

Manual do script kdesrc-build

Michael Pyne

Carlos Woelz

Tradução: Marcus Gama

Tradução: André Marcelo Alvarenga



Manual do script kdesrc-build

Conteúdo

1	Introdução	8
1.1	Uma breve introdução ao kdesrc-build	8
1.1.1	O que é o kdesrc-build?	8
1.1.2	O funcionamento do kdesrc-build 'em resumo'	8
1.2	Visão Geral da Documentação	8
2	Começando	9
2.1	Preparando o Sistema para Compilar o KDE	9
2.1.1	Configurar uma nova conta de usuário	9
2.1.2	Certifique-se de que o seu sistema está pronto para compilar o código do KDE	9
2.1.3	Configurar o kdesrc-build	11
2.1.3.1	Instalar o kdesrc-build	11
2.1.3.2	Preparar o arquivo de configuração	11
2.2	Definindo os Dados de Configuração	12
2.3	Organização e seleção dos módulos	12
2.3.1	Organização do Software do KDE	12
2.3.2	Selecionar os módulos a compilar	13
2.3.3	Conjuntos de Módulos	14
2.3.4	Procurar automaticamente módulos da base de dados oficial do KDE	15
2.3.5	Filtrar módulos do projeto do KDE	16
2.4	Usando o script kdesrc-build	17
2.5	Configurando o ambiente para executar o seu ambiente de trabalho Plasma do KDE	19
2.5.1	Instalar automaticamente um driver de autenticação	19
2.5.1.1	Adicionar o suporte da xsession para as distribuições	19
2.5.1.2	Adicionar manualmente o suporte ao xsession	19
2.5.2	Configurar manualmente o ambiente	20
3	Funcionalidades do Programa	21
3.1	Visão Geral das Funcionalidades	21
3.2	Registro de compilação do kdesrc-build	23
3.2.1	Visão geral do registro	23
3.2.1.1	Disposição da pasta de registro	23

4	Configurando o kdesrc-build	24
4.1	Visão geral da configuração do kdesrc-build	24
4.1.1	Layout do arquivo de configuração	24
4.1.1.1	Configuração global	24
4.1.1.2	Configuração do módulo	24
4.1.1.3	Módulos de 'options'	25
4.1.2	Incluindo outros arquivos de configuração	26
4.1.3	Opções de configuração comumente usadas	26
4.2	Tabela das opções de configuração disponíveis	27
5	Opções da Linha de Comando e Variáveis de Ambiente	58
5.1	Uso da linha de comando	58
5.1.1	Opções de linha de comando comumente usadas	58
5.1.2	Especificando os módulos a compilar	59
5.2	Variáveis de Ambiente Suportadas	59
5.3	Parâmetros de linha de comando suportados	59
6	Usando o kdesrc-build	66
6.1	Prefácio	66
6.2	Funcionalidades básicas do kdesrc-build	66
6.2.1	Suporte para o 'qt'	66
6.2.2	Opções normais adicionadas pelo kdesrc-build	67
6.2.3	Mudando a prioridade de compilação do kdesrc-build	67
6.2.4	Instalação como superusuário	68
6.2.5	Mostrando o progresso da compilação de um módulo	68
6.3	Funcionalidades avançadas	69
6.3.1	Compilar parcialmente um módulo	69
6.3.1.1	Extrair partes de um módulo	69
6.3.1.2	Removendo pastas de uma compilação	69
6.3.2	Suporte de marcas e ramificações do kdesrc-build	70
6.3.2.1	O que são ramificações e marcas?	70
6.3.2.2	Como usar as ramificações e marcas	70
6.3.2.3	Opções avançadas de suporte a ramificações	71
6.3.3	Como o kdesrc-build tenta garantir uma compilação com sucesso	71
6.3.3.1	Recompilações automáticas	71
6.3.3.2	Recompilando manualmente um módulo	72
6.3.4	Mudando a configuração das variáveis de ambiente	72
6.3.5	Prosseguir com as compilações	73
6.3.5.1	Prosseguir uma compilação falhada ou cancelada	73
6.3.5.2	Ignorando os módulos numa compilação	73

Manual do script kdesrc-build

6.3.6	Mudando as opções na linha de comando	73
6.3.6.1	Modificando as opções globais	73
6.3.6.2	Mudando as opções do módulo	74
6.4	Funcionalidades para os programadores do KDE	74
6.4.1	Verificações do Agente do SSH	74
6.5	Outras funcionalidades do kdesrc-build	74
6.5.1	Mudando a quantidade de mensagens do resultado do kdesrc-build	74
6.5.2	Resultado colorido	75
6.5.3	Removendo as pastas desnecessárias após uma compilação	75
7	CMake, o sistema de compilação do KDE 4	77
7.1	Introdução ao CMake	77
8	Créditos e licença	78
A	Organização dos módulos e do código-fonte do KDE	79
A.1	O 'Módulo'	79
A.1.1	Módulos individuais	79
A.1.2	Grupos de módulos relacionados	79
A.1.3	'Grupos de ramificações' do módulo	80
B	Procedimentos de configuração do perfil obsoletos	82
B.1	Configurar um perfil de autenticação do KDE	82
B.1.1	Alterando a configuração do seu perfil inicial	82
B.1.2	Iniciando o KDE	83

Lista de Tabelas

4.1 Tabela de Opções	57
--------------------------------	----

Resumo

O kdesrc-build é um script que compila e instala programas do KDE diretamente a partir do repositório de código-fonte do projeto do KDE.

Capítulo 1

Introdução

1.1 Uma breve introdução ao kdesrc-build

1.1.1 O que é o kdesrc-build?

O kdesrc-build é um script que ajuda os usuários a instalar programas do KDE a partir dos seus repositórios de código do [Subversion](#) e do [Git](#).

Além de simplesmente instalar aplicações do KDE, o programa também pode ser usado para atualizar os programas do KDE instalados. Isto permite mantê-lo atualizado em relação ao desenvolvimento do KDE.

O programa kdesrc-build pode ser usado para manter um módulo individual único ou um ambiente de trabalho completo do KDE, dependendo de como estiver configurado.

1.1.2 O funcionamento do kdesrc-build ‘em resumo’

O kdesrc-build funciona intencionalmente com as mesmas ferramentas que estão disponíveis para o usuário pela linha de comando. Quando o kdesrc-build é executado, é seguida a sequência abaixo:

1. O kdesrc-build usa uma [linha de comando](#) e um [arquivo de configuração](#) para controlar os módulos que são compilados, onde são instalados, etc.
2. O kdesrc-build executa uma atualização de código para cada [módulo](#). A atualização prossegue até que todos os módulos tenham sido atualizados. Os módulos que não consigam ser atualizados normalmente não interrompem o processo – você será notificado no fim sobre quais os módulos não foram atualizados.

1.2 Visão Geral da Documentação

Este guia é uma visão geral para descrever os seguintes aspectos de operação do kdesrc-build:

- Uma [visão geral](#) dos passos necessários para iniciar.
- [Funcionalidades](#) notáveis.
- A sintaxe do [arquivo de configuração](#) e suas opções.
- As [opções de linha de comando](#).

Também são documentados aqui os passos que você deve realizar usando outras ferramentas (em outras palavras, passos que não são automaticamente realizados pelo kdesrc-build).

Capítulo 2

Começando

Neste capítulo, será apresentada a forma de utilização do kdesrc-build para obter os módulos do repositório do KDE e compilá-los. Nós daremos também uma explicação básica da estrutura do código-fonte do KDE e os passos que você terá que efetuar antes de executar o programa.

Todos os tópicos presentes neste capítulo são cobertos com ainda mais detalhes no artigo [Compilando o KDE 4 a partir do Código-Fonte](#), na [Página TechBase do KDE](#). Se você estiver compilando o KDE pela primeira vez, é uma boa ideia lê-lo ou consultá-lo como fonte de referência. Você irá encontrar informações detalhadas sobre as ferramentas de empacotamento e os requisitos, os problemas de compilação comuns e as estratégias e informações sobre a execução da instalação do seu novo KDE.

2.1 Preparando o Sistema para Compilar o KDE

2.1.1 Configurar uma nova conta de usuário

É recomendável que você use uma conta de usuário diferente para compilar, instalar e executar seus programas KDE, uma vez que menos permissões são necessárias e para evitar a interferência com os pacotes de sua distribuição. Se você já possui pacotes do KDE instalados, a melhor opção seria criar um usuário diferente (dedicado) para compilar e executar o novo KDE.

DICA

Deixar seu sistema KDE intocado também permite que você volte ao estado anterior em uma emergência caso um KDE compilado seja instável por qualquer razão.

Mais tarde, você poderá fazer uma instalação de sistema se desejar. Este documento não cobre uma instalação a nível do sistema. Se estiver efetuando uma instalação a nível do sistema, provavelmente já deverá saber o que está fazendo. Caso contrário, recomenda-se consultar a documentação ou páginas de ajuda da sua distribuição, para saber como preparar e usar a instalação do sistema corretamente.

2.1.2 Certifique-se de que o seu sistema está pronto para compilar o código do KDE

Antes de usar o programa kdesrc-build (ou qualquer outra estratégia de compilação), você deverá instalar as ferramentas de desenvolvimento e bibliotecas necessárias para o KDE. A lista

quase completa das ferramentas necessárias pode ser encontrada na página [KDE TechBase Build Requirements](#).

Aqui está uma lista das coisas que você irá precisar:

- Você precisa do CMake. A versão necessária poderá variar de acordo com a versão do KDE 4 que estiver compilando; veja a TechBase para detalhes, mas uma boa dica é ter instalada a versão mais recente. O CMake é o programa usado pelo kdesrc-build para lidar com a configuração e compilação da maioria dos aplicativos do KDE.
- É também necessário instalar os programas-cliente usados para extrair o código-fonte do KDE. Isto significa que você precisa pelo menos o seguinte:
 - O Subversion, que costumava ser o único sistema de controle de versões usado, e que ainda continua a ser utilizado para alguns módulos com arquivos de dados grandes. Você pode verificar se possui instalado digitando `svn --version`.
 - Você irá precisar do gerenciador de controle de versões Git instalado, para os projetos do KDE baseados em Git. .
 - Ainda que não seja necessário, o gerenciador de controle de versões Bazaar é usado para um único módulo (libdbusmenu-qt) que é necessário para as bibliotecas do KDE. A maioria dos usuários poderão instalar esta biblioteca através dos pacotes da sua distribuição, mas o kdesrc-build suporta a compilação do mesmo, se assim desejar. Entretanto, para compilar o libdbusmenu-qt, você precisa ter o Bazaar instalado.
- Você precisará do ambiente de desenvolvimento C++ completo. O GCC 4.6 ou superior é recomendado.
- Finalmente, você precisa de uma ferramenta 'make'. O GNU Make é o recomendado e deve estar disponível no seu gerenciador de pacotes. Após o `cmake` ter sido executado pelo kdesrc-build, o `make` lida com o processo de compilação, por isto ele é obrigatório.

NOTA

A maioria das distribuições e sistemas operacionais inclui um método para instalar facilmente as ferramentas de desenvolvimento necessárias. Consulte a página [Começando](#) da TechBase, na sua seção 'Pacotes necessários para sua Distribuição' para ver se estas instruções já estão disponíveis.

Uma exceção às bibliotecas necessárias é a biblioteca Qt™. O kdesrc-build irá instalar normalmente uma cópia do Qt™, quer já a tenha instalado ou não. Assim, não é necessário possuí-la. Se você não quiser usar a cópia do Qt™, terá que fazer o seguinte:

- Certifique-se de que removeu o módulo 'qt' do seu [arquivo de configuração](#), uma vez que você não irá precisar dele e, se possuí-lo, isso só vai adicionar um tempo adicional à sua compilação.
- Mude a configuração da opção `qtdir` no seu [arquivo de configuração](#), de modo a apontar para o seu Qt™ de sistema. A localização do seu Qt™ poderá ser encontrada com o comando `qmake -queryQT_INSTALL_PREFIX`.

NOTA

O comando `qmake` poder ser chamado de `qmake4` ou `qmake-qt4` na sua distribuição.

- Se você não tiver já o Qt™ instalado, instale-o, incluindo os pacotes `-dev` ou `-devel` necessários. Você deverá ter pelo menos o Qt™ 4.7 se for compilar o KDE 4.

IMPORTANTE

Alguns destes pacotes estão divididos em bibliotecas, programas ou utilitários e pacotes de desenvolvimento. Você precisará, pelo menos, do programa ou biblioteca e do seu pacote de desenvolvimento. As bibliotecas que necessita irão variar de acordo com os módulos que pretende compilar, uma vez que cada módulo possui os seus próprios requisitos. O [TechBase do KDE](#) tem mais detalhes sobre as ferramentas e técnicas específicas usadas para instalar e encontrar os programas necessários.

2.1.3 Configurar o kdesrc-build

2.1.3.1 Instalar o kdesrc-build

Você provavelmente já tem uma versão do programa kdesrc-build instalada no seu sistema. Contudo, se não tiver, poderá obtê-lo na [página do kdesrc-build](#) ou poderá encontrá-lo no seu local original, no repositório de código do KDE.

DICA

Se você usar um kdesrc-build mais recente baixado de sua página Web, você deve lembrar-se de executar o script kdesrc-build que você baixou. Você pode usar a opção `--version` no kdesrc-build como uma maneira rápida de verificar isto.

1. Para baixar o kdesrc-build a partir de sua página, basta ir para a [página do kdesrc-build](#) e transferir a versão mais recente. A versão está dentro de um pacote TAR comprimido, que você pode extrair com o Ark ou o `tar`. O conteúdo do pacote inclui o programa kdesrc-build atual, um arquivo de configuração de exemplo (`kdesrc-buildrc-sample`) e um programa de configuração rápida.
2. Ou você poderá obter o kdesrc-build a partir do seu repositório de origem, executando o comando:

```
$ git clone git://anongit.kde.org/kdesrc-build ~/kdesrc-build
```

Substitua `~/kdesrc-build` pela pasta aonde você deseja instalar.

Independentemente da técnica que usar, você deverá certificar-se de que o arquivo `kdesrc-build` é executável. Por conveniência, deverá colocá-lo em uma pasta que esteja definida na variável de ambiente `PATH`, caso contrário, poderá obter mensagens que indiquem que o comando não foi encontrado ou que está rodando uma versão errada por engano.

2.1.3.2 Preparar o arquivo de configuração

O kdesrc-build usa um [arquivo de configuração](#) (localizado em `~/kdesrc-buildrc`) para controlar que módulos compilar, onde eles serão instalados, etc.

Você pode usar um programa incluído com o kdesrc-build, chamado `kdesrc-build-setup` para preparar uma configuração básica do kdesrc-build. Você pode então editar o `~/kdesrc-buildrc` a partir desta configuração e então fazer as mudanças que desejar.

O `kdesrc-build-setup` em si é executado a partir de um terminal (ao invés de usar uma interface gráfica), da mesma maneira que o kdesrc-build, assim você pode usá-lo mesmo se a interface gráfica ainda não estiver disponível.

DICA

Você pode usar a configuração de exemplo `kdesrc-buildrc-sample` para obter explicações sobre as diversas opções disponíveis.

Você poderá encontrar mais informações sobre a sintaxe do [arquivo de configuração](#) em Seção 2.2 e em capítulo 4.

2.2 Definindo os Dados de Configuração

Para usar o `kdesrc-build`, você deverá ter um arquivo na sua pasta pessoal chamado `.kdesrc-buildrc`, que define as opções gerais e os módulos que deseja baixar e compilar.

NOTA

É possível usar arquivos de configuração diferentes para o `kdesrc-build`, que estão descritos em capítulo 4. Se você precisar usar várias configurações, veja esta seção. Aqui, nós iremos considerar que a configuração está armazenada em `~/.kdesrc-buildrc`.

A forma mais simples de prosseguir é usar o arquivo `kdesrc-buildrc-sample` como modelo, definir as opções globais de acordo com as suas necessidades e a lista dos módulos que deseja compilar.

A configuração padrão já deverá ser apropriada para efetuar uma compilação do KDE. Algumas opções que você poderá querer alterar incluem:

- [use-stable-kde](#) para alterar a versão padrão dos módulos do KDE a compilar. Por padrão, o `kdesrc-build` irá compilar a última versão do KDE. Se quiser compilar a última versão estável do KDE, em vez de usar os pacotes da sua distribuição, iria usar esta opção como `true`.

DICA

Lembre-se que esta opção baseia-se em informações disponíveis na base de dados do projeto KDE, assim esta funcionalidade só funciona nesses módulos. Veja também em Seção 2.3.4.

- [kdedir](#), que muda a pasta de destino para onde o KDE será instalado. Ela é por padrão a `~/kde`, que é uma instalação para um único usuário.
- [qtdir](#), que controla o local da instalação do Qt™ a ser usado. Por padrão, usa um Qt™ compilado pelo `kdesrc-build`, usando o módulo especial 'qt' do repositório e o último código-fonte disponível. (`~/kdesrc/build/qt`).

NOTA

Isto também controla onde instalar o qt.

- [svn-server](#), que seleciona a URL de onde baixar o código. Isto é útil se você for um programador com uma [conta Subversion para o KDE](#).
- Você provavelmente desejará selecionar módulos diferentes para compilar, os quais estão descritos em Seção 2.3.2.

2.3 Organização e seleção dos módulos

2.3.1 Organização do Software do KDE

As aplicações do KDE estão divididas em diversos componentes, muitos dos quais poderão ser compilados pelo `kdesrc-build`. A compreensão desta organização irá ajudá-lo a selecionar adequadamente os módulos de aplicações que deseja compilar.

1. No nível inferior vem a biblioteca Qt™ da Digia, que é uma biblioteca multi-plataforma e muito poderosa. O KDE baseia-se no Qt™ e algumas das bibliotecas fora do KDE que são necessárias também se baseiam no Qt™. O `kdesrc-build` poderá compilar o Qt™ ou usar o que já está instalado no seu sistema, caso seja uma versão recente o suficiente.

2. No topo do Qt™ estão as bibliotecas necessárias para o sistema KDE funcionar. Algumas destas bibliotecas não são consideradas como parte do KDE propriamente dito, devido à sua natureza genérica, mas são ainda essenciais para a Plataforma do KDE. Estas bibliotecas costumavam estar reunidas sob um único módulo `kdesupport`.

NOTA

Desde a Plataforma do KDE 4.6, muitas bibliotecas do módulo do 'kdesupport' estão sendo migradas para o git.kde.org, ainda que continuem a não fazer parte da Plataforma.

3. Sobre essas bibliotecas essenciais, vem a Plataforma do KDE. Estas são as bibliotecas que são necessárias para que as aplicações do KDE funcionem. Todavia, não é fornecido um ambiente de trabalho completo com esta plataforma.

Para o `kdesrc-build`, a camada da Plataforma consiste nos módulos `kdelibs`, `kdepimlibs` e `kde-runtime`.

4. No topo da Plataforma, vêm diversas coisas diferentes:

- Aplicações de 'terceiros'. Estas são aplicações que usam a Plataforma do KDE mas não são da autoria ou estão associadas ao projeto KDE.
- Um ambiente e respectiva 'área de trabalho' completos. Isto é o que os usuários normalmente veem quando 'se autenticarem no KDE'. Isto é fornecido pelo Ambiente do Plasma, majoritariamente em `kde-workspace`.
- A Compilação de Software do KDE (SC). Esta é uma coleção de aplicações úteis que vem incluída na Plataforma e no Ambiente de Trabalho Plasma, agrupada em módulos individuais. Estes são os módulos que têm nomes que começam por `kde`. Por exemplo, o `kdepim` é um componente da Compilação de Software que contém aplicações de e-mail, leitura de notícias, organização e calendários, etc., enquanto o `kdegames` contém uma coletânea de jogos de alta qualidade para lhe ocupar o tempo.
- Finalmente, existe uma coleção de aplicações (também agrupada em módulos) cujo desenvolvimento é suportado pelos recursos do KDE (como as traduções, o controle de código, o registro de erros, etc.) mas que não é lançada pelo KDE ou considerada como parte da Compilação de Software. Estes módulos são conhecidos como 'Extragear' e têm nomes do tipo `extragear/network`. Como no `kdesupport`, algumas dessas aplicações do Extragear estão migrando para o git.kde.org.

2.3.2 Selecionar os módulos a compilar

A seleção de qual dos módulos possíveis a compilar é controlada pelo [arquivo de configuração](#). Depois da seção `global`, existe uma lista com os módulos a compilar, delimitada pelas linhas `'module ... end module'`. Aparece um item de exemplo para um módulo em Exemplo 2.1.

Example 2.1 Item de exemplo de módulo no arquivo de configuração

```
module nome-módulo
  # As opções deste módulo vão aqui, por exemplo:
  make-options -j4 # Rodar 4 compilações de cada vez
end module
```

DICA

É possível declarar um módulo sem opções. De fato, a maioria dos seus módulos serão provavelmente declarados desta forma.

O `kdesrc-build` só compila os módulos que tiver apresentado no seu arquivo de configuração. Além disso, os módulos são compilados na ordem indicada no arquivo de configuração. Por essa razão, você deverá certificar-se de que a ordem dos módulos no seu arquivo de configuração é consistente com a organização indicada em Seção 2.3.1.

Existe um arquivo de exemplo, que vem com o `kdesrc-build`, chamado `kdesrc-buildrc-samp1e`. Recomenda-se que copie este arquivo para um outro chamado `~/.kdesrc-buildrc` (*Repare no ponto inicial à frente do 'kdesrc-buildrc'!*). Depois disso, edite o arquivo novo para ajustar as opções pré-definidas a seu gosto. (Cada opção está descrita com mais detalhes no capítulo 4). Os módulos padrão deverão ser suficientes para garantir a instalação de um KDE relativamente completo; contudo, poderá remover muitos dos módulos que aparecem após o `kdebase`, caso queira poupar espaço em disco ou tempo de compilação.

2.3.3 Conjuntos de Módulos

O `kdesrc-build` é normalmente capaz de adivinhar onde transferir o código-fonte para um determinado módulo de forma bastante simples, usando a sua configuração do `svn-server` e o nome do módulo para criar uma única URL do Subversion, que descreve exatamente de onde transferir o código-fonte.

Com a passagem para o Git, muitos módulos maiores do Subversion foram subdivididos no processo, não existindo garantia de como encontrar um módulo com base no seu módulo. Devido a isto, foi desenvolvido um conceito chamado 'conjuntos de módulos' para o `kdesrc-build` 1.12.1.

Se usar um conjunto de módulos, poderá declarar rapidamente muitos módulos do Git para serem transferidos e compilados, como se tivesse definido uma declaração de módulo em separado para cada um. A opção `repository` é tratada de forma especial para definir de onde foi transferido cada módulo, sendo todas as outras opções existentes no conjunto de módulos copiadas para cada módulo gerado desta forma.

Example 2.2 Usar os conjuntos de módulos

```
global
  git-repository-base kde-git kde:
end global

module qt
  # Opções removidas para resumir
end module

module-set kde-support-libs
  repository kde-git
  use-modules automoc attica akonadi
end module-set

# Outros módulos conforme necessário...
module kdesupport
end module
```

Em Exemplo 2.2, aparece um breve conjunto de módulos. Quando o `kdesrc-build` encontrar este conjunto de módulos, irá atuar como se, para cada módulo indicado em `use-modules`, fosse declarado um módulo individual com o seu `repository` igual à opção `repository` do conjunto, seguida do nome do módulo indicado.

Além disso, poderá passar outras opções num conjunto de módulos, as quais são copiadas para os novos módulos que forem criados desta forma. Ao usar os conjuntos de módulos, é possível declarar rapidamente vários módulos do Git que se baseiam todos na mesma URL de repositório.

Além disso, desde o kdesrc-build 1.13, é possível atribuir um nome aos conjuntos de módulos (como demonstrado no exemplo), o que lhe permite referenciar diretamente o grupo inteiro de módulos a partir da linha de comando.

NOTA

Os conjuntos de módulos são usados no suporte para transferências de módulos a partir da base de dados de módulos do projects.kde.org para o KDE. Veja também em Seção 2.3.4.

Os conjuntos de módulos usam as opções

`git-repository-base`
`use-modules`

2.3.4 Procurar automaticamente módulos da base de dados oficial do KDE

Com a migração do código-fonte do KDE para se migrar para o git.kde.org, houve um crescimento explosivo no número de módulos (por exemplo, um único módulo do Subversion chamado `kdegraphics` transforma-se em 16 módulos diferentes no Git).

Isto foi feito principalmente por cada módulo do Git conter o histórico completo do projeto (isto desperdiça de fato menos espaço em disco do que acontece na maioria dos repositórios do KDE, uma vez que o Git é extremamente eficiente gravando repositórios).

O KDE permite agrupar os repositórios do Git em coleções de módulos relacionados (p.ex., o 'kdegraphics'). Estes módulos poderão por sua vez ser agrupados (p.ex., Compilação de Software do KDE). O Git não reconhece estes agrupamentos, mas o kdesrc-build pode ser configurado para lidar com estes grupos.

A forma como isto é feito é através da utilização de [conjuntos de módulos](#). Em vez de usar um repositório específico `git://` ou o nome de um repositório criado pelo `git-repository-base`, é usado o nome de repositório especial `'kde-projects'`.

O kdesrc-build irá reconhecer que o repositório `kde-projects` precisa de tratamento especial e ajustar o processo de compilação em conformidade. Entre outras coisas, o kdesrc-build irá:

- Obter a última base de dados de módulos do projects.kde.org.
- Tentar procurar um módulo com o nome indicado na opção `use-modules` do conjunto de módulos.
- Para cada módulo que encontrar, o kdesrc-build irá ver se existe uma opção do repositório para esse módulo na base de dados. Se existir um repositório, o kdesrc-build irá usá-la automaticamente para transferir ou atualizar o código-fonte. Se não estiver definido nenhum, o kdesrc-build irá tratar desse módulo como um grupo, tentando incluir todos os repositórios de código do Git que encontrar nesse grupo.

NOTA

Na base de dados atual, não só alguns grupos de módulos têm uma coleção de módulos, como *também* declaram o seu próprio repositório de Git. Nestas situações, o kdesrc-build irá preferir o repositório de Git do grupo em vez de incluir os repositórios dos filhos.

O seguinte exemplo demonstra como usar a base de dados de módulos do KDE para instalar a biblioteca multimídia Phonon.

Manual do script kdesrc-build

```
module-set media-support
  # Esta opção deverá ser 'kde-projects' para usar a base de dados de ↔
  módulos.
  repository kde-projects

  # Esta opção escolhe quais os módulos procurar na base de dados.
  use-modules phonon/phonon phonon-gstreamer phonon-vlc
end module-set
```

DICA

O `phonon/phonon` é usado uma vez que (com o projeto de banco de dados atual) o `kdesrc-build` caso contrário teria que decidir entre o grupo de projetos chamado 'phonon' ou o projeto individual chamado 'phonon'. Atualmente, o `kdesrc-build` pegaria o mais antigo, que compilaria mas infraestruturas do que o necessário.

O exemplo a seguir será talvez mais realista e demonstra uma funcionalidade que só está disponível na base de dados de módulos do KDE: a compilação de todos os aplicativos gráficos do KDE com apenas uma única declaração.

```
module-set kdegraphics
  # Esta opção deverá ser igual a 'kde-projects' para usar a base de ↔
  dados de módulos
  repository kde-projects

  # Esta opção escolhe os módulos a pesquisar na base de dados.
  use-modules kdegraphics/libs kdegraphics/*
end module-set
```

Existem duas importantes habilidades demonstradas aqui:

1. O `kdesrc-build` permite que você especifique módulos que são descendentes de um determinado módulo, sem compilar o módulo pai, usando a sintaxe `nome-módulo /*`. Isto é necessário de fato neste caso uma vez que o módulo base, o `kdegraphics`, está marcado como inativo, de modo que ele não seja acidentalmente compilado com seus módulos-filho. Especificar os módulos descendentes permite ao `kdesrc-build` pular o módulo desativado.
2. O `kdesrc-build` também não adicionará um módulo fornecido a uma lista de compilação mais de uma vez. Isto nos permite definir manualmente que `okdegraphics/libs` seja compilado primeiro, antes do restante do `kdegraphics`, sem que o `kdegraphics/libs` seja compilado duas vezes.

NOTA

Convém apontar que o `kdesrc-build` irá tentar compilar os módulos pela ordem correta (por exemplo, se apenas o `kdegraphics/*` tiver sido incluído acima), mas isso depende da atualização de outros bancos de dados. Se necessário, você pode listar os módulos na ordem correta manualmente usando a técnica descrita acima.

2.3.5 Filtrar módulos do projeto do KDE

Você pode decidir que gostaria de compilar todos os programas dentro de um agrupamento de módulos do KDE, *exceto* um determinado programa.

Por exemplo, o grupo `kdeutils` inclui um programa chamado `kremotecontrol`. Se o seu computador não tiver o hardware adequado para receber os sinais enviados por controles remotos, você pode optar por não baixar, compilar e instalar o `kremotecontrol` todas as vezes que atualizar o `kdeutils`.

Desde o `kdesrc-build` 1.16, você pode conseguir isto usando a opção de configuração [ignore-modules](#).

Example 2.3 Exemplo para ignorar um módulo de um grupo do projeto KDE

```
module-set utils
  repository kde-projects

  # Esta opção escolhe quais os módulos procurar no banco de dados.
  use-modules kdeutils
  # Esta opção "subtrai" os módulos escolhidos pela opção 'use-modules' ←
  acima.
  ignore-modules kremotecontrol
end module-set

module-set graphics
  repository kde-projects

  # Esta opção escolhe os módulos a pesquisar no banco de dados.
  use-modules extragear/graphics

  # Esta opção "retira" os módulos dos escolhidos acima pelo 'use-modules ←
  '.
  # Nesse caso, *tanto* o 'extragear/graphics/kipi-plugins' quanto o
  # extragear/graphics/kipi-plugins/kipi-plugins-docs são ignorados
  ignore-modules extragear/graphics/kipi-plugins
end module-set
```

2.4 Usando o script kdesrc-build

Agora você está pronto para executar o programa. A partir de uma janela de terminal, autentique-se com o usuário que usará para compilar o KDE e execute o programa:

```
% kdesrc-build
```

Depois disso, você deverá ver um resultado semelhante ao do Exemplo 2.4:

Example 2.4 Resultado de exemplo da compilação de um único módulo

```
% kdesrc-build kdelibs
O programa iniciou o processamento em Qua Dez 22 13:21:45 2010
<<< Processo de Compilação >>>
Compilando o kdelibs (1/1)
    Esperando a atualização do código-fonte.
    Atualização do código completa para o kdelibs: 48 arquivos afetados ←
    .
    Verificando os conflitos de código...
    Compilando...
    A compilação foi bem sucedida ao fim de 13 minutos e 6 segundos.
    Instalando o kdelibs.
    O tempo global do kdelibs foi 13 minutos e 53 segundos.

<<< Compilação Completa >>>

<<< PACOTES COMPILADOS COM SUCESSO >>>
kdelibs

O programa terminou o processamento em Qua Dez 22 13:35:38 2010
Os seus registros são gravados em /home/kde-src/log-kdesrc-build ←
    /2010-12-22-01
```

Neste momento, o kdesrc-build deverá começar a baixar o código e a compilá-lo. Dependendo dos módulos que baixar, pode ser possível que o kdesrc-build não consiga ter sucesso na primeira vez que compilar o KDE. Não se desespere!

O kdesrc-build registra o resultado de todos os comandos que executa. Por padrão, os arquivos de registro são mantidos em `~/kdesrc/log`. Para ver o que causou um determinado erro num módulo no último comando do kdesrc-build, normalmente é suficiente ver o `~/kdesrc/log/latest/ nome-módulo /error.log`.

DICA

Talvez a forma mais simples de descobrir que erro provocou um problema na compilação de um módulo será pesquisar para trás sem distinção de maiúsculas/minúsculas, a partir do fim do arquivo, pela palavra `erro`. Assim que a encontrar, suba um pouco para garantir que não existem outros erros por perto. A primeira mensagem de erro de um grupo é normalmente o problema subjacente.

Nesse arquivo, você irá ver o erro que fez com que a compilação falhasse para esse módulo. Se o arquivo disser (no fim) que faltam alguns pacotes, tente instalar o pacote (incluindo os pacotes `-dev` apropriados) antes de tentar compilar esse módulo. Certifique-se de que executa o kdesrc-build de novo para passar a opção `--reconfigure`, para que o kdesrc-build force a verificação dos pacotes em falta por parte do módulo de novo.

Ou, se o erro parecer ser um erro de compilação do código do KDE (como um erro de sintaxe, 'protótipo incorreto', 'tipo desconhecido' ou algo do gênero), pode ser que isso seja resolvido dentro de alguns dias. Se não for resolvido em tempo útil, sintá-se à vontade para enviar um e-mail para kde-devel@kde.org (poderá ser necessário inscrever-se) para comunicar o erro de compilação.

Você poderá encontrar mais exemplos comuns das coisas que poderão correr mal e as suas soluções, assim como algumas sugestões e estratégias gerais para compilar o KDE, no [Compilando o KDE 4 a Partir do Código](#).

Por outro lado, assumindo que tudo correu bem, você deverá ter uma nova instalação do KDE no seu computador, sendo agora apenas uma questão de executá-lo, sendo isto descrito a seguir em [Seção 2.5](#).

NOTA

Para mais informações sobre as funcionalidades de registro do kdesrc-build, veja em Seção 3.2.

2.5 Configurando o ambiente para executar o seu ambiente de trabalho Plasma do KDE

Assumindo que você está usando um usuário dedicado para compilar o KDE e já possui uma versão do KDE instalada, a execução do seu KDE novo precisará de alguns truques, uma vez que o KDE novo terá de ter precedência sobre o antigo. Você deve alterar as variáveis de ambiente dos seus scripts de inicialização para ter certeza de que é usada a mais nova compilação do ambiente de trabalho.

2.5.1 Instalar automaticamente um driver de autenticação

Desde a versão 1.16, o kdesrc-build irá tentar instalar um driver de autenticação apropriado, o que lhe permitirá iniciar a sua sessão no seu ambiente de trabalho KDE compilado pelo kdesrc-build a partir do seu gerenciador de autenticação. Isto poderá ser desativado com a opção `install-session-driver` do arquivo de configuração.

NOTA

A configuração da sessão não ocorre quando o kdesrc-build estiver em execução no modo simulado.

Este driver funciona ao definir um tipo de sessão personalizado com o `'xsession'`. Este tipo de sessão deverá funcionar por padrão com o gerenciador de autenticação kdm (onde aparece como uma sessão 'Personalizada'), mas os outros gerenciadores de autenticação (como o LightDM e o gdm) poderão necessitar de arquivos adicionais instalados para ativar o suporte ao `xsession`.

2.5.1.1 Adicionar o suporte da `xsession` para as distribuições

Os gerenciadores de autenticação predefinidos de algumas distribuições poderão necessitar de pacotes adicionais instalados para ter suporte as autenticações com o `xsession`.

- A distribuição Linux® [Fedora](#) precisa que o pacote `xorg-x11-xinit-session` esteja instalado para ter o suporte personalizado de sessões do `xsession`.
- As distribuições de Linux® [Debian](#) e derivadas deverão ter suporte às autenticações personalizadas com o `xsession`, mas precisam que a opção `allow-user-xsession` esteja configurada no arquivo `/etc/X11/Xsession.options`. Veja também a documentação do Debian sobre [como personalizar a sessão de X](#).
- Para outras distribuições, vá para Seção 2.5.1.2.

2.5.1.2 Adicionar manualmente o suporte ao `xsession`

Se não existirem instruções específicas para a sua distribuição em Seção 2.5.1.1, você pode adicionar manualmente um item de 'Sessão personalizada do `xsession`' na lista de tipos de sessões da sua distribuição da seguinte forma:

NOTA

Este procedimento irá necessitar de privilégios administrativos para ser concluído.

Manual do script kdesrc-build

1. Crie o arquivo `/usr/share/xsessions/kdesrc-build.desktop`.
2. Certifique-se de que o arquivo recém-criado tenha o seguinte texto:

```
Type=XSession
Exec=$HOME/.xsession
Name=Área de Trabalho Plasma do KDE (instável; kdesrc-build)
```

- ❶ O item `$HOME` deverá ser substituído pela localização completa da sua pasta pessoal (por exemplo, `/home/ usuário`). A definição de itens desktop não permite arquivos genéricos para os usuários.
3. Quando o gerenciador de autenticação for reiniciado, ele deve mostrar um novo tipo de sessão, o 'Área de Trabalho Plasma do KDE (instável; kdesrc-build)' na sua lista de sessões, que deverá então tentar executar o arquivo `.xsession` instalado pelo `kdesrc-build`, caso o tenha selecionado na sua autenticação.

NOTA

Pode ser mais simples reiniciar o computador para reiniciar o gerenciador de autenticação, caso este não acompanhe as atualizações da pasta `/usr/share/xsessions`.

2.5.2 Configurar manualmente o ambiente

Esta documentação era utilizada para incluir instruções sobre as variáveis de ambiente a definir para carregar o novo ambiente de trabalho. Estas instruções foram movidas para um apêndice (Seção B.1).

Se pretender configurar o seu próprio suporte para autenticações, poderá consultar esse apêndice ou ver o arquivo `sample-kde-env-master.sh`, que vem incluído com o código do `kdesrc-build`.

Capítulo 3

Funcionalidades do Programa

3.1 Visão Geral das Funcionalidades

As funcionalidades do kdesrc-build incluem:

- Você poderá ‘fingir’ efetuar as operações. Se passar a opção `--pretend` ou `-p` na linha de comando, o programa fornecerá uma descrição completa dos comandos que está prestes a executar, sem executá-los de fato. Contudo, se você nunca executou o kdesrc-build, deverá primeiro executar o comando **kdesrc-build --metadata-only** para que a opção `--pretend` funcione.

DICA

Para uma descrição ainda mais detalhada do que o kdesrc-build está fazendo, tente usar a opção `--debug`.

- O kdesrc-build pode (com a assistência do servidor de FTP do KDE) permitir extrações rápidas de alguns módulos do Subversion. Se o módulo que estiver extraindo já possuir um pacote equivalente na página Web, então o kdesrc-build irá baixar essa imagem e prepará-la para o seu computador.

DICA

Normalmente não existe necessidade de qualquer preparação para a obtenção inicial de um módulo do Git, uma vez que é necessário transferir o repositório do Git inteiro, de qualquer forma, sendo simples para o servidor determinar o que deve enviar.

Isto é mais rápido para você, e ajuda a liberar alguma carga nos servidores anônimos de Subversion do kde.org.

- Outra melhoria de performance é conseguida ao iniciar o processo de compilação de um módulo assim que o código-fonte desse módulo tenha sido transferido. (Disponível desde a versão 1.6)
- Um excelente suporte para compilação da biblioteca Qt™ (no caso de o ambiente do KDE que você está tentando compilar depender de uma versão recente do Qt™ que não esteja disponível na sua distribuição).
- O kdesrc-build não precisa de uma GUI presente para funcionar. Por isso, você poderá compilar o KDE sem precisar de um ambiente gráfico alternativo.

Manual do script kdesrc-build

- Suporta a definição de opções padrão para todos os módulos (como as opções de compilação ou de configuração). Essas opções também poderão ser modificadas normalmente para módulos específicos.

Do mesmo modo, o kdesrc-build irá **adicionar opções padrão**, como for apropriado, para lhe poupar trabalho e alguns erros possíveis, definindo-as você mesmo.

- O kdesrc-build poderá baixar uma **marca ou ramificação específica** de um módulo. Você poderá também ter a certeza de que é usada uma **versão** específica em vez de um módulo.
- O kdesrc-build poderá mudar automaticamente a pasta de origem para baixar a partir de um repositório, ramificação ou marca diferentes. Isto acontece automaticamente quando você mudar alguma opção que altere o URL do repositório, se bem que você deve usar a opção **--src-only** para deixar que o kdesrc-build saiba que é aceitável efetuar a mudança.
- O kdesrc-build pode **obter apenas partes de um módulo**, para aquelas situações em que você só precisa de um programa de um módulo maior.
- Para desenvolvedores: O kdesrc-build irá **lembrar de você** se você usar o `svn+ssh://` mas o `ssh-agent` não estiver em execução, uma vez que isso levará a pedidos de senha repetidos do SSH.
- Pode **remover a pasta de compilação** de um módulo, após a sua instalação, para poupar espaço, em detrimento de um maior tempo de compilação.
- Os locais das pastas usadas pelo kdesrc-build são configuráveis (até mesmo por módulo).
- Pode usar o Sudo ou um comando definido pelo usuário para **instalar os módulos**, para que o kdesrc-build não precise ser executado como superusuário.
- O kdesrc-build executa **com prioridade reduzida** por padrão, para lhe permitir ainda usar o seu computador, enquanto o kdesrc-build estiver trabalhando.
- Possui suporte para usar as **marcas e ramificações** do KDE.
- Existe o suporte para **prosseguir com uma compilação** de um determinado módulo. Você pode até **ignorar alguns módulos** temporariamente para uma determinada compilação.
- O kdesrc-build mostrará o **progresso da sua compilação** ao usar o CMake, e irá medir sempre o tempo do processo de compilação, para que você saiba quanto tempo levou efetivamente.
- Vem com um conjunto seguro de opções padrão, apropriadas para compilar uma instalação mono-usuário básica do KDE a partir do repositório de código-fonte anônimo.
- Expansão do til (~) para as suas opções de configuração. Por exemplo, você poderá especificar:

```
qtdir ~/kdesrc/build/qt
```

- Configura automaticamente um sistema de compilação, onde a pasta de código não é a mesma que a pasta de compilação, para que você mantenha as suas pastas de origem limpas.
- Você poderá indicar as opções globais a aplicar a todos os módulos a serem obtidos, assim como poderá indicar opções para serem aplicadas aos módulos individualmente.
- Você pode forçar compilações completas do zero, caso execute o kdesrc-build com a opção `--refresh-build`.
- Você poderá indicar vários valores de ambiente a usar durante a compilação, incluindo o `KDEDIR`, o `QTDIR`, o `DO_NOT_COMPILE` e o `CXXFLAGS`.
- Registro dos comandos. Os arquivos de registro são datados e numerados, para que você tenha sempre um relatório de execução de um programa. Do mesmo modo, é criada uma ligação simbólica 'latest' para apontar sempre para o item de registro mais recente na respectiva pasta.
- Você poderá obter apenas uma parte de um módulo do KDE no Subversion. Por exemplo, poderá obter apenas a biblioteca `taglib` do `kdesupport`.

3.2 Registro de compilação do kdesrc-build

3.2.1 Visão geral do registro

O registro é uma funcionalidade do kdesrc-build que consiste no fato do resultado de todos os comandos que o kdesrc-build executa serem salvos em um arquivo para serem posteriormente examinados, caso seja necessário. Isto é feito porque frequentemente é necessário ter o resultado destes programas quando houver um erro de compilação, uma vez que poderão existir várias razões para a compilação falhar, em primeiro lugar.

3.2.1.1 Disposição da pasta de registro

Os registros são sempre salvos sob a pasta de registro. O destino da pasta de registro é controlado pela opção `log-dir`, que por padrão é `$(source-dir) /log` (onde o `$(source-dir)` é o valor da opção `source-dir`. No resto desta seção, este valor será referido como `$(log-dir)`.

Sob a `$(log-dir)`, existe um conjunto de pastas, uma de cada vez que o kdesrc-build foi executado. Cada pasta possui como nome a data e o número da execução. Por exemplo, a segunda vez que o kdesrc-build foi executado em 26 de maio de 2004, será criada uma pasta chamada `2004-05-26-02`, onde o `2004-05-26` corresponde à data e o `-02` é o número da execução.

Para sua conveniência, o kdesrc-build irá também criar uma ligação para os registros da sua última execução, chamada `latest`. Assim, os registros da execução mais recente do kdesrc-build deverão estar sempre sob a `$(log-dir) /latest`.

Agora, cada pasta de uma execução do kdesrc-build irá conter, por sua vez, um conjunto de pastas, uma por cada módulo KDE que o kdesrc-build tenta compilar. Do mesmo modo, irá conter um arquivo chamado `build-status` nessa pasta, que lhe permitirá determinar quais os módulos que compilaram bem e os que falharam.

NOTA

Se um módulo em si tiver um submódulo (como o `extragear/multimedia`, `playground/utils` ou `KDE/kdelibs`), então, haverá uma disposição equivalente na pasta de registros. Por exemplo, os registros do `KDE/kdelibs`, após a última execução do kdesrc-build, irão aparecer em `$(log-dir) /latest/KDE/kdelibs` e não em `$(log-dir) /latest/kdelibs`.

Na pasta de registro de cada módulo, você irá encontrar um conjunto de arquivos para cada operação que o kdesrc-build efetuar. Se o kdesrc-build atualizar um módulo, você verá arquivos chamados `svn-co.log` (para um módulo extraído por completo) ou `svn-up.log` (ao atualizar um módulo que já tinha sido extraído). Se o comando **configure** foi executado, então você provavelmente verá um arquivo `configure.log` nessa pasta.

Se ocorreu um erro, você deverá ser capaz de ver uma explicação do motivo em um dos arquivos. Para ajudá-lo a determinar qual o arquivo que contém o erro, o kdesrc-build irá criar um link do arquivo que contém o erro (do tipo `build-1.log` para um arquivo chamado `error.log`).

A vantagem de tudo isto é que, para ver onde um arquivo falhou na compilação, após a sua última execução do kdesrc-build, o arquivo que deverá ver primeiro é o `$(log-dir) /latest/nome-módulo /error.log`.

DICA

Se o arquivo `error.log` estiver em branco (especialmente após uma instalação), então provavelmente não houve qualquer erro. Algumas das ferramentas usadas pelo sistema de compilação do KDE irão comunicar por engano um erro quando não ocorreu nada.

Do mesmo modo, alguns comandos poderão fugir ao redirecionamento do resultado do kdesrc-build e ignorar o arquivo de registro em certas circunstâncias (normalmente ao obter o conteúdo do Subversion na primeira vez), e o resultado do erro, nesse caso, não estará no arquivo de registro mas sim no Konsole ou no terminal onde executou o kdesrc-build.

Capítulo 4

Configurando o kdesrc-build

4.1 Visão geral da configuração do kdesrc-build

Para usar o script, você deverá ter um arquivo na sua pasta pessoal chamado `.kdesrc-buildrc`, que descreve os módulos que deseja baixar e compilar, assim como todas as opções ou parâmetros de configuração a serem usados para esses módulos.

4.1.1 Layout do arquivo de configuração

4.1.1.1 Configuração global

O arquivo de configuração começa com as opções globais, indicadas da seguinte forma:

```
global
nome-opção valor-opção
[...]
end global
```

4.1.1.2 Configuração do módulo

Segue-se então uma ou mais seções dos módulos, especificadas em uma das duas formas a seguir:

- ```
module nome-módulo
nome-opção valor-opção
[...]
end module
```

- ```
module-set nome-conjunto-módulos
  repository kde-projects or git://host.org/path/to/repo.git
  use-modules nomes-módulos

# Outras opções podem também ser definidas
nome-opção valor-opção
[...]
end module-set
```


IMPORTANTE

Note que a segunda forma, com conjuntos de módulos, *só funciona para módulos baseados no Git*.

Para módulos do Subversion, *nome-módulo* deverá ser um módulo do repositório de Subversion do KDE (por exemplo, o kdeartwork ou o kde-wallpapers), apesar de ser possível contornar isto se [especificar manualmente a URL do repositório Subversion](#).

Para módulos do Git, o nome do módulo pode ser essencialmente o que desejar, desde que ele não duplique qualquer outro nome de módulo na configuração. Tenha em mente que o layout do diretório de código e de compilação será baseado no nome do módulo se você não usar a opção [dest-dir](#).

No entanto, para os *conjuntos de módulos* do Git o *nomes-módulos* devem corresponder com os módulos atuais do git no `repository` escolhido. Veja [git-repository-base](#) ou [use-modules](#) para mais informações.

4.1.1.3 Módulos de 'options'

Existe um tipo final de item do arquivo de configuração, os grupos `options`, que podem ser indicados sempre que um `module` ou `module-set` puder ser usado.

```
options nome-módulo
nome-opção valor-opção
[...]
end options
```

Um grupo de `options` pode ter opções definidas para ele, como na declaração de um módulo, estando associado a um módulo existente. Todas as opções definidas dessa forma serão usadas para *substituir* as opções definidas para o módulo associado.

IMPORTANTE

O nome do módulo associado *deve* corresponder ao nome indicado na declaração `options`. Tenha cuidado com os erros na grafia do nome.

Isto é útil para permitir declarar um `module-set` inteiro composto por vários módulos, todos com as mesmas opções e usando os grupos de `options` para fazer alterações individuais.

Example 4.1 Exemplo de uso do 'options'

Neste exemplo optamos por compilar todos os módulos do grupo de aplicativos multimídia do KDE. Contudo, queremos usar uma versão diferente do aplicativo KMix (talvez para testar uma correção de erro). Isto funciona da seguinte forma:

```
module-set conjunto-kde-multimedia
  repository kde-projects
  use-modules kde/kdemultimedia
  branch master
end module-set

# kmix é uma parte do grupo kde/kdemultimedia, ainda que nunca tenhamos
# citado o kmix anteriormente, o kdesrc-build irá perceber a alteração.
options kmix
  branch KDE/4.12
end options
```

Agora, quando executar o kdesrc-build, todos os programas multimídia do KDE serão compilados a partir da ramificação 'master' do repositório de código, mas o KMix será compilado a partir da ramificação mais antiga 'KDE/4.12'. Ao usar o options, você não deve listar individualmente todos os outros programas multimídia do KDE para lhes dar a opção de ramificação correta.

NOTA

Lembre-se de que essa funcionalidade só está disponível a partir da versão 1.16 do kdesrc-build ou da sua versão em desenvolvimento após o dia 12/01/2014.

4.1.2 Incluindo outros arquivos de configuração

Dentro do arquivo de configuração, você pode referenciar outros arquivos usando a chave include com um arquivo, que funcionará como se o arquivo referenciado tivesse sido inserido no arquivo de configuração naquele ponto.

Por exemplo, você poderá fazer algo como isto:

```
global
  include ~/common-kdesrc-build-options

  # Insira opções específicas aqui.
end global
```

NOTA

Se você não indicar o caminho completo do arquivo a incluir, então o arquivo será pesquisado a partir da pasta que contém o arquivo de origem. Isso funciona também de forma recursiva.

4.1.3 Opções de configuração comumente usadas

Segue-se uma lista das opções normalmente usadas. Clique na opção para descobrir mais sobre ela. Para visualizar a lista completa de opções, veja em Seção 4.2.

- `cmake-options` para definir as opções com que um módulo poderá ser configurado com o CMake.
- `branch`, para obter uma determinada ramificação, em vez da `'/trunk'` (para o Subversion) ou `master` (para o Git).
- `configure-flags` para definir as opções com que o Qt™ poderá ser configurado.
- `kdedir`, para indicar a pasta onde o KDE será instalado.
- `make-options`, para passar opções ao programa Make (como o número de CPUs a usar).
- `qtdir`, para definir a localização do Qt™.
- `source-dir`, para mudar o local onde colocar o código transferido.

4.2 Tabela das opções de configuração disponíveis

Aqui está uma tabela com as várias opções, contendo a seguinte informação:

- O nome da opção
- Uma descrição de como responde o kdesrc-build se a opção for definida tanto na seção global quanto na seção do módulo do [arquivo de configuração](#) ao compilar um módulo.
- Comentários especiais sobre o objetivo e utilização da opção.

Nome da Opção	Comportamento do Módulo -> Global	Notas
apidox		Esta opção foi usada para permitir a compilação da documentação da API do módulo do KDE. Ela foi removida do kdesrc-build 1.6.3 devido a não ser suportada no KDE 4. A documentação on-line da API está disponível em kde.org . Além disso, é possível compilar a documentação da API do KDE 4 usando o script <code>kdedoxgen.sh</code> , incluído no módulo <code>'kde-dev-scripts'</code> . Para mais detalhes, consulte o KDE TechBase .
apply-qt-patches		Esta opção foi removida no kdesrc-build 1.10. Para obter o mesmo efeito, veja Seção 6.2.1 e a opção repository .

<p>async</p>	<p>Não poderá ser sobreposta</p>	<p>Esta opção ativa o modo de operação assíncrono, onde a atualização do código fonte e o processo de compilação são executados em paralelo, ao invés de esperar por toda a atualização do código antes de iniciar o processo de compilação. O padrão é o modo assíncrono ativado. Para desativar, configure a opção para false Esta opção está disponível desde a versão 1.6.</p>
<p>binpath</p>	<p>A configuração do módulo substitui a global</p>	<p>Ative esta opção para modificar a variável de ambiente PATH durante a compilação. Não poderá sobrepor esta opção numa opção do módulo. O valor padrão é o \$PATH que é definido quando o programa inicia. Esta variável de ambiente deverá incluir os locais, separados por dois-pontos das suas ferramentas de desenvolvimento. Os locais \$KDEDIR /bin e \$QTDIR /bin são adicionados automaticamente. Você poderá usar o til (~) para os locais que adicionar com esta opção.</p>

ramificação	A configuração do módulo substitui a global	<p>Altere esta opção para obter uma ramificação do KDE, em vez do valor padrão <i>master</i> (para os módulos de Git) ou <i>trunk</i> (para o Subversion), onde ocorre o desenvolvimento do KDE. Por exemplo, para obter a versão 4.6 do KDE, você deve configurar esta opção como <i>4.6</i>.</p> <p>Se o kdesrc-build não conseguir obter adequadamente uma ramificação com esta opção, você poderá ter que indicar manualmente o URL de onde baixar usando as opções module-base-path ou override-url.</p> <div data-bbox="997 947 1341 1161" style="border: 1px solid black; padding: 5px;"><p>NOTA Para a maioria dos módulos do KDE, você provavelmente vai querer usar a opção branch-group e esta opção para exceções pontuais.</p></div>
-------------	---	--

<p>branch-group</p>	<p>A configuração do módulo substitui a global</p>	<p>Configure esta opção para um grupo geral de onde deseja escolher os módulos. Para os tipos de módulos com suporte pelo Git, o kdesrc-build irá determinar a ramificação atual a ser usada automaticamente, com base nas regras determinadas pelos desenvolvedores do KDE (estas regras podem ser consultadas no repositório do código <code>kde-build-metadata</code> da sua pasta de código). Depois de determinar uma ramificação, ela será usada como se você mesmo tivesse indicado a opção <code>branch</code>. Isto é útil se você estiver apenas tentando manter-se atualizado sobre algum trajeto normal de desenvolvimento, sem necessidade prestar atenção a todas as mudanças de nome das ramificações. Os grupos de ramificações atuais (em 11/08/2013) são:</p> <ul style="list-style-type: none"> • <code>stable-qt4</code>, para seguir as correções de erros das bibliotecas e aplicativos do KDE, baseados na Qt™ 4. • <code>latest-qt4</code>, para seguir o desenvolvimento e as novas funcionalidades das bibliotecas e aplicativos do KDE, baseado na Qt™ 4. • <code>kf5-qt5</code>, para seguir o desenvolvimento ‘bleeding-edge’ para o KDE 5, Frameworks 5, Espaço de Trabalho Plasma 2, etc., baseados na Qt™ 5. <p>Lembre-se de que, se você escolher <i>criar</i> um <code>branch</code> você mesmo, isto irá substituir esta opção. O mesmo também se aplica para outras opções de seleção da ramificação</p>
---------------------	--	---

<p>build-dir</p>	<p>A configuração do módulo substitui a global</p>	<p>Use esta opção para mudar a pasta que contém o código compilado. Existem três formas diferentes de usá-la:</p> <ol style="list-style-type: none"> 1. Relativa à pasta de código do KDE no Subversion (ver a opção 'source-dir'). Esta é a opção padrão e fica selecionada se digitar o nome de uma pasta que não comece por um til (~) ou uma barra (/). O valor padrão é build. 2. Um local absoluto. Se indicar um local que comece por um /, então será usado esse local diretamente. Por exemplo, o /tmp/kde-obj-dir/ . 3. Relativo à sua pasta pessoal. Se indicar uma pasta que comece com um '~', então o local será usado em relação à sua pasta pessoal, de forma semelhante à expansão do til da linha de comando. Por exemplo, o ~/builddir irá configurar a pasta de compilação como /home/usuario/builddir. <p>Surpreendentemente, esta opção poderá ser alterada por módulo.</p>
------------------	--	--

<p>build-when-unchanged</p>	<p>A configuração do módulo substitui a global</p>	<p>Use esta opção para controlar se o kdesrc-build sempre tentará compilar um módulo que ainda não tenha quaisquer atualizações do código-fonte. Ao configurar o <code>build-when-unchanged</code> como true (verdadeiro), o kdesrc-build sempre tentará a fase de compilação para um módulo, mesmo que este não tenha qualquer atualização no código-fonte. Esta é a configuração padrão, uma vez que, provavelmente, é a que proporciona uma compilação correta. Ao configurar o <code>build-when-unchanged</code> como false (falso), o kdesrc-build só irá tentar executar a fase de compilação de um módulo, caso este tenha tido uma atualização do código-fonte ou em outras situações em que seja realmente necessária uma nova compilação. Isto pode poupar tempo, especialmente se você executa o kdesrc-build diariamente ou com ainda mais frequência.</p> <div data-bbox="997 1507 1338 1936" style="border: 1px solid black; padding: 5px;"><p>IMPORTANTE Esta funcionalidade é oferecida apenas como uma otimização. Como todas as otimizações, existem compromissos face à correção da sua instalação. Por exemplo, as alterações nos módulos 'qt' ou 'kdelibs' poderão provocar uma recompilação necessária de outros módulos, mesmo que o código-fonte não tenha sido alterado.</p></div>
-----------------------------	--	--

Manual do script kdesrc-build

checkout-only	A configuração do módulo substitui a global	<p>Altere esta opção para obter o código do Subversion passo-a-passo. O valor desta opção deverá ser uma lista separada por espaços das pastas a transferir. Ainda que esta opção se sobreponha à global, observe que esta opção, como global, não faz sentido.</p> <p>Note que esta configuração não tem nenhum efeito nos módulos do Git devido ao modo de operação do sistema de controle de código do Git.</p> <p>Veja um exemplo em Seção 6.3.1.1.</p>
---------------	---	---

cmake-options	Adiciona às opções globais para o sistema de compilação padrão e substitui as globais para outros sistemas de compilação.	<p>Use esta opção para indicar quais opções passar ao CMake quando criar o sistema de compilação do módulo. Quando isto é usado como uma opção global, é aplicada a todos os módulos onde este programa se compila. Quando é usado como uma opção de um módulo, é adicionada ao fim das opções globais. Isto permite-lhe definir opções comuns do CMake na seção global.</p> <p>Esta opção não se aplica ao qt (que não usa o CMake). Use a opção configure-flags neste caso.</p> <p>Uma vez que estas opções são passadas diretamente à linha de comando do CMake, deverão ser indicados como seriam passados no CMake. Por exemplo:</p> <pre>cmake-options -DCMAKE_BUILD_TYPE=RelWithDebInfo</pre> <p>Já que isto é muito trabalhoso, o kdesrc-build faz tudo para que as opções restantes sejam definidas corretamente. Assim, você poderá deixar esta opção em branco. (Em outras palavras, os parâmetros <i>necessários</i> do CMake serão definidos automaticamente)</p>
colorful-output	Não poderá ser sobreposta	Configure esta opção como false para desativar o resultado colorido do kdesrc-build. Por padrão é <i>true</i> . Lembre-se que o kdesrc-build não irá mostrar os códigos de cores em nada que não seja um terminal (como o 'xterm', o Konsole ou o console normal do Linux®).

configure-flags	Adiciona às opções globais para o sistema de compilação padrão e substitui as globais para outros sistemas de compilação.	Use esta opção para indicar quais opções passar ao './configure' quando criar o sistema de compilação do módulo. Quando isto é usado como uma opção global, é aplicada a todos os módulos que este programa compila. <i>Esta opção somente funciona para o qt.</i> Para mudar a configuração das opções para os módulos do KDE 4, veja cmake-options .
custom-build-command	A configuração do módulo substitui a global (opção do sistema de compilação)	Esta opção pode ser definida para executar um comando diferente (outro que não seja o make , por exemplo) para realizar o processo de compilação. O kdesrc-build deve fazer a coisa certa, em geral, de modo que você não deve precisar definir esta opção. No entanto, isto pode ser útil para usar sistemas de compilação alternativos. O valor desta opção é usado como a linha de comando a ser executada, modificada pela opção make-options como usualmente.
cxxflags	Adiciona às opções globais para o sistema de compilação padrão e substitui as globais para outros sistemas de compilação.	Use esta opção para indicar as opções a usar para compilar o módulo. Esta opção é indicada aqui em vez do configure-flags ou do cmake-options , uma vez que esta opção irá também definir a variável de ambiente CXXFLAGS durante o processo de compilação. Observe que para o KDE 4 e qualquer outro módulo que use o CMake, é necessário configurar a opção CMAKE_BUILD_TYPE para none ao configurar o módulo. Isto pode ser feito usando a opção cmake-options .

Manual do script kdesrc-build

dest-dir	A configuração do módulo substitui a global	Use esta opção para mudar o nome de um módulo no disco. Por exemplo, se o seu módulo era 'extragear/network', você pode mudá-lo para 'extragear-network' com esta opção. Repare que, ainda que isto mude o nome do módulo no disco, não é uma boa ideia incluir pastas ou separadores de pastas no nome, uma vez que isso irá interferir com as opções build-dir ou source-dir .
disable-agent-check	Não poderá ser sobreposta	Normalmente, se você estiver usando o SSH para baixar o código do Subversion (se, por exemplo, estiver usando o protocolo 'svn+ssh'), o kdesrc-build irá tentar certificar-se de que, se estiver usando o 'ssh-agent', ele está de fato gerenciando algumas identidades do SSH. Isto é para tentar evitar que o SSH pergunte a sua frase-senha para todos os módulos. Você poderá desativar esta opção configurando a opção <code>disable-agent-check</code> como true .

Manual do script kdesrc-build

do-not-compile	A configuração do módulo substitui a global	Use esta opção para especificar um conjunto de pastas para não ser compilado num módulo (em vez de serem todas). As pastas que não serão compiladas deverão estar separadas por espaços. Lembre-se que o código dos programas será ainda transferido. Você poderá usar a instrução checkout-only para escolher as pastas que deseja obter. Por exemplo, para manter o JuK e o KsCD no módulo 'kdemultimedia' sem serem compilados, você poderá adicionar "do-not-compile juk kscd" no seu módulo 'kdemultimedia'. Veja em Seção 6.3.1.2 um exemplo.
email-address	Não poderá ser sobreposta	Esta opção foi removida no kdesrc-build 1.14.
email-on-compile-error	Não poderá ser sobreposta	Esta opção foi removida no kdesrc-build 1.14.
inst-apps		Esta opção foi removida na versão 1.10

git-desired-protocol	Não poderá ser sobreposta	<p>Esta opção somente se aplica a módulos de um repositório do projeto KDE. O que esta opção faz de fato é configurar qual protocolo de rede preferir ao atualizar o código-fonte destes módulos. Normalmente, o eficiente protocolo <code>git</code> é usado, mas ele pode estar bloqueado em algumas redes (por exemplo, Intranets corporativas, Wi-Fi público). Um protocolo alternativo que é muito melhor suportado é o protocolo <code>HTTP</code> usado para páginas Web na Internet. Se você estiver usando uma destas redes restritas, você pode definir esta opção para <code>http</code> para dar preferência a comunicações<code>HTTP</code>.</p> <div data-bbox="997 1066 1341 1272" style="border: 1px solid black; padding: 5px;"><p>DICA Você pode também precisar fornecer a opção <code>http-proxy</code> se um proxy <code>HTTP</code> também for necessário para o tráfego de rede.</p></div> <p>Em qualquer outra situação, você não deve definir esta opção uma vez que o protocolo padrão é mais eficiente. Esta opção foi adicionada com o <code>kdesrc-build</code> 1.16.</p>
----------------------	---------------------------	---

git-repository-base	Não poderá ser sobreposta	<p>Esta opção, adicionada na versão 1.12.1, é usada para criar um nome curto que referencia a URL de base de um dado repositório de Git, no caso das declarações de conjuntos de módulos, o qual é útil para declarar rapidamente diversos módulos do Git para compilar. Você deverá indicar duas coisas (separadas por espaço): O nome a atribuir à URL de base e a URL de base propriamente dita. Por exemplo:</p> <pre>global # outras opções # Isto é o caminho comum para todos os módulos git-repository-base kde-git kde: end global # Declarações de módulo module-set # Agora você pode usar apelidos como definições # em um conjunto de módulos. repository kde-git use-modules module1.git module2.git end module-set</pre> <p>A opção <code>use-modules</code> do conjunto de módulos criou dois módulos a nível interno, com o <code>kdesrc-build</code> comportando-se como se tivesse lido:</p> <pre>module module1 repository kde:module1.git end module module module2 repository kde:module2.git end module</pre> <p>O prefixo <code>kde:</code> de repositório Git usado acima é um atalho que será configurado automaticamente pelo <code>kdesrc-build</code>. Veja o artigo Renomeação de URL na TechBase para obter mais</p>	
---------------------	---------------------------	--	--

<p>git-user</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção destina-se aos desenvolvedores do KDE. Se estiver definida, será usada para configurar automaticamente a informação de identidade do software de controle de versões Git para os módulos <i>recém-baixados</i> do Git (incluindo a grande maioria dos módulos do KDE). Especificamente, os campos do nome e e-mail do usuário, para cada novo repositório Git, são preenchidos com os valores definidos por esta opção. O valor deve ser indicado no formato Nome do usuário <email@exemplo.com>. Por exemplo, um desenvolvedor chamado 'João Silva' com o endereço de e-mail 'joao@abc.xyz' iria usar:</p> <pre>git-user João Silva <joao@abc.xyz></pre> <p>Esta opção foi introduzida no kdesrc-build 15.09.</p>
<p>http-proxy</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção, se definida, usa a URL especificada como servidor proxy a ser usado para qualquer comunicação de rede por HTTP (por exemplo, ao baixar snapshots para novos módulos, ou o banco de dados do projeto KDE). Além disso, o kdesrc-build tentara certificar-se que as ferramentas das quais ele depende também usem este servidor proxy, se possível, definindo a variável de ambiente <code>http_proxy</code> para o servidor indicado, <i>se esta variável já não estiver definida</i>. Esta opção foi introduzida com o kdesrc-build 1.16.</p>

Manual do script kdesrc-build

<code>ignore-kde-structure</code>	A configuração do módulo substitui a global	<p>Esta opção é usada para armazenar o código e os arquivos de compilação diretamente no nome do módulo. Por exemplo, <code>source/extragear/network/telepathy/ktp-text-ui</code> torna-se <code>source/ktp-text-ui</code>. Esta opção fica desativada por padrão. Se quiser ativá-la, você precisa definir como true.</p> <p>Esta opção foi introduzida com o <code>kdesrc-build</code> 1.16.</p>
-----------------------------------	---	---

<p>ignore-modules</p>	<p>Não poderá ser sobreposta</p>	<p>Os módulos indicados com esta opção, que deveriam ser escolhidos pelo kdesrc-build com uma opção <code>use-modules</code>, são assim excluídos por completo. Use esta opção quando quiser compilar um grupo de projetos <code>kde-projects</code> inteiro, <i>exceto para</i> alguns módulos específicos.</p> <p>O valor da opção não tem necessariamente que indicar o nome diretamente ao módulo. Qualquer módulo que tenha partes consecutivas completas do caminho do módulo nos projetos do KDE que correspondam a um dos valores da opção será ignorado, podendo ignorar vários módulos desta forma.</p> <p>Por exemplo, um valor de opção <code>libs</code> resultaria em exclusão do <code>kde/kdegraphics/libs</code> e do <code>playground/libs</code> (ainda que não o <code>kde/kdelibs</code>, uma vez que a parte completa <code>'kdelibs'</code> é que é comparada).</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>DICA Veja também o Exemplo 2.3.</p> </div> <p>Esta opção foi introduzida com o kdesrc-build 1.16.</p>
-----------------------	----------------------------------	---

<p>include-dependencies</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção, quando definida como true (verdadeiro), pede ao kdesrc-build para incluir também as dependências conhecidas destes módulos na sua compilação, sem que o usuário indique essas dependências (mesmo indiretamente).</p> <div data-bbox="997 653 1343 926" style="border: 1px solid black; padding: 5px;"> <p>NOTA Esta opção só funciona para os módulos baseados no kde-project e necessita que os metadados mantidos pelos desenvolvedores do KDE estejam corretos para o seu branch-group selecionado.</p> </div> <p>Esta opção está desativada por padrão, para evitar a seleção inconsciente de um grupo de módulos não desejados. Foi adicionada ao kdesrc-build 1.16 em 2015.</p>
<p>install-after-build</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção é usada para instalar o pacote depois dele ser compilado com sucesso. Esta opção está ativa por padrão. Se quiser desativá-la, terá que configurar esta opção como false no arquivo de configuração. Você poderá também usar a opção da linha de comando <code>--no-install</code>.</p>

<p>install-session-driver</p>	<p>Não poderá ser sobreposta</p>	<p>Por padrão, o kdesrc-build irá tentar instalar um driver para o gerenciador de autenticações gráfico, permitindo-lhe autenticar-se no seu ambiente de trabalho KDE compilado pelo kdesrc-build. Este driver irá alterar os seguintes arquivos:</p> <ul style="list-style-type: none"> • <code>~/.xsession</code> • <code>\$XDG_CONFIG_HOME/kde-env-master.sh</code> (normalmente encontrado em <code>~/.config/kde-env-master.sh</code>). • <code>\$XDG_CONFIG_HOME/kde-env-user.sh</code> (normalmente encontrado em <code>~/.config/kde-env-user.sh</code>). <p>Se continuar com o seu próprio driver de autenticação, então você poderá desativar esta funcionalidade se configurar esta opção como <i>false</i>. Esta opção foi introduzida com o kdesrc-build 1.16.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>DICA O kdesrc-build não irá substituir os seus arquivos existentes (se estiverem presentes) a menos que você também passe a opção de linha de comando <code>--delete-my-settings</code>.</p> </div>
<p>kdedir</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção define a pasta em que o KDE será instalado após a compilação. Por padrão, será em <code>~/kde</code>. Se mudar isto para uma pasta que precise do acesso do 'root', será útil também ler algumas informações sobre a opção make-install-prefix.</p>

<p>kde-languages</p>	<p>Não poderá ser sobreposta</p>	<p>Esta opção permite-lhe selecionar os pacote de localização baixados e instalados juntamente com o KDE. Você poderá fazer isto se não morar nos Estados Unidos e quiser usar o KDE traduzido para o seu idioma nativo. Para usar esta opção, configure-a como uma lista separada por espaços dos idiomas a instalar. Cada língua possui um código associado a ele, que poderá ser encontrado nesta página: http://i18n.kde.org/teams/.</p> <p>É aceitável escolher apenas uma língua. Por padrão, não é baixada nenhuma, o que significa que o KDE irá aparecer em Inglês dos EUA.</p> <p>Por exemplo, para optar por instalar o Português Brasileiro, você deverá configurar a opção com algo do tipo: kde-languages pt_BR. Terá ainda que usar o Configurações do Sistema para escolher o idioma Português do Brasil, todavia.</p>
<p>libpath</p>	<p>A configuração do módulo substitui a global</p>	<p>Configure esta opção para definir a variável de ambiente LD_LIBRARY_PATH, enquanto efetua a compilação. Você não poderá sobrepor esta opção num módulo. O valor padrão está em branco, mas os locais \$KDEDIR /lib e \$QTDIR /lib são adicionados automaticamente. Você poderá usar o til (~) para os locais que adicionar com esta opção.</p>

Manual do script kdesrc-build

log-dir	A configuração do módulo substitui a global	Use esta opção para mudar a pasta que irá conter os arquivos de registro gerados pelo programa.
make-install-prefix	A configuração do módulo substitui a global	Configure esta variável com uma lista separada por espaços, que será interpretada como um comando e as suas opções, para anteceder o comando make install , que é usado para instalar os módulos. Isto é útil para instalar os pacotes com o Sudo, por exemplo, mas tenha cuidado ao trabalhar com os privilégios do 'root'.
make-options	A configuração do módulo substitui a global (opção do sistema de compilação)	Configure esta variável para passar as opções da linha de comando ao comando make . Isto é útil para os programas, como o distcc ou para os sistemas com mais de um núcleo de processamento.
manual-build	A configuração do módulo substitui a global	Configure o valor da opção como true para evitar que o processo de compilação tente compilar este módulo. Ele ainda será mantido atualizado ao atualizar a partir do Subversion. Esta opção é exatamente equivalente à opção da linha de comando --no-build .
manual-update	A configuração do módulo substitui a global	Configure o valor da opção como true para evitar que o processo de compilação tente atualizar (e, por conseguinte, compilar ou instalar) este módulo. Se você definir esta opção para um módulo, então você basicamente poderá tê-la comentada.

module-base-path	A configuração do módulo substitui a global	<p>Configure esta opção para se sobrepor à pasta padrão do kdesrc-build para o módulo em questão. Isto poderá ser usado, por exemplo, para usar ramificações ou marcas específicas de determinadas bibliotecas. O Visualizador de Código do KDE é extremamente valioso para ajudá-lo a escolher o local correto.</p> <p>Lembre-se que o kdesrc-build constrói o local final de acordo com o seguinte modelo: <code>\$svn-server /home/kde/ \$module-base-path.</code> O valor padrão é <code>trunk/ \$módulo</code> ou <code>trunk/KDE/ \$módulo</code>, dependendo do nome do módulo.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>DICA Use as opções branch ou tag em alternativa, sempre que forem aplicáveis.</p> </div>
niceness	Não poderá ser sobreposta	Configure esta opção com um valor entre 20 e 0. Quanto mais alto o número, menor prioridade terá o kdesrc-build, ou seja, quanto maior o número, mais "amigável" o programa será. O valor padrão é 10.
no-svn	A configuração do módulo substitui a global	Se esta opção for igual a 'true', então o kdesrc-build não irá atualizar o código-fonte do módulo automaticamente. Ele ainda tentará compilar o módulo, caso já não tenha tentado anteriormente.
no-rebuild-on-fail		Esta opção foi removida na versão 1.10, uma vez que este comportamento não é mais útil em função de correções aplicadas no sistema de compilação.

<p>override-build-system</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta é uma opção avançada, foi adicionada com o kdesrc-build 1.16. Normalmente o kdesrc-build irá detectar o sistema de compilação apropriado a ser usado em um determinado módulo, depois de baixá-lo. Isto é feito ao verificar a existência de arquivos específicos na pasta do código-fonte do módulo.</p> <p>Alguns módulos poderão incluir mais que um conjunto de arquivos obrigatório, o que poderia confundir a detecção automática. Nesse caso, você pode indicar manualmente o tipo de compilação correto. Os tipos de compilação suportados atualmente que podem ser definidos são:</p> <p>KDE</p> <p>Usado para compilar os módulos do KDE. De fato, ele pode ser usado para compilar praticamente qualquer módulo que use o CMake, mas é melhor não confiar nisso.</p> <p>Qt</p> <p>Usado para compilar a biblioteca Qt™.</p> <p>qmake</p> <p>Usado para compilar módulos do Qt™ que usam os arquivos .pro do qmake.</p> <p>generic</p> <p>Usado para compilar módulos que usam Makefiles simples e que não precisam de nenhuma configuração especial.</p>
------------------------------	--	---

override-url	A configuração do módulo substitui a global	Se configurar esta opção, o kdesrc-build irá usar o seu valor como o URL a passar ao Subversion, <i>sem qualquer alteração</i> . Você geralmente deverá usar esta opção, se quiser obter uma versão específica, mas o kdesrc-build não conseguir descobrir o que você quer dizer ao usar a opção branch .
persistent-data-file	Não poderá ser sobreposta	Use esta opção para alterar onde o kdesrc-build armazena os seus dados persistentes. Por padrão, ele armazena estes dados em um arquivo chamado <code>.kdesrc-build-data</code> , colocado na mesma pasta do arquivo de configuração em uso. Se tiver várias configurações disponíveis na mesma pasta, você pode querer definir manualmente esta opção, para que as diferentes configurações não fiquem com dados persistentes em conflito. Esta opção foi adicionada com o kdesrc-build 1.15.
prefix	A configuração do módulo substitui a global	Esta opção controla onde será instalado o módulo (normalmente em <code>kdedir</code> , se esta opção for usada). O uso desta opção permite-lhe instalar um módulo numa pasta diferente da pasta de instalação das bibliotecas da Plataforma do KDE; por exemplo, você poderá estar usando o kdesrc-build apenas para compilar as aplicações. Você pode usar o <code>\${MODULE}</code> ou <code>\$MODULE</code> no caminho para tê-los expandidos para o nome do módulo.
purge-old-logs	A configuração do módulo substitui a global	Esta opção controla se as pastas de registro antigas serão automaticamente excluídas ou não. O valor padrão é <code>true</code> .

Manual do script kdesrc-build

qmake-options	A configuração do módulo substitui a global	<p>Todas as opções indicadas aqui são passadas para o comando qmake, para módulos que usam o sistema de compilação qmake. Por exemplo, você pode usar a opção PREFIX =/caminho/para/qt para o qmake substituir o local onde o módulo deve ser instalado.</p> <p>Esta opção foi adicionada no kdesrc-build 1.16.</p>
qtdir	A configuração do módulo substitui a global	<p>Configure esta opção para definir a variável de ambiente <code>QTDIR</code> enquanto compila. Você poderá sobrepor esta configuração numa opção de um módulo. Se não indicar esta opção, será por padrão a <code>\${source-dir} /build/qt</code>, que irá usar o módulo 'qt' que está incluído no repositório de código do KDE. Você poderá usar um til (~) que representa a sua pasta pessoal.</p>

<p>remove-after-install</p>	<p>A configuração do módulo substitui a global</p>	<p>Se estiver com pouco espaço em disco, você poderá querer usar esta opção para remover automaticamente a pasta de compilação (ou as pastas de compilação e de código, para instalações episódicas), após o módulo ter sido instalado com sucesso.</p> <p>Os valores possíveis para esta opção são:</p> <ul style="list-style-type: none"> • none - Não remove nada (Este é o valor padrão). • builddir - Remove a pasta de compilação, mas não o código. • all - Remove tanto o código-fonte como a pasta de compilação. <p>Note que a utilização desta opção poderá ter um grande impacto na utilização da sua largura de banda (se usar o <code>all</code>) e o tempo que dura a compilação do KDE, uma vez que o <code>kdesrc-build</code> será incapaz de efetuar compilações incrementais.</p>
<p>repository</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção foi introduzida com a versão 1.10 e é usada para especificar o repositório do Git para transferir o código-fonte para o módulo. O Qt™ (e consequentemente o qt) precisará desta opção, bem como vários módulos do KDE que estão em processo de conversão para usar o Git.</p>

<p>revision</p>	<p>A configuração do módulo substitui a global</p>	<p>Se esta opção for configurada com um valor diferente de 0 (zero), o kdesrc-build irá obrigar a atualização do código ao obter o módulo que tenha exatamente esse número de versão, mesmo que as opções como o branch estejam em efeito. Se o módulo já estiver na versão indicada, então não será mais atualizado, a menos que esta opção seja alterada ou removida da configuração.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTA Esta opção não funciona para módulos baseados no git (incluindo os módulos kde-projects) até a versão 1.16 do kdesrc-build.</p> </div>
<p>run-tests</p>	<p>A configuração do módulo substitui a global (opção do sistema de compilação)</p>	<p>Se definido para true, então o módulo será compilado com suporte para executar seu pacote de testes, e o pacote de testes será executado como parte do processo de compilação. O kdesrc-build mostrará um relatório resumido dos resultados do teste. Isto é útil para desenvolvedores ou para aqueles que desejam se certificar de que seu sistema foi configurado corretamente.</p>

set-env	A configuração do módulo substitui a global	<p>Esta opção aceita um conjunto de valores separados por espaços, onde o primeiro valor é a variável de ambiente a definir e o resto dos valores é o que deseja atribuir à sua variável. Por exemplo, para configurar a variável <code>RONALD</code> como <code>McDonald</code>, você deve colocar este comando na seção apropriada:</p> <pre>set-env RONALD McDonald</pre> <p>Esta opção é especial, na medida em que poderá ser repetida sem sobrepor a configuração anterior do 'set-env' na mesma seção do arquivo de configuração. Desta forma, você poderá definir mais de uma variável de ambiente por módulo (ou a nível global).</p>
source-dir	A configuração do módulo substitui a global	<p>Esta opção é usada para definir a pasta do seu computador onde será guardado o código-fonte do Subversion do KDE. Se não indicar este valor, será usado o valor padrão <code>~/kdesrc</code>. Você poderá usar o til (<code>~</code>) para representar a pasta pessoal se usar esta opção.</p>
ssh-identity-file	Não poderá ser sobreposta	<p>Defina esta opção para controlar qual arquivo de chave SSH privada é passado para o comando ssh-add quando o <code>kdesrc-build</code> estiver baixando o código-fonte dos repositórios que necessitam de autenticação. Veja também: Seção 6.4.1. Esta opção foi adicionada na versão 1.14.2.</p>

<p>stop-on-failure</p>	<p>A configuração do módulo substitui a global</p>	<p>Configure o valor desta opção como true para fazer com que o programa pare a execução, após ocorrer um erro durante o processo de compilação ou instalação. Esta opção está desativada por padrão.</p>
<p>svn-server</p>	<p>A configuração do módulo substitui a global</p>	<p>Esta opção é usada para definir o servidor usado para obter o código do Subversion. Por padrão, é usado o repositório anônimo de Subversion, em <code>svn://anonsvn.kde.org/</code></p> <div data-bbox="997 810 1343 1056" style="border: 1px solid black; padding: 5px;"> <p>NOTA Se você estiver programando para o KDE, use o repositório de Subversion que lhe foi fornecido quando recebeu a sua conta de desenvolvedor, em vez do repositório anônimo.</p> </div>
<p>tag</p>	<p>A configuração do módulo substitui a global</p>	<p>Use esta opção para obter uma versão específica de um módulo. <i>Nota:</i> As chances serão muito boas quando <i>não quiser</i> usar esta opção. As versões do KDE estão disponíveis em pacotes compactados no servidor de FTP do KDE ou num dos seus servidores espelho.</p> <div data-bbox="997 1461 1343 1646" style="border: 1px solid black; padding: 5px;"> <p>NOTA Esta opção para os módulos baseados no git somente tem suporte a partir da versão 1.16 do kdesrc-build.</p> </div>

<p>use-clean-install</p>	<p>A configuração do módulo substitui a global (opção do sistema de compilação)</p>	<p>Defina esta opção para true para que o kdesrc-build execute um make uninstall imediatamente antes de executar um make install. Isto pode ser útil para garantir que não existam arquivos de biblioteca desatualizados, metadados do CMake, etc, que podem causar problemas em instalações antigas do KDE. No entanto, isto somente funciona em sistemas de compilação que suportam o make uninstall. Esta opção foi adicionada com o kdesrc-build 1.12, mas não foi documentada até o kdesrc-build 1.16.</p>
<p>use-cmake</p>		<p>Esta opção foi removida no kdesrc-build 1.4, uma vez que todos os 4 módulos do KDE necessitam do CMake e a utilização do CMake não é permitida em nenhum outro módulo.</p>
<p>use-idle-io-priority</p>	<p>Não poderá ser sobreposta</p>	<p>Esta opção, adicionada no kdesrc-build 1.12, fará com que uma baixa prioridade seja usada para o disco e outro uso de I/O, o que pode melhorar significativamente a responsividade do resto do sistema ao custo de tempos de execução levemente maiores para o kdesrc-build. O padrão é estar desativado. Para ativar a baixa prioridade para o disco, defina isto para true.</p>

use-modules

Só poderá usar no [module-set](#)

Esta opção, adicionada no kdesrc-build 1.12.1, permite-lhe indicar facilmente vários módulos diferentes para compilar ao mesmo tempo no [arquivo de configuração](#).

Esta opção *deverá* ser usada dentro de um `module-set`.

Todos os identificadores passados a esta opção serão convertidos internamente para um módulo do kdesrc-build, com uma opção `repository` igual ao repositório do conjunto de módulos, em conjunto com o nome do identificador, para definir o repositório final de onde transferir os dados. Todas as outras opções que estão atribuídas no conjunto de módulos são também copiadas para os módulos gerados sem qualquer alteração.

A ordem em que os módulos são definidos nesta opção é importante, porque será também a ordem em que o kdesrc-build processará os módulos gerados ao atualizar, compilar e instalar. Todos os módulos definidos no conjunto de módulos fornecido serão manipulados antes que o kdesrc-build se mova para o próximo módulo após o conjunto de módulos.

Se você precisar de alterar as opções para um módulo gerado, basta declarar o módulo de novo após a sua definição no conjunto de módulos, definindo depois as opções que desejar.

Ainda que vá alterar as opções definidas para o módulo desta forma, o módulo será ainda atualizado e compilado na ordem definida pelo conjunto de módulos (i.e., não poderá reordenar a sequência de compilação

Manual do script kdesrc-build

use-qt-builddir-hack	A configuração do módulo substitui a global	Esta opção foi removida devido a melhorias no sistema de compilação do Qt™.
use-stable-kde	Não poderá ser sobreposta	Esta opção está obsoleta e será removida (novamente) em uma versão futura. Em vez disso, use a opção <code>branch-group</code> , que oferece mais possibilidades de escolha.

Tabela 4.1: Tabela de Opções

Capítulo 5

Opções da Linha de Comando e Variáveis de Ambiente

5.1 Uso da linha de comando

O kdesrc-build foi desenhado para ser executado como a seguir:

```
kdesrc-build [--options...] [módulos a compilar...]
```

Se nenhum módulo a ser compilado for especificado na linha de comando, então o kdesrc-build compilará todos os módulos do seu arquivo de configuração, na ordem listada neste arquivo (apesar disto poder ser modificado por várias opções do arquivo de configuração).

5.1.1 Opções de linha de comando comumente usadas

A lista completa das opções da linha de comando é fornecida em Seção 5.3. As opções mais comumente usadas incluem:

--pretend (ou -p)

Esta opção faz com que o kdesrc-build indique que ações seriam tomadas, sem de fato realizá-las. Isto pode ser útil para certificar-se de que os módulos que você pensa que serão compilados sejam de fato compilados.

--refresh-build

Esta opção força que o kdesrc-build compile os módulos fornecidos a partir de um ponto inicial totalmente limpo. Qualquer diretório de compilação existente para este módulo será removido e ele será recompilado. Esta opção é útil se você tiver erros ao compilar um módulo e algumas vezes é necessária quando as bibliotecas Qt™ ou KDE mudam.

--no-src

Esta opção faz com que seja pulado o processo de atualização de código. Você pode usá-la se você tiver atualizado recentemente o código-fonte (talvez você tenha feito isto manualmente ou tenha executado o kdesrc-build recentemente) mas ainda deseja recompilar alguns módulos.

--no-build

Esta opção é semelhante ao `--no-src` acima, mas com ela o processo de compilação é pulado.

5.1.2 Especificando os módulos a compilar

Em geral, especificar módulos a compilar é tão simples quanto passar os nomes dos módulos como você definiu no arquivo de configuração. Você pode também passar módulos que são parte de um conjunto de módulos, como nomeados em [use-modules](#), ou o nome de todo o conjunto de módulos em si se você fornecer o seu nome.

No caso específico de conjuntos de módulos baseados no [banco de dados do projeto KDE](#), o kdesrc-build expandirá o nome de módulo dos componentes para determinar o módulo exato que você deseja. Por exemplo, a entrada do kdesrc-build no projeto do KDE está localizada em `extragear/utls/kdesrc-build`. Você pode especificar qualquer uma das opções a seguir para compilar o kdesrc-build:

```
% kdesrc-build +extragear/utls/kdesrc-build
% kdesrc-build +utls/kdesrc-build
% kdesrc-build +kdesrc-build
```

NOTA

Os comandos no exemplo anterior precede o nome do módulo com um +. Isto força que o nome do módulo seja interpretado como um módulo do banco de dados de projeto do KDE, mesmo se este módulo não tiver sido definido em seu arquivo de configuração.

Seja cuidadoso ao especificar projetos muito genéricos (por exemplo, `extragear/utls` em si), uma vez que isto pode levar a uma grande quantidade de módulos a serem compilados. Você deve usar a opção `--pretend` antes de compilar um novo módulo para certificar-se de que serão compilados somente os módulos que deseja.

5.2 Variáveis de Ambiente Suportadas

O kdesrc-build não usa variáveis de ambiente. Se você precisar definir variáveis de ambiente para o ambiente de compilação ou de instalação, veja a opção [set-env](#).

5.3 Parâmetros de linha de comando suportados

O programa aceita as seguintes opções da linha de comando:

--async

Ativa o [modo assíncrono](#), que pode realizar atualizações do código-fonte e compilar o módulo ao mesmo tempo. Isto é o padrão, e esta opção somente precisa ser especificada se você quiser desabilitá-la na configuração.

--help

Mostra apenas uma ajuda simples deste programa.

--version

Mostra a versão do programa.

--author

Mostra as informações de contato do autor.

--color

Ativa a saída em cores. (Isto é o padrão para terminais interativos).

Manual do script kdesrc-build

--nice=valor

Este valor ajusta a prioridade da CPU do computador solicitada pelo kdesrc-build, e deve estar no intervalo de 0-20. O 0 é a prioridade mais alta (por isso ele é o menos 'satisfatório') e o 20 é a prioridade mais baixa. O padrão do kdesrc-build é o 10.

--no-async

Desativa o **modo assíncrono** de atualização. Ao invés da atualização ser realizada completamente antes da compilação iniciar. Esta opção irá aumentar o tempo de todo o processo, mas se você encontrar erros de IPC ao rodar o kdesrc-build tente usar esta opção e submeter um **relatório de erro**.

--no-color

Desativa o resultado do programa à cores.

--pretend (ou -p)

O kdesrc-build executará o processo de atualização e compilação, mas ao invés de realizar qualquer ação efetivamente, ele apresentará o que o programa deveria fazer (quais comandos executar, passos gerais feitos, etc.).

NOTA

Comandos simples apenas para leitura (como a leitura das informações de um arquivo) pode ainda ser executados para tornar o resultado mais relevante (como simular corretamente se o código-fonte deve ser transferido ou atualizado).

IMPORTANTE

Esta opção necessita que alguns metadados obrigatórios estejam disponíveis, que normalmente são baixados automaticamente, mas os downloads estão desativados no modo simulado. Se você nunca executou o kdesrc-build (e, como tal, não tiver esses metadados), deverá primeiro executar o comando **kdesrc-build --metadata-only** para baixar os metadados obrigatórios.

--quiet (ou -q)

Não é tão descritivo com o resultado exibido pela compilação. Com esta opção, somente as mensagens básicas são apresentadas.

--really-quiet

Só mostra as mensagens de avisos e erros.

--verbose (ou -v)

Torna-se bastante descritivo com o que está acontecendo e o que o kdesrc-build está fazendo.

--src-only (ou --svn-only)

Somente realiza uma atualização do código. (O **--svn-only** somente é suportado para compatibilidade com programas antigos).

--build-only

Efetua apenas o processo de compilação.

--install-only

Efetua apenas o processo de instalação.

--metadata-only

Só executa o processo de download dos metadados. O kdesrc-build normalmente faz isso de forma automática, mas você poderá usar esta opção manualmente para permitir que a opção **--pretend** da linha de comando funcione.

--rebuild-failures

Use esta opção para compilar apenas os módulos que não conseguiram compilar em uma execução anterior do kdesrc-build. Isso é útil se ocorrer uma quantidade significativa de falhas em conjunto com compilações bem-sucedidas. Depois de corrigir o erro que provocou os problemas de compilação, os módulos que falharam anteriormente poderão ser facilmente compilados.

NOTA

Lembre-se de que a lista de 'módulos que falharam anteriormente' é apagada toda vez que uma execução do kdesrc-build termina com falha em algum módulo. Contudo, a lista não é apagada com uma compilação completamente bem-sucedida, para que você possa recompilar com sucesso um ou dois módulos e esta opção continuará funcionando.

Esta opção foi adicionada no kdesrc-build 15.09.

--ignore-modules

Não inclui os módulos passados no resto da linha de comando no processo de atualização/compilação (isto é útil se você deseja compilar a maioria dos módulos em seu [arquivo de configuração](#) e apenas pular alguns).

--no-src (ou --no-svn)

Não contacta o servidor Subversion. (O parâmetro `--no-svn` somente é suportado para compatibilidade com versões antigas do programa).

--no-build

Salta o processo de compilação.

--no-metadata

Não baixa automaticamente os metadados extras que são necessários para os módulos Git do KDE. As atualizações de código para os módulos propriamente ainda será efetuada, a menos que você indique também a opção `--no-src`.

Isto pode ser útil se estiver frequentemente executando novamente o kdesrc-build, uma vez que os metadados não são alterados com muita frequência. Mas lembre-se que muitas outras funcionalidades dependem da disponibilidade dos metadados. Você pode considerar a execução do kdesrc-build com a opção `--metadata-only` uma vez e depois usar essa opção para as execuções subsequentes.

--no-install

Não instala automaticamente os pacotes, após a sua compilação.

--no-build-when-unchanged, --force-build

Esta opção desativa de forma explícita a eliminação do processo de compilação (uma otimização controlada pela opção `build-when-unchanged`). Isto é útil para fazer com que o kdesrc-build execute a compilação, caso tenha alterado algo que o kdesrc-build não consegue verificar.

O `--force-build` efetua exatamente a mesma função, sendo talvez mais fácil de se lembrar.

--debug

Ativa o modo de depuração para o programa. Atualmente, isto significa que todo o resultado será enviado para o saída padrão, além de ficar registrado na pasta de registro normalmente. Do mesmo modo, muitas das funções tornar-se-ão mais descritivas sobre o que estão fazendo no modo de depuração.

--no-rebuild-on-fail

Não tenta compilar de novo os módulos que não conseguiram ser compilados do zero. O kdesrc-build nunca irá tentar fazer isto num módulo que já se tentou compilar do zero.

Manual do script kdesrc-build

--refresh-build

Recria o sistema de compilação e compila a partir do zero.

--reconfigure

Executa o **cmake** (para os módulos do KDE) ou o **configure** (para o Qt™) novamente, sem limpar a pasta de compilação. Você normalmente não tem que especificar isto, uma vez que o kdesrc-build detectará quando você muda opções relevantes e automaticamente re-executará a configuração da compilação. Esta opção está implícita se o *--refresh-build* for usado.

--resume-from

Esta opção é usada para prosseguir com a compilação a partir do módulo indicado, que é o parâmetro seguinte na linha de comando. Você não deverá indicar os nomes dos outros módulos na linha de comando.

NOTA

Esta opção antigamente adicionava o *--no-src*, mas já não o faz mais (desde o kdesrc-build 1.13). Se quiser evitar as atualizações de código ao prosseguir, basta passar *--no-src* além das outras opções.

Veja também: *--resume-after* e Seção 6.3.5.1. Você pode preferir usar esta opção de linha de comando se tiver corrigido um erro de compilação e deseja que o kdesrc-build conclua a compilação.

--resume-after

Esta opção é usada para prosseguir com a compilação após o módulo indicado, que é o parâmetro seguinte na linha de comando. Você não deverá indicar os nomes dos outros módulos na linha de comando.

NOTA

Esta opção antigamente adicionava o *--no-src*, mas já não o faz mais (desde o kdesrc-build 1.13). Se quiser evitar as atualizações de código ao prosseguir, basta passar *--no-src* além das outras opções.

Veja também: *--resume-from* e Seção 6.3.5.1. Você pode preferir usar esta opção de linha de comando se tiver corrigido um erro de compilação e tiver compilado e instalado o módulo você mesmo, e deseja que o kdesrc-build inicie novamente com o próximo módulo.

--resume

Esta opção pode ser usada para executar o kdesrc-build após um problema de compilação. Ela continua com a compilação a partir do módulo que falhou, usando a lista de módulos que estavam aguardando compilação e desativando também as atualizações de código e metadados. O caso de uso é quando um erro simples ou a falta de uma dependência provocar a falha na compilação. Assim que corrigir o erro, você poderá retornar rapidamente à compilação dos módulos que estavam sendo compilados, sem precisar mexer nas opções *--resume-from* e *--stop-before*.

Isso ainda é mais útil com o argumento da linha de comandos *--stop-on-failure*, especialmente se estiver iniciando a configuração do seu ambiente de desenvolvimento.

Esta opção foi adicionada com o kdesrc-build 1.16.

--stop-before

Esta opção de linha de comando é usada para interromper o processo normal de compilação somente *antes* da compilação normal de um módulo.

Por exemplo, se a lista normal de compilações fosse móduloA, móduloB, móduloC, então o *--stop-before=móduloB* faria com que o kdesrc-build compilasse apenas o móduloA.

Esta opção de linha de comando foi adicionada com o kdesrc-build 1.16.

--stop-after

Esta opção de linha de comando é usada para interromper o processo normal de compilação somente *após* a compilação normal de um módulo.

Por exemplo, se a lista normal de compilações fosse móduloA, móduloB, móduloC, então o `--stop-after=móduloB` faria com que o kdesrc-build compilasse o móduloA e o móduloB.

Esta opção de linha de comando foi adicionada com o kdesrc-build 1.16.

--stop-on-failure

Esta opção faz com que a compilação seja interrompida assim que ocorrer algum erro. É útil quando estiver configurando o seu ambiente de desenvolvimento inicial. Sem essa opção, o kdesrc-build irá tentar pressionar o resto dos módulos na compilação para evitar perder tempo, caso o problema ocorra apenas com um único módulo.

Esta opção foi adicionada no kdesrc-build 1.16. Veja também a opção [stop-on-failure](#).

--rc-file

Isto interpreta o próximo parâmetro da linha de comando como o arquivo de onde ler as opções de configuração. O valor padrão deste parâmetro é `kdesrc-buildrc` (verificado na pasta atual) se ele estiver presente, ou, caso não esteja, o `~/kdesrc-buildrc`. Veja também capítulo 4.

--print-modules

Executa todas as ações e a reordenação das dependências dos módulos indicados na linha de comando (ou arquivo de configuração), apresenta os módulos que seriam processados, um em cada linha, e então sai sem nenhuma outra ação.

Os metadados do `kde-project` são baixados primeiro (porém, veja as opções `--pretend` e `--no-src`).

O resultado não é totalmente compatível com a utilização pelos scripts, como as outras mensagens do resultado podem ser geradas até que seja apresentada a lista dos módulos.

Isto é útil, principalmente, para determinar rapidamente o que o kdesrc-build reconhece como sendo as dependências de um módulo, o que significa que é apenas útil para os módulos do `kde-projects`. Essa opção também é compatível com `--resume-from`, `--resume-after`, `--stop-before` e `--stop-after`.

--run

Esta opção interpreta o item a seguir na linha de comando como um programa a executar, fazendo com que o kdesrc-build acabe de ler o arquivo de configuração normalmente e depois execute o programa indicado.

Isto não irá funcionar para iniciar uma linha de comando com o ambiente do kdesrc-build, contudo, uma vez que as linhas de comando interativas tipicamente limpam algumas partes das variáveis de ambiente (como a `PATH` e a `KDEDIRS`) na sequência de inicialização.

DICA

Se você deseja ver o ambiente usado pelo kdesrc-build, você pode executar o comando **printenv**:

```
$ kdesrc-build --run printenv
KDE_SESSION_VERSION=4
SDL_AUDIODRIVER=alsa
LANGUAGE=
XCURSOR_THEME=Oxygen_Blue
LESS=-R -M --shift 5
QMAIL_CONTROLDIR=/var/qmail/control
... etc.
```

Manual do script kdesrc-build

--prefix=</localização/do/kde>

Isto permite-lhe mudar a pasta onde o KDE será instalado na linha de comando. Esta opção implica a `--reconfigure`, mas usar o `--refresh-build` pode também ser necessário.

--revision

Esta opção faz com que o kdesrc-build obtenha um número de revisão específico para cada módulo do Subversion, sobrescrevendo qualquer opção `branch`, `tag`, ou `revision` já configurada para estes módulos.

Esta opção provavelmente não é uma boa ideia, e somente é suportada para compatibilidade com programas antigos.

--build-system-only

Esta opção faz com que o kdesrc-build cancele a compilação de um módulo exatamente antes do comando `make` ser executado. Isto é suportado somente para compatibilidade com versões antigas, e seu efeito não é útil para o sistema de compilação atual do KDE.

--install

Se esta for a única opção da linha de comando, ele irá tentar instalar todos os módulos incluídos em `log/latest/build-status`. Se as opções da linha de comando forem indicadas após o `--install`, então assumem-se que são módulos para instalar (mesmo se eles não compilaram com sucesso na última execução).

--no-snapshots

Fornecer esta opção faz com que o kdesrc-build sempre realize uma verificação inicial normal de um módulo ao invés de usar o instantâneo rápido (somente disponível para módulos do Git do repositório `kde-projects`). Observe que esta opção deve somente ser usada se existir uma falha ao usar os instantâneos, uma vez que o uso do instantâneo rápido reduz a carga nos repositórios de código do KDE.

NOTA

Imagens do módulo são transferências reais. Você não deve precisar especificar esta opção, sendo ela somente um auxílio para a resolução de problemas.

--delete-my-patches

Esta opção é usada para permitir ao kdesrc-build excluir as pastas de código que possam conter dados do usuário, para que o módulo possa ser baixado novamente. Isto normalmente só seria útil para os desenvolvedores do KDE (que poderiam ter alterações locais que seriam excluídas).

Isto só é usado atualmente para obter os módulos que foram convertidos do Subversion para o Git. Você não deve usar esta opção normalmente; o kdesrc-build irá solicitar para ser executado com ela, caso seja necessário.

--delete-my-settings

Esta opção é usada para permitir ao kdesrc-build substituir os arquivos existentes que possam conter dados do usuário.

Isto só é usado atualmente para a configuração do `xsession` para o gerenciador de autenticação. Você não deve usar esta opção normalmente; o kdesrc-build irá solicitar para ser executado com ela, caso seja necessário.

--<nome-opção>=

Você poderá usar esta opção para sobrepor outra opção no seu [arquivo de configuração](#) para todos os módulos. Por exemplo, para sobrepor a opção `log-dir`, você deverá digitar: `--log-dir= /local/da/pasta`.

Manual do script kdesrc-build

NOTA

Esta funcionalidade só pode ser usada para os nomes de opções já reconhecidos pelo kdesrc-build, que ainda não sejam pelas opções relevantes da linha de comando. Por exemplo, a opção `async` do arquivo de configuração tem opções `--async` e `--no-async` específicas na linha de comando que são preferidas pelo kdesrc-build.

`--set-module-value-option=<nome-módulo>, <nome-opção>, <valor-opção>`

Ou você poderá usar esta opção para substituir outra no seu [arquivo de configuração](#), para um determinado módulo.

Todas as outras opções da linha de comando assumem-se como sendo módulos para atualizar e compilar. Por favor, não misture a compilação com a instalação.

Capítulo 6

Usando o kdesrc-build

6.1 Prefácio

Normalmente, a utilização do kdesrc-build após ter passado pelo capítulo 2 é tão simples como fazer o seguinte, a partir de uma linha de comando:

```
% kdesrc-build
```

O kdesrc-build irá então baixar o código do KDE, tentará configurá-lo e compilá-lo e, no fim, tentará instalá-lo.

Leia mais para saber como o kdesrc-build faz isso, assim como o que você poderá fazer com esta ferramenta.

6.2 Funcionalidades básicas do kdesrc-build

6.2.1 Suporte para o 'qt'

O kdesrc-build suporta a compilação do conjunto de ferramentas do Qt™ usado pelos programas do KDE como uma conveniência para os usuários. Este suporte é manipulado por um módulo especial denominado 'qt'.

NOTA

O Qt™ é desenvolvido sob um repositório separado dos programas do KDE, localizado em <http://qt.gitorious.org/qt>.

Para compilar o Qt™, você deve certificar-se de que a configuração `qtdir` está definida para a pasta que você gostaria de instalar o Qt™, conforme descrito em Seção 2.2.

Você deve certificar-se de que o módulo do 'qt' está adicionado ao seu `.kdesrc-buildrc`, antes de qualquer outro módulo no arquivo. Se você estiver usando o arquivo de configuração de exemplo, basta retirar o comentário da entrada existente para o módulo 'qt'.

Agora você deve verificar se as opções `repository` e `branch` foram definidas apropriadamente:

1. A primeira opção é para compilar o Qt™ usando um espelho mantido nos repositórios de código do KDE (nenhuma outra alteração é aplicada, ele é simplesmente um clone do código oficial). Isto é altamente recomendado devido a problemas ocasionais com o processo de clonagem completa do módulo do Qt™ a partir do seu repositório oficial.

Você pode definir a opção `repository` do módulo `qt` para `kde:qt` para usar esta opção.

2. Caso contrário, para compilar o Qt™ padrão, defina sua opção `repository` para `git://github.com/qt/qt.git`. Observe que você pode experimentar problemas ao executar a clonagem inicial do Qt™ a partir deste repositório.

Em ambos os casos, a opção 'branch' deve ser definida para `master` (a menos que você deseje compilar uma 'branch' diferente).

6.2.2 Opções normais adicionadas pelo kdesrc-build

Para lhe poupar tempo, o kdesrc-build adiciona alguns locais padrão ao seu ambiente:

- O local das bibliotecas do KDE e do Qt™ é adicionado automaticamente à variável `LD_LIBRARY_PATH`. Isto significa que não irá precisar editar a `libpath` para incluí-los.
- O local dos programas de suporte ao desenvolvimento do KDE e do Qt™ é adicionado automaticamente à variável `PATH`. Isto significa que não irá precisar editar a `binpath` para incluir.
- O local do `pkg-config` fornecido com o KDE é adicionado automaticamente ao `PKG_CONFIG_PATH`. Isto significa que não terá de usar o `set-env` para os incluí-los.
- A configuração do `kdedir` é propagada automaticamente para a variável de ambiente `KDEDIR` durante a compilação. (O `KDEDIRS` não é afetado).
- A configuração do `qtdir` é propagada automaticamente para a variável de ambiente `QTDIR` durante a compilação.

6.2.3 Mudando a prioridade de compilação do kdesrc-build

Os programas poderão ser executados com diferentes níveis de prioridades nos Sistemas Operacionais, incluindo o Linux® e o BSD. Isto permite ao sistema reservar tempo para os diferentes programas, de acordo com a sua importância.

O kdesrc-build irá reservar uma prioridade baixa para si próprio, para que o resto dos programas do seu sistema não sejam afetados e possam rodar normalmente. Com esta técnica, o kdesrc-build irá usar CPU extra quando estiver disponível.

O kdesrc-build irá ainda manter uma prioridade suficientemente alta, para que se execute antes dos processos rotineiros e antes dos programas de doação de CPU, como o `Seti@Home`.

Para alterar o kdesrc-build, de modo a usar um nível de prioridade maior (ou menor) de forma permanente, então você terá que ajustar a opção `niceness` no `arquivo de configuração`. A opção `niceness` controla quão 'legal' é o kdesrc-build com os outros programas. Em outras palavras, um `niceness` maior, dará ao kdesrc-build uma prioridade menor. Assim, para dar ao kdesrc-build uma maior prioridade, reduza o `niceness` (e vice-versa). O `niceness` pode ir de 0 (nada legal, prioridade máxima) até 20 (super legal, prioridade mínima).

Você poderá também mudar temporariamente a prioridade do kdesrc-build ao usar a `opção de linha de comando --nice`. O valor da opção é usado da mesma forma que no `niceness`.

NOTA

É possível para alguns programas executados pelo superusuário ter um valor de prioridade negativo, com uma prioridade de sistema corresponde ainda maior para alguns programas. A configuração de um valor negativo (ou mesmo 0) do `niceness` para o kdesrc-build não é uma boa ideia, uma vez que não irá melhorar muito o tempo de compilação, mas tornará o seu computador muito lento, se ainda precisar usá-lo.

Para executar o kdesrc-build com uma prioridade igual a 15 (menor que o normal):

```
% kdesrc-build --nice=15
```

Ou você poderá editar o [arquivo de configuração](#) para tornar a mudança permanente:

```
niceness 15
```

DICA

A opção `niceness` somente afeta o uso do(s) processador(es) do computador. Uma outra que afeta significativamente o desempenho do computador se refere a quantos dados de entrada ou saída (I/O) um programa usa. Para controlar quanto I/O um programa pode usar, os sistemas operacionais Linux[®] modernos suportam uma ferramenta similar denominada `ionice`. O `kdesrc-build` suporta o `ionice`, (mas somente para ativá-lo ou desativá-lo completamente) usando a opção `use-idle-io-priority`, desde a versão 1.12 do `kdesrc-build`.

6.2.4 Instalação como superusuário

Você poderá ter o `kdesrc-build` executando a instalação com privilégios de superusuário. Isto pode acontecer no caso da instalação a nível do sistema (não recomendada). Também é útil se você usar uma instalação normal para um único usuário, todavia. Isto acontece porque alguns módulos (especialmente o `kdebase`) instalam programas que, por breves instantes, irão necessitar de permissões elevadas ao serem executados. Eles não conseguirão obter esses níveis de permissões, a menos que tenham sido instalados com as permissões elevadas.

Você poderá simplesmente executar o `kdesrc-build` diretamente como superusuário, mas isto não é recomendável, uma vez que o programa não foi testado para esse tipo de uso. Ainda que possa ser seguro executá-lo dessa forma, é melhor evitar executar este programa como superusuário, sempre que possível.

Para cuidar disso, o `kdesrc-build` oferece a opção `make-install-prefix`. Você poderá usar esta opção para indicar um comando a usar para efetuar a instalação como outro usuário. A forma recomendada de usar este comando é com o programa `Sudo`, que irá então executar o comando de instalação como superusuário.

Por exemplo, para instalar todos os módulos com o `Sudo`, você poderá fazer algo como:

```
global
  make-install-prefix sudo
  # Outras opções
end global
```

Para usar o `make-install-prefix` apenas num módulo, isto irá funcionar:

```
module nome-módulo-svn
  make-install-prefix sudo
end module
```

6.2.5 Mostrando o progresso da compilação de um módulo

Esta funcionalidade está sempre disponível e está ativada automaticamente, sempre que possível. O que isto faz é mostrar o progresso estimado da compilação ao compilar um módulo, para que você saiba desta forma quanto tempo mais irá levar para compilar um módulo.

6.3 Funcionalidades avançadas

6.3.1 Compilar parcialmente um módulo

É possível compilar apenas partes de um único módulo do KDE. Por exemplo, você pode querer compilar apenas um programa de um módulo. O `kdesrc-build` possui funcionalidades para simplificar isto. Existem várias formas complementares de fazer esta tarefa.

6.3.1.1 Extrair partes de um módulo

Esta será talvez a melhor forma de fazer isto. Quando funciona, lhe poupará tempo de transferência e espaço em disco. O que acontece é que o `kdesrc-build` irá baixar apenas as partes do módulo que indicar. Isto é feito com a opção `checkout-only` para um determinado módulo, que irá indicar uma lista das pastas a baixar.

DICA

Se você ainda não souber o que vai transferir de um determinado módulo, poderá ser uma boa ideia navegar pela disposição no Subversion para um módulo, em primeiro lugar, usando o [WebSVN](#).

Para obter apenas o `KUser` e o `KSystemLog` do `'kdeadmin'`, você poderá usar o `checkout-only` da seguinte forma:

```
module kdeadmin
  checkout-only kuser ksystemlog
end module
```

IMPORTANTE

As pastas serão compiladas na ordem que forem indicadas na opção. Se uma das pastas necessitar de alguma coisa além do módulo para compilar, então você terá que garantir que ambas se encontram na linha `checkout-only` e que a dependência necessária venha antes da pasta que necessita dela. Da mesma forma, algumas vezes um aplicativo poderá necessitar de outras pastas e isso pode ser difícil de descobrir. Isto poderá ser solucionado através de tentativa e erro, adicionando constantemente pastas à opção para descobrir qual é a necessária. Esta opção depende do suporte no sistema de compilação do módulo, só sendo útil para os módulos que sejam coleções de aplicações individuais.

Uma nota final sobre esta opção: Se você alterar esta opção deverá usar o `kdesrc-build --refresh-build módulo` para garantir que o módulo é configurado de novo convenientemente. Além disso, o `kdesrc-build` nunca irá remover os arquivos existentes, se você retirar o número de pastas da sua opção `checkout-only` ou adicionar a opção a um módulo que já foi extraído.

6.3.1.2 Removendo pastas de uma compilação

Em vez de restringir o que é baixado, é possível transferir tudo mas fazer com que o sistema de compilação deixe de fora algumas pastas ao compilar. Isto pode ser útil se uma pasta estiver sempre invalidando a compilação e for desnecessária para o resto do módulo.

Isto é controlado com a opção `do-not-compile`. Funciona de forma semelhante à opção `checkout-only` descrita acima, e é apenas uma lista das pastas que não deverão ser compiladas.

IMPORTANTE

Da mesma forma que no [checkout-only](#), esta opção necessita pelo menos que o sistema de compilação do módulo seja reconfigurado após a alteração. Isto é feito com o comando **kdesrc-build --reconfigure módulo**.

Para remover a pasta `python` do processo de compilação do 'kdebindings':

```
module kdebindings
  do-not-compile python
end module
```

NOTA

Esta função depende de algumas convenções normais, usadas na maioria dos módulos do KDE. Assim, ela poderá não funcionar para todos os programas.

6.3.2 Suporte de marcas e ramificações do kdesrc-build

6.3.2.1 O que são ramificações e marcas?

O Subversion suporta o gerenciamento do histórico do código do KDE. O KDE usa esse suporte para criar ramificações ('branches') para desenvolvimento e para marcar o repositório com marcas ('tags'), sempre que lança uma nova versão.

Por exemplo, os programadores do KMail poderão estar trabalhando numa nova funcionalidade, numa ramificação diferente, para evitar corromper a versão que está em uso pela maioria dos programadores. Esta ramificação tem desenvolvimentos em curso nela, mesmo quando a ramificação principal (chamada de /trunk) também possui desenvolvimentos em curso nela.

Uma marca, por outro lado, é uma imagem do repositório de código numa determinada posição no tempo. Isto é usado pela equipe de administração do KDE para marcar uma versão do código adequada para lançamento, permitindo ainda aos programadores trabalharem no código.

No Subversion, não existem diferenças entre ramificações, marcas ou o 'trunk' no código. É apenas uma convenção usada pelos programadores. Isso torna difícil suportar convenientemente as ramificações e marcas no kdesrc-build. Contudo, existem algumas coisas que poderão ser feitas.

6.3.2.2 Como usar as ramificações e marcas

O suporte para ramificações e marcas é tratado por um conjunto de opções, que vão desde um simples pedido de uma versão até um URL específico de transferência, para os usuários mais avançados.

O método mais simples é usar as opções `branch` e `tag`. Você poderá usar a opção em conjunto com o nome da marca ou ramificação de um módulo, para que o kdesrc-build determine o local apropriado, dentro do repositório do KDE, de onde deve baixar. Na maioria dos módulos do KDE, isto funciona muito bem.

Para baixar o 'kdelibs' do KDE 4.6 (que é simplesmente conhecido pela ramificação '4.6'):

```
module kdelibs
  branch 4.6
  # outras opções...
end module
```

Ou, para baixar o 'kdemultimedia' como foi lançado no KDE 4.6.1:

```
module kdemultimedia
  tag 4.6.1
  # outras opções...
end module
```

DICA

Você poderá definir aqui um valor global para o 'branch'. Mas, se o fizer, não se esqueça de indicar uma ramificação diferente para os módulos que não devem usar a ramificação global!

6.3.2.3 Opções avançadas de suporte a ramificações

O kdesrc-build suporta duas opções para as situações em que o `branch` e o `tag` adivinham o local de forma incorreta: o `module-base-path` e o `override-url`.

- O `module-base-path` é usado para ajudar o kdesrc-build a preencher a parte faltante da localização de um módulo. No repositório do KDE, todos os locais possuem o formato `raizSvn/caminho-módulo-base/ nome-módulo`. Normalmente, o kdesrc-build pode descobrir a parte intermediária apropriada por si só. Quando não conseguir, você poderá usar o 'caminho-módulo-base' ou `module-base-path` da seguinte forma:

```
module kdesupport
  # o 'kdesupport' suporta várias marcas para organizar facilmente as
  # aplicações necessárias para uma determinada versão da plataforma do ↔
  KDE.
  module-base-path tags/kdesupport-do-4.5
end module
```

Isto deverá fazer com que o kdesrc-build baixe o 'kdesupport' de (neste exemplo) `svn://anon.svn.kde.org/home/kde/ tags/kdesupport-for-4.5`.

DICA

Nas versões anteriores do kdesrc-build, o `module-base-path` era tratado de forma diferente. Se você tiver problemas ao usar uma definição antiga do 'module-base-path', talvez deva confirmar se a localização atual é a mesma que o kdesrc-build está esperando, usando a opção `--pretend`.

- A opção `override-url`, por outro lado, necessita que você indique o local exato de onde baixar. Contudo, isto permite-lhe transferir de locais que versões anteriores do kdesrc-build não teriam chances de baixar. Atualmente, a opção `module-base-path` deve ser suficiente para qualquer URL de fonte de Subversion.

IMPORTANTE

O kdesrc-build não irá mexer ou corrigir o valor que indicar no kdesrc-build, por isso, se mudar a sua configuração do `svn-server`, também terá que atualizar isto.

6.3.3 Como o kdesrc-build tenta garantir uma compilação com sucesso

6.3.3.1 Recompilações automáticas

O kdesrc-build costumava incluir recursos para automaticamente tentar recompilar o módulo após uma falha (uma vez que esta nova tentativa poderia funcionar, devido a falhas no sistema

de compilação naquela época). Graças a mudança para o CMake, o sistema de compilação não sofre mais destas falhas, e assim o kdesrc-build não tentará compilar um módulo mais de uma vez. Existem situações onde o kdesrc-build automaticamente executará uma recompilação:

- Se você mudar o [configure-flags](#) ou o [cmake-options](#) para um módulo, então o kdesrc-build detectará e automaticamente re-executará o 'configure' ou o 'cmake' para aquele módulo.
- Se o sistema de compilação não existir (mesmo se o kdesrc-build não o removeu) então o kdesrc-build automaticamente irá recriá-lo. Isto é útil para permitir a execução de um [--refresh-build](#) completo para um módulo específico sem ter que executá-lo em outros módulos.

6.3.3.2 Recompilando manualmente um módulo

Se você tiver feito alterações em alguma opção de um módulo, ou se o código-fonte do módulo mudar de uma forma que o kdesrc-build não reconheça, você poderá ter que recompilar o módulo manualmente.

Você poderá se isto executando simplesmente o **kdesrc-build --refresh-build módulo**.

Se você quiser que o kdesrc-build recompile de forma automática o módulo, durante a próxima atualização normal, você poderá criar um arquivo especial. Cada módulo possui uma pasta de compilação. Se você criar um arquivo chamado `.refresh-me` na pasta de compilação de um módulo, o kdesrc-build irá reconstruir o módulo da próxima vez que o processo de compilação rodar, mesmo que fosse executar normalmente a compilação incremental.

DICA

Por padrão, a pasta de compilação é a `~/kdesrc/build/ módulo / .`. Se você mudar a opção [build-dir](#), então use essa em vez da `~/kdesrc/build`.

Recompilar usando o `.refresh-me` para o módulo `kdelibs`:

```
% touch ~/kdesrc/build/kdelibs/.refresh-me
% kdesrc-build
```

6.3.4 Mudando a configuração das variáveis de ambiente

Normalmente, o kdesrc-build usa o ambiente que está presente quando inicia e executa os programas para atualizar e compilar. Isto é útil para quando você está rodando o kdesrc-build a partir da linha de comando.

Contudo, você poderá desejar alterar a configuração das variáveis de ambiente para as quais o kdesrc-build não possui uma opção direta. (Por exemplo, para configurar qualquer variável de ambiente ao rodar o kdesrc-build em um agendador de tarefas como o Cron) Isto é possível com a opção [set-env](#).

Ao contrário da maioria das opções, ela poderá ser definida mais de uma vez e aceita dois itens separados por um espaço. O primeiro item é o nome da variável de ambiente a definir e o resto da linha é o valor.

Configura a **DISTRO=BSD** para todos os módulos:

```
global
  set-env DISTRO BSD
end global
```


6.3.5 Prosseguir com as compilações

6.3.5.1 Prosseguir uma compilação falhada ou cancelada

Você poderá indicar ao kdesrc-build para começar a compilar a partir de um módulo diferente daquele onde iria prosseguir. Isto poderá ser útil quando falhou um conjunto de módulos ou se tiver cancelado a execução de uma compilação no meio. Você poderá controlar isto com as opções `--resume-from` e `--resume-after`.

NOTA

As versões mais antigas do kdesrc-build iriam saltar a atualização do código ao prosseguir com uma compilação. Isto já não é mais feito por padrão, mas você ainda poderá continuar a usar a opção da linha de comando `--no-src` para saltar a atualização do código.

Prosseguir a compilação começando no 'kdebase':

```
% kdesrc-build --resume-from=kdebase
```

Prosseguir a compilação começando após o 'kdebase' (no caso de você corrigir manualmente um problema e ter instalado o módulo você mesmo):

```
% kdesrc-build --resume-after=kdebase
```

Se a última compilação do kdesrc-build terminou com erro, você poderá também usar a opção `--resume` da linha de comando, a qual prossegue com a última compilação a partir do módulo que falhou. As atualizações de código e metadados também são ignoradas (porém, se precisar delas, normalmente é melhor usar a opção `--resume-from` em alternativa).

6.3.5.2 Ignorando os módulos numa compilação

Como acontece quando deseja [prosseguir com a compilação de um módulo](#), você poderá optar em alternativa por atualizar e compilar tudo normalmente, só que ignorando um conjunto de módulos.

Você poderá fazer isso com a opção `--ignore-modules`. Esta opção diz ao kdesrc-build para ignorar todos os módulos da linha de comando, ao efetuar a atualização e a compilação.

Ignorar o extragear/multimedia e o kdereview numa execução completa:

```
% kdesrc-build --ignore-modules extragear/multimedia kdereview
```

6.3.6 Mudando as opções na linha de comando

6.3.6.1 Modificando as opções globais

Você poderá alterar a configuração das opções lidas do [arquivo de configuração](#) diretamente a partir da linha de comando. Esta alteração irá substituir a configuração do arquivo, mas é apenas temporária. Só terá efeito enquanto ainda estiver presente na linha de comando.

O kdesrc-build permite-lhe alterar as opções com nome do tipo *nome-opção* passando um argumento na linha de comando com o formato `--nome-opção=valor`. O kdesrc-build irá reconhecer se não souber qual a opção, e irá procurar o nome na sua lista de nomes de opções. Se não reconhecer o nome, ele irá avisá-lo, caso contrário irá recordar o valor que definiu e substituir o valor do arquivo de configuração.

Configurar o `source-dir` como `/dev/null` para fins de teste:

```
% kdesrc-build --pretend --source-dir=/dev/null
```

6.3.6.2 Mudando as opções do módulo

Também é possível alterar as opções apenas para um módulo específico. A sintaxe é semelhante: `--módulo,nome-opção=valor`.

Esta alteração substitui qualquer opção duplicada do módulo encontrado no [arquivo de configuração](#) e só se aplica enquanto a opção for passada na linha de comando.

Usando uma pasta de compilação diferente para o módulo 'kdeedu':

```
% kdesrc-build --kdeedu,build-dir=temp-build
```

6.4 Funcionalidades para os programadores do KDE

6.4.1 Verificações do Agente do SSH

O kdesrc-build poderá certificar-se de que os programadores do KDE que usem o SSH para acessar ao repositório de código do KDE não se esqueçam acidentalmente da ferramenta SSH-Agent ligada. Isto poderá fazer com que o kdesrc-build fique eternamente à espera que o programador insira a sua senha de SSH, por isso e por padrão o kdesrc-build irá verificar se o Agente está rodando antes de efetuar as atualizações do código.

NOTA

Isto só é feito para os programadores do KDE que usam o SSH. Isto acontece porque não é necessária qualquer senha para a atualização e extração de código anônima do repositório. O Subversion irá lidar com as senhas para o segundo protocolo possível dos programadores do KDE, o HTTPS.

Você poderá querer desativar a verificação do Agente do SSH, no caso das situações em que o kdesrc-build não detecte bem a presença de um agente. Para desativar a verificação do agente, configure a opção `disable-agent-check` como **true**.

Desativando a verificação do agente do SSH:

```
global
  disable-agent-check true
end global
```

6.5 Outras funcionalidades do kdesrc-build

6.5.1 Mudando a quantidade de mensagens do resultado do kdesrc-build

O kdesrc-build possui várias opções para controlar a quantidade de dados de saída que o programa gera. Em qualquer caso, os erros serão sempre apresentados.

- A opção `--quiet` (o seu formato curto é o `-q`) faz com que o kdesrc-build seja majoritariamente silencioso. Só as mensagens importantes, avisos ou erros serão apresentados. Quando estiver disponível, a informação de progresso é apresentada.
- A opção `--really-quiet` (sem formato curto) faz com que o kdesrc-build mostre apenas os avisos importantes ou os erros, enquanto estiver em execução.
- A opção `--verbose` (o formato curto é o `-v`) faz com que o kdesrc-build seja muito detalhado nos seus resultados.
- A opção `--debug` serve apenas para fins de depuração, uma vez que faz com que o kdesrc-build atue como se o `--verbose` estivesse ativo, fazendo com que os comandos sejam também apresentados no terminal e mostrando a informação de depuração para várias funções.

6.5.2 Resultado colorido

Ao ser executado a partir do Konsole ou de um terminal diferente, o kdesrc-build irá mostrar normalmente o texto a cores.

Você poderá desativar isto com a opção `--no-color` na linha de comando ou configurando a opção `colorful-output` no [arquivo de configuração](#) como `false`.

Desativando as mensagens a cores no arquivo de configuração:

```
global
  colorful-output false
end global
```

6.5.3 Removendo as pastas desnecessárias após uma compilação

Você está com falta de espaço em disco mas deseja ainda fazer uma extração atualizada do KDE? O kdesrc-build poderá ajudar a reduzir o uso do seu disco ao compilar o KDE a partir do Subversion.

NOTA

Tenha em mente que a compilação do KDE ocupa bastante espaço. Existem muitos componentes grandes em termos de espaço ao usar o kdesrc-build:

1. A extração de código atual poderá levar bastante espaço. Os módulos padrão possuem cerca de 1,6 gigabytes de espaço em disco. Você poderá reduzir esta quantidade, se tiver certeza que só vai compilar os módulos que deseja de fato. O kdesrc-build não irá remover o código-fonte do disco, mesmo que remova o item do [arquivo de configuração](#), assim certifique-se de que são apagados os módulos extraídos e não usados da pasta de código. Lembre-se que os arquivos de código são obtidos da Internet, assim você *não deverá* removê-los se for usar estes arquivos de fato, pelo menos até acabar de usar o kdesrc-build.

Da mesma forma, se você já tiver um Qt™ instalado pela sua distribuição (e é bastante possível que isso aconteça), provavelmente não irá necessitar instalar o módulo 'qt'. Isso irá poupar cerca de 200 MB do código no disco.

Uma coisa a observar é que, devido à forma como o Subversion funciona: existem de fato dois arquivos no disco para cada arquivo extraído do repositório. O kdesrc-build não está apto por enquanto a minimizar o tamanho do código, quando ele não estiver sendo usado.

2. O kdesrc-build irá criar uma pasta de compilação separada, para onde irá compilar o código-fonte. Algumas vezes o kdesrc-build terá que copiar uma pasta de código para criar uma pasta de compilação falsa. Quando isso acontecer, são usadas as ligações simbólicas, caracterizadas por poupar espaço, assim isso não deverá ser um problema a nível de espaço em disco. A pasta de compilação será tipicamente muito maior que a pasta de código de um módulo. Por exemplo, a pasta de compilação do 'kdebase' ocupa cerca de 1050 megabytes, enquanto o código do 'kdebase' ocupa apenas cerca de 550 megabytes.

Felizmente, a pasta de compilação não é necessária após um módulo ter sido compilado e instalado com sucesso. O kdesrc-build poderá remover automaticamente a pasta de compilação após a instalação de um módulo; veja os exemplos abaixo para mais informações. Lembre-se que a utilização deste passo tornará impossível ao kdesrc-build efetuar as compilações incrementais que poupam tempo.

3. Finalmente, existe o espaço em disco necessário para a própria instalação do KDE, que não se executa a partir da pasta de compilação. Este espaço é normalmente menor que o da pasta de compilação. É mais difícil ter os valores exatos, de qualquer forma.

Como reduz os requisitos de espaço do KDE? Uma forma é usar as opções de compilação adequadas, para otimizar a redução de espaço em vez da velocidade. Outra forma, que poderá ser bem eficaz, é remover a informação de depuração da sua compilação do KDE.

ATENÇÃO

Você deverá saber bem o que está fazendo, antes de decidir remover a informação de depuração. A execução de aplicativos muito atuais significa que está rodando programas que, potencialmente, poderão ter tendência a travar, algo que não acontece numa versão estável. Se você estiver rodando 'software' sem informações de depuração, poderá ser bastante difícil criar um bom relatório de erros para que o seu erro seja resolvido; assim, é possível que tenha que reativar a informação de depuração do aplicativo afetado e recompilá-lo para ajudar um programador a resolver o problema. Então, remova a informação de depuração por sua conta e risco!

A remoção da pasta de compilação, após a instalação de um módulo. A pasta do código é ainda mantida e a depuração fica ativa:

```
global
  configure-flags      --enable-debug
  remove-after-install builddir      # Remover a pasta de compilação após ↵
    a instalação
end global
```

A remoção da pasta de compilação após a instalação, sem informações de depuração e com otimizações do tamanho.

```
global
  cxxflags             -Os             # Otimizar o tamanho
  configure-flags      --disable-debug
  remove-after-install builddir      # Remover a pasta de compilação após ↵
    a instalação
end global
```

Capítulo 7

CMake, o sistema de compilação do KDE 4

7.1 Introdução ao CMake

Em Março de 2006, o programa CMake ultrapassou os vários adversários e foi selecionado para ser o sistema de compilação do KDE 4, substituindo o sistema baseado nas 'autotools' que o KDE tem usado desde o início.

Está disponível uma página de introdução ao CMake no [TechBase do KDE](#). Basicamente, em vez de rodar o `make -f Makefile.cvs`, depois o `configure` e depois o Make, basta simplesmente rodar o CMake e depois o Make.

O kdesrc-build possui suporte para o CMake. Algumas funcionalidades do kdesrc-build são realmente funcionalidades do sistema de compilação subjacente, incluindo o [configure-flags](#) e o [do-not-compile](#). Quando estiverem disponíveis funcionalidades equivalentes, serão disponibilizadas. Por exemplo, a equivalente da opção 'configure-flags' é a [cmake-options](#), e a opção [do-not-compile](#) também é suportada para o CMake desde a versão 1.6.3 do kdesrc-build.

Capítulo 8

Créditos e licença

Tradução de Marcus Gama marcus.gama@gmail.com

Revisão e atualização de André Marcelo Alvarenga alvarenga@kde.org

Esta documentação é licenciada sob os termos da [Licença de Documentação Livre GNU](#).

Apêndice A

Organização dos módulos e do código-fonte do KDE

A.1 O 'Módulo'

O KDE agrupa os seus programas em 'módulos' de diversos tamanhos. Isto era apenas um agrupamento lógico de alguns módulos grandes, mas com a introdução dos [repositórios de código-fonte](#) baseados no [Git](#), estes grandes módulos foram por sua vez divididos em muitos módulos mais pequenos.

O kdesrc-build também usa este conceito de módulos. Em resumo, um 'módulo' é um agrupamento de código que poderá ser transferido, compilado, testado e instalado.

A.1.1 Módulos individuais

É fácil indicar ao kdesrc-build para compilar um único módulo. Na lista a seguir está um exemplo de como deveria ser uma declaração de um módulo baseado em Subversion no [arquivo de configuração](#).

```
module kdefoo
  cmake-options -DCMAKE_BUILD_TYPE=Debug
end module
```

DICA

Este é um módulo baseado no Subversion, uma vez que não usa uma opção [repository](#). Do mesmo modo, a opção `cmake-options` aparece apenas como exemplo, não sendo obrigatória.

A.1.2 Grupos de módulos relacionados

Agora, a maioria dos módulos de código do KDE são baseados no Git do KDE, estando normalmente reunidos em grupos de módulos.

Deste modo, o kdesrc-build também suporta grupos de módulos, usando os [conjuntos de módulos](#). Por exemplo:

Manual do script kdesrc-build

```
module-set base-modules
  repository kde-projects
  use-modules kde-runtime kde-workspace kde-baseapps
end module-set
```

DICA

Você poderá deixar o nome do conjunto de módulos (*módulos-base* neste caso) vazio se o desejar. Esta opção `repository` indica ao `kdesrc-build` de onde transferir o código, mas poderá também usar uma URL `git://`.

Uma funcionalidade especial do ‘`repository kde-projects`’ é que o `kdesrc-build` irá incluir automaticamente todos os módulos do Git que estiverem agrupados sob os módulos que apresentar (na base de dados do Projeto KDE).

A.1.3 ‘Grupos de ramificações’ do módulo

Levando adiante o conceito de [grupo de módulos](#), os desenvolvedores do KDE descobriram eventualmente que a sincronização dos nomes das ramificações Git por uma grande quantidade de repositórios estava ficando mais difícil, especialmente durante o esforço de desenvolvimento das novo KDE Frameworks para a Qt™ 5.

Assim, o conceito de ‘grupos de ramificações’ foi desenvolvido de modo a permitir aos usuários e desenvolvedores selecionar um de alguns grupos, permitindo ao script selecionar automaticamente a ramificação Git apropriada.

O `kdesrc-build` tem suporte a essa funcionalidade desde a versão 1.16-pre2, através da opção [branch-group](#).

Example A.1 Exemplo de uso do branch-group

O `branch-group` pode ser usado no arquivo de configuração da seguinte forma:

```
global
  # Selecionar o KDE Frameworks 5 e outros aplicativos baseados na Qt5
  branch-group kf5-qt5

  # Outras opções globais aqui...
end global

module-set
  # O branch-group só funciona para o kde-projects
  repository kde-projects

  # O branch-group herda da definição global, mas pode ser definido aqui.

  use-modules kdelibs kde-workspace
end module-set

# A ramificação kdelibs será a "frameworks"
# A ramificação kde-workspace será a "master" (em agosto de 2013)
```

Nesse caso, o mesmo `branch-group` devolve diferentes nomes de ramificações para cada módulo Git.

Manual do script kdesrc-build

Esta funcionalidade requer a manutenção de alguns dados por parte dos desenvolvedores do KDE em um repositório Git chamado `kde-build-metadata`. Contudo, esse módulo será incluído automaticamente pelo `kdesrc-build` (ainda que você possa vê-lo aparecer no resultado do script).

DICA

Os módulos do KDE que não têm um nome de ramificação definido para o grupo de ramificações que você escolher terão, por padrão, um nome de ramificação apropriado, como se não tivesse indicado o `branch-group` em tudo.

Apêndice B

Procedimentos de configuração do perfil obsoletos

B.1 Configurar um perfil de autenticação do KDE

Estas instruções descrevem como configurar o perfil necessário para garantir que o seu computador consegue iniciar uma sessão no seu ambiente de trabalho Plasma do KDE recém compilada. O kdesrc-build irá tentar fazer isto de forma automática (veja em Seção 2.5.1). Esta seção do apêndice poderá ser útil para os que não consigam usar o suporte do kdesrc-build para a configuração dos perfis de autenticação. Contudo, as instruções nem sempre estão atualizadas, podendo ser útil consultar o arquivo `kde-env-master.sh` que vem incluído com o código do kdesrc-build.

B.1.1 Alterando a configuração do seu perfil inicial

IMPORTANTE

O arquivo `.bash_profile` é o arquivo de configuração do usuário na linha de comando `bash`, que é usada por muitas distribuições Linux[®]. Se você usar uma linha de comando diferente, então poderá ter que ajustar os exemplos dados nesta seção, para a sua linha de comando em particular.

Abra ou crie o arquivo `.bash_profile` na pasta pessoal com o seu editor favorito e adicione ao fim do arquivo: Se você estiver compilando o módulo 'qt' (o padrão), adicione em alternativa:

```
QTDIR=(local da qtdir) # Por padrão, ~/kdesrc/build/qt.
KDEDIR=(local da kdedir) # Por padrão ~/kde.
KDEDIRS=$KDEDIR
PATH=$KDEDIR/bin:$QTDIR/bin:$PATH
MANPATH=$QTDIR/doc/man:$MANPATH

# Atuar adequadamente se a LD_LIBRARY_PATH não estiver definida
if [ -z $LD_LIBRARY_PATH ]; then
    LD_LIBRARY_PATH=$KDEDIR/lib:$QTDIR/lib
else
    LD_LIBRARY_PATH=$KDEDIR/lib:$QTDIR/lib:$LD_LIBRARY_PATH
fi

export QTDIR KDEDIRS PATH MANPATH LD_LIBRARY_PATH
```

Manual do script kdesrc-build

ou, se não estiver compilando o 'qt' (e usar o Qt™ do seu sistema), adicione isto em alternativa:

```
KDEDIR=(local da kdedir) # Por padrão ~/kde.
KDEDIRS=$KDEDIR
PATH=$KDEDIR/bin:$QTDIR/bin:$PATH

# Atuar adequadamente se a LD_LIBRARY_PATH não estiver definida.
if [ -z $LD_LIBRARY_PATH ]; then
    LD_LIBRARY_PATH=$KDEDIR/lib
else
    LD_LIBRARY_PATH=$KDEDIR/lib:$LD_LIBRARY_PATH
fi

export KDEDIRS PATH LD_LIBRARY_PATH
```

Se você não estiver usando um usuário dedicado, defina um novo \$KDEHOME para o seu novo ambiente no seu arquivo .bash_profile:

```
export KDEHOME="${HOME}/.kde-svn"

# Criar a pasta se necessário
[ ! -e ~/.kde-svn ] && mkdir ~/.kde-svn
```

NOTA

Se, mais tarde, o seu Menu K estiver vazio ou completamente cheio de aplicativos da sua distribuição, você poderá ter que definir as suas variáveis de ambiente do XDG no seu .bash_profile:

```
XDG_CONFIG_DIRS="/etc/xdg"
XDG_DATA_DIRS="${KDEDIR}/share:/usr/share"
export XDG_CONFIG_DIRS XDG_DATA_DIRS
```

B.1.2 Iniciando o KDE

Agora que você ajustou a sua configuração do ambiente para usar o KDE correto, é importante certificar-se de que o programa **startkde** correto seja usado também.

Abra o arquivo de texto .xinitrc da sua pasta pessoal, ou crie-o se necessário. Adicione a linha:

```
exec ${KDEDIR}/bin/startkde
```

IMPORTANTE

Em algumas distribuições, poderá ser necessário efetuar os mesmos passos com o arquivo .xsessi on, também na pasta pessoal. Isto aplica-se especialmente nos gerenciadores de login gráficos como o kdm, o gdm ou o xdm.

Agora inicie o seu novo KDE: nos sistemas BSD e Linux®, com o suporte para terminais virtuais, os atalhos **Ctrl+Alt+F1 ... Ctrl+Alt+F12** são usados para mudar para o Console Virtual 1 a 12. Isto permite-lhe executar mais de um ambiente gráfico ao mesmo tempo. Os primeiros seis são terminais de texto e os seis seguintes são gráficos.

Se, quando você iniciar o seu computador, aparecer o gerenciador gráfico de login, você poderá usar o novo ambiente do KDE, mesmo que não apareça como uma opção. A maioria dos gerenciadores de login, incluindo o kdm, possuem uma opção para usar uma 'Sessão Personalizada' ao

Manual do script kdesrc-build

se logar. Com esta opção, a sua configuração de sessões é carregada do arquivo `.xsession` da sua pasta pessoal. Se você já tiver modificado este arquivo como descrito acima, esta opção deverá carregar a sua nova instalação do KDE.

Caso contrário, existe algo que você poderá tentar que normalmente funciona: Pressione **Ctrl+Alt+F2**, para que apareça um terminal de texto. Logue-se com o usuário dedicado e digite:

```
startx -- :1
```

DICA

Você poderá executar o KDE a partir do código e o KDE antigo ao mesmo tempo! Logue-se com o seu usuário normal e inicie o ambiente de trabalho do KDE estável. Pressione **Ctrl+Alt+F2** (ou **F1**, **F3**, etc..), e verá um terminal de texto. Logue-se com o usuário dedicado ao Subversion do KDE e digite:

```
startx -- :1
```

Você poderá voltar ao ambiente de trabalho do KDE do seu usuário normal, pressionando a tecla de atalho do ambiente em execução. Ela é normalmente o **Ctrl+Alt+F7**, sendo que você poderá também precisar usar o **F6** ou o **F8** em alternativa. Para voltar ao seu KDE compilado pelo `kdesrc-build`, você irá usar a mesma sequência, usando, no entanto, a tecla de função seguinte. Por exemplo, se você precisar inserir **Ctrl+Alt+F7** para mudar para o seu KDE normal, poderá ter que inserir **Ctrl+Alt+F8** para voltar ao seu KDE do `kdesrc-build`.