de programação nova. Este é implementado em C e os módulos também são. Módulos e GCompris se uniram dinamicamente em tempo de execução(plugins) o que torna possível distribuir boards

novas independentemente do centro do GCompris.

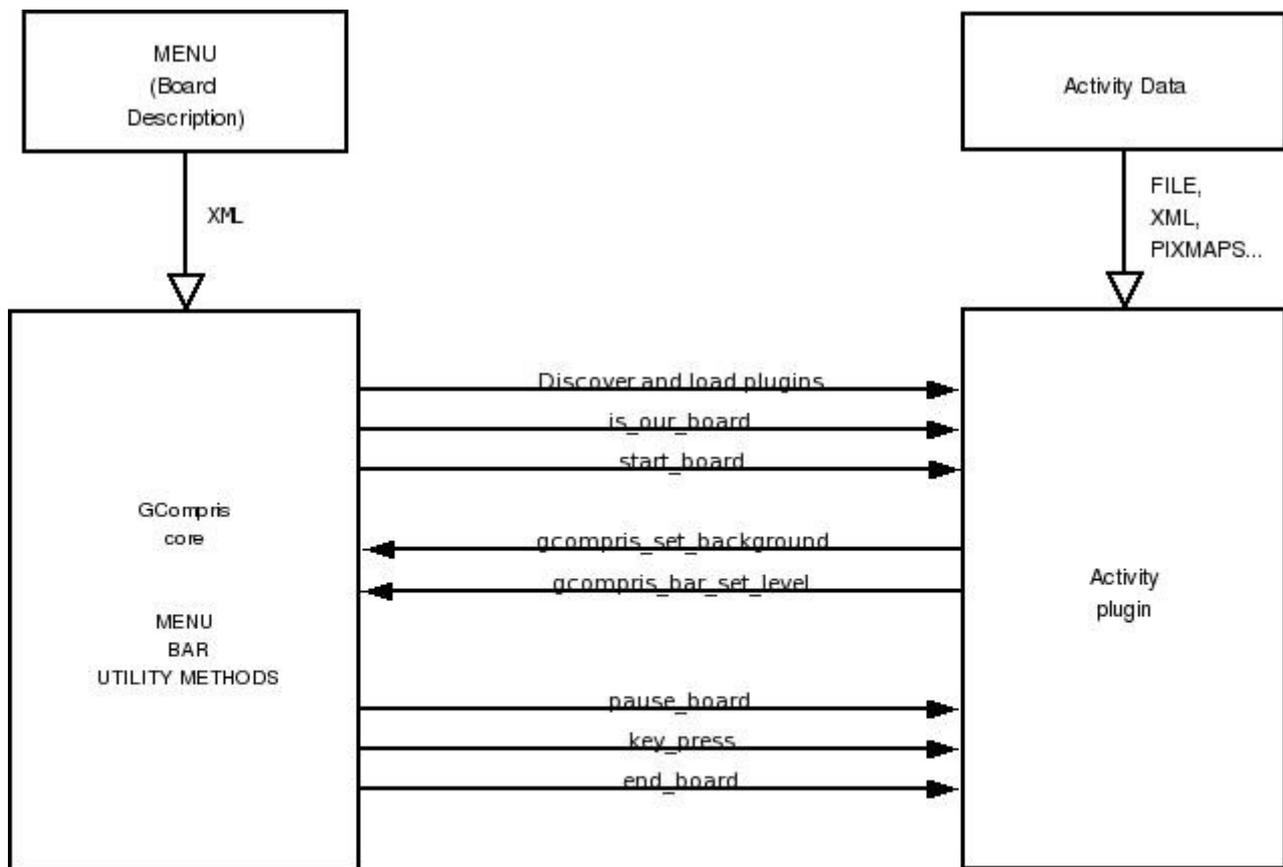
A implementação atual está baseado no widget do Gnome Canvas (veja documentações de [<http://www.gnome.org>]). O Gnome Canvas provê um alto nível de widget nivelado. Deste modo, os programadores ficam livres da manipulação de bitmap, quando fazem aplicações gráficas. Na implementação atual, não é possível escrever módulos que usam outro widget. Se parecer que o Canvas não é poderoso o suficiente, GCompris terá que ser modificado para apoiar outros widgets.

Outra abordagem é ter módulos mais inteligentes que podem ler arquivos de dados e se comportar adequadamente. Neste caso nós podemos prever um editor de módulos do GCompris que deixará o não programador acrescentar conteúdo ao GCompris. Como o objetivo do GCompris são definitivamente os professores e pedagogos, esta opção parece muito promissora. Note o seguinte caminho: a forma do módulo já recebe seus dados por um arquivo de XML.

4.2 Como acrescentar novos módulos no Gcompris

4.2.1 A figura completa

Aqui está uma representação gráfica de como os módulos se comunicam ao núcleo do GCompris através de chamadas de funções.



**Overview of a typical game sequence
between GCompris and an activity plugin**

4.2.2 GCompris API

GCompris provê várias utilidades para desenvolver os módulos. Estes serviços são descritos no arquivo *gcompris/gcompris.h*. Eis alguns:

- *GnomeCanvasItem *gcompris_set_background(GnomeCanvasGroup *parent, gchar *file);*
Seta a imagem de fundo para seu jogo. Você pode obter ajuda de *gameutil.h* para carregar a imagem.
- *void gcompris_bar_set_level (GcomprisGame *gcomprisGame);*
O usuário necessitou e teve que ir para o próximo nível. Antes de chamar isto, *gcomprisGame->level* deve ser atualizado pelo módulo.

- *void gcompris_bar_set_timer (guint time);*

O cronômetro é uma barra de progresso que normalmente é usada para informar ao usuário se ele/ela está no módulo. Este valor deve estar no alcance 0 para 'maxtimer' determinado com *bar_set_maxtimer ()*.

- *void bar_set_maxtimer (guint time);*

Seta o valor máximo da barra de progresso.

Assim que você criar um módulo, você tem que incluir *gcompris/plugin.h* para ter acesso a esses métodos (isto já é incluído em *gcompris/gcompris.h*).

plugin.h também define a estrutura de BoardPlugin. Este é o coração de GCompris. Cada módulo cria seu próprio BoardPlugin e devolve isto ao GCompris quando este chama o ponto de entrada de *get_bplugin_info ()*:

```
BoardPlugin* *get_bplugin_info ();
```

O BoardPlugin retornado deve conter chamadas de volta a seus métodos de implementação.

A estrutura de BoardPlugin tem que prover callbaks pelo menos para:

- *void *GcomprisStartBoard (GcomprisBoard *gcomprisBoard);*

Isto é chamado pelo GCompris quando o módulo em questão for pedido pelo usuário.

- *void *GcomprisEndBoard (void);*

Isto é chamado pelo GCompris quando o usuário clica em 'end' no menu de controle.

Opcionalmente, a estrutura do jogo do GCompris pode conter callbacks para:

- *gint *GcomprisKeyPress (guint keyval);*

Se o módulo quiser receber keypress de usuário, tem que prover este método de callback. O keyval não é filtrado e é devolvido como retorno do Gnome. Confira a documentação do Gnome para entender como manipular

isto.

- `void *GcomprisOK (void);`
Este método é chamado quando o usuário clica em 'OK', botão na barra de controle. É uma boa prática para fazer o enter, e retornar a chave que se comporta como o botão 'ok', mas nem sempre pode ser apropriado. Por isto, é a responsabilidade da implementação do módulo fazer isto.
- `void *GcomprisSetLevel (guint level);`
Isto é chamado quando o nível é mudado pelo usuário. Você pode provê o número máximo do nível em `gcomprisGame-> maxlevel`.

4.2.3 Exemplo de como duplicar um módulo já existente

Isto descreve todos os passos exigidos para acrescentar um novo módulo ao GCompris.

Como exemplo, descreve-se aqui como o Bruno Croudin criou o módulo do relógio.

1. Copie um jogo existente (perto da sua board nova). `cp algebra.c clockgame.c`.
2. Edite `src/boards/Makefile.am` e acrescente o arquivo na lista de `lib_LTLIBRARIES`. Então acrescente ao fim as entradas para `libclockgame_la_LDFLAGS`, `libclockgame_la_LIBADD` e `libclockgame_la_SOURCES`. Olhe para os exemplos em `src/boards/Makefile.am` para ver como fazer isso.
3. Em `clockgame.c` substitua todos os 'algebra' por 'clockgame'.
Em `clockgame.c` sete a própria informação para o jogo novo:
`if(g_strcasecmp(gcomprisBoard->type, "clockgame")==0)`

Você deve então criar ou copiar a descrição da board ao local apropriado.

```
cp boards/algebra.xml boards/clockgame.xml
```

Então edite *clockgame.xml* com sua própria descrição e ícone. Sete o arquivo de ícone no local determinado em *clockgame.xml*.

4. Em *boards/Makefile.am* acrescente seu arquivo na lista. Respeite a ordem alfanumérica. Mesmo passo em *po/POTFILES.in*.
5. Deverá compilar tudo certo.
6. '*make install*' é obrigatório ou o ícone não será achado em tempo de execução.
7. Agora, executando o GCompris deverá aparecer o novo ícone e você poderá jogar o jogo que você copiou. Agora está na hora de codificar o módulo em *clockgame.c*.
8. Atualize os arquivos de po:

```
cd gcompris/po
edit POTFILES.in and add src/clockgame.c)
make
make update-po
```
9. Agora você pode acrescentar a tradução nos arquivos de po.
10. 'Digite make' e depois 'make install' deverá compilar e instalar sua tradução.

4.2.4 Solução Alternativa

Mesmo seguindo todos os passos descritos na seção anterior, muitas pessoas encontram dificuldades em duplicar um módulo para começar a desenvolver a sua própria idéia. Uma solução alternativa, é sobrescrever algum módulo já existente e depois procurar ajuda na comunidade GCompris para de fato acrescentar o novo módulo e disponibilizá-los para todos os interessados.

Justamente neste ponto é que podemos comprovar o poder da comunidade do software livre. Na comunidade brasileira do GCompris, o “GCompris Brasil” e no projeto “Seeduca”, através de listas de discussões, encontra-se todo o apoio e atenção para acrescentar o novo módulo, assim como para quaisquer outras dúvidas.

E é justamente por esse apoio da comunidade, que o GCompris, e os demais softwares livres evoluem tão rapidamente e de maneira segura.